

Contents

1	Package <code>com.imagos.axilice.client.traceability.gui</code>	8
1.1	Classes	9
1.1.1	CLASS <code>TraceabilityTreeItem</code>	9
1.1.2	CLASS <code>TraceabilityTreeView</code>	15
2	Package <code>com.imagos.axilice.core.extTables</code>	24
2.1	Classes	25
2.1.1	CLASS <code>ExtFieldBooleanValueOT</code>	25
2.1.2	CLASS <code>ExtFieldDateValueOT</code>	26
2.1.3	CLASS <code>ExtFieldLongValueOT</code>	27
2.1.4	CLASS <code>ExtFieldNumberValueOT</code>	29
2.1.5	CLASS <code>ExtFieldOT</code>	30
2.1.6	CLASS <code>ExtFieldStringValueOT</code>	34
2.1.7	CLASS <code>ExtFieldValueOT</code>	35
2.1.8	CLASS <code>ExtFieldValuePK</code>	36
2.1.9	CLASS <code>ExtTableDAO</code>	37
2.1.10	CLASS <code>ExtTableException</code>	43
2.1.11	CLASS <code>ExtTableLN</code>	46
2.1.12	CLASS <code>ExtTableOT</code>	50
3	Package <code>com.imagos.axilice.core.management</code>	53
3.1	Interfaces	54
3.1.1	INTERFACE <code>ManagementManager</code>	54
3.2	Classes	54
3.2.1	CLASS <code>DocumentLineOT</code>	54
3.2.2	CLASS <code>DocumentOT</code>	58
3.2.3	CLASS <code>DocumentTypeOT</code>	65
3.2.4	CLASS <code>MovementOT</code>	68
3.2.5	CLASS <code>SerialNumberOT</code>	72
4	Package <code>com.imagos.axilice.core.traceability.tree.parser</code>	75
4.1	Classes	76
4.1.1	CLASS <code>TraceabilityTreeEntityResolver</code>	76
4.1.2	CLASS <code>TraceabilityTreeParser</code>	77
5	Package <code>com.imagos.axilice.core.partners</code>	86
5.1	Interfaces	87
5.1.1	INTERFACE <code>PartnersManager</code>	87
5.2	Classes	87
5.2.1	CLASS <code>AddressOT</code>	87

5.2.2	CLASS BankAccountOT	91
5.2.3	CLASS ContactOT	93
5.2.4	CLASS DepartmentOT	96
5.2.5	CLASS DiscountOT	98
5.2.6	CLASS PartnerGroupOT	101
5.2.7	CLASS PartnerOT	102
5.2.8	CLASS PaymentMethodOT	108
5.2.9	CLASS RappelIntervalOT	111
5.2.10	CLASS RappelTypeOT	113
5.2.11	CLASS VendorOT	115
6	Package com.imagos.axilice.server.common.dao	119
6.1	Classes	120
6.1.1	CLASS BarcodeTypeDAO	120
6.1.2	CLASS CommonDAO	121
6.1.3	CLASS VatTypeDAO	122
7	Package com.imagos.axilice.client.warehouse	125
7.1	Classes	126
7.1.1	CLASS LotClient	126
7.1.2	CLASS MeasureUnitClient	139
7.1.3	CLASS ProductClient	152
7.1.4	CLASS ProductClientListener	166
7.1.5	CLASS ProductGroupClient	167
7.1.6	CLASS ProductStockClient	175
7.1.7	CLASS StockClient	188
7.1.8	CLASS TestClient	201
7.1.9	CLASS WarehouseClient	202
8	Package com.imagos.axilice.core.traceability.tree	216
8.1	Classes	217
8.1.1	CLASS AppccInfo	217
8.1.2	CLASS CommentInfo	217
8.1.3	CLASS ProcessInfo	217
8.1.4	CLASS ProductInfo	222
8.1.5	CLASS PropertyInfo	229
8.1.6	CLASS PurchaseInfo	234
8.1.7	CLASS TraceabilityTree	235
8.1.8	CLASS TraceabilityTreeBuilder	237
8.1.9	CLASS WarehouseInfo	238
8.1.10	CLASS WorkcenterInfo	240
8.1.11	CLASS WorkerInfo	242
9	Package com.imagos.axilice.core.warehouse	244
9.1	Interfaces	245
9.1.1	INTERFACE WarehouseManager	245
9.2	Classes	246
9.2.1	CLASS LotOT	246
9.2.2	CLASS MeasureUnitOT	249
9.2.3	CLASS ProductGroupOT	252

9.2.4	CLASS ProductOT	255
9.2.5	CLASS ProductStockOT	265
9.2.6	CLASS StockOT	268
9.2.7	CLASS WarehouseOT	270
10	Package com.imagos.axilice.server.util.db	275
10.1	Classes	276
10.1.1	CLASS CriteriaBean	276
10.1.2	CLASS CriteriaQueue	278
10.1.3	CLASS HibernateUtil	279
10.1.4	CLASS SQLGenerator	280
11	Package com.imagos.axilice.client.gui	282
11.1	Interfaces	284
11.1.1	INTERFACE AxiliceEventListener	284
11.1.2	INTERFACE GUIObject	284
11.2	Classes	285
11.2.1	CLASS AxiliceCoreView	285
11.2.2	CLASS AxiliceDateSelect	288
11.2.3	CLASS AxiliceDetailEntityView	288
11.2.4	CLASS AxiliceEditEntityView	293
11.2.5	CLASS AxiliceEntitySelectionView	294
11.2.6	CLASS AxiliceEntityView	296
11.2.7	CLASS AxiliceEvent	300
11.2.8	CLASS AxiliceExtensionFieldsView	301
11.2.9	CLASS AxiliceGUIBuilder	303
11.2.10	CLASS AxiliceInitView	305
11.2.11	CLASS AxiliceModuleView	306
11.2.12	CLASS AxiliceValueInput	307
11.2.13	CLASS EntityExtensionCatalogListener	308
11.2.14	CLASS EntityExtensionCatalogResizeListener	308
11.2.15	CLASS EntityExtensionCatalogTextListener	309
11.2.16	CLASS GUIOperation	309
11.2.17	CLASS ImageUtils	310
12	Package com.imagos.axilice.server.management	311
12.1	Classes	312
12.1.1	CLASS ManagementManagerBuilder	312
12.1.2	CLASS ManagementManagerImpl	312
13	Package com.imagos.axilice.server	314
13.1	Classes	315
13.1.1	CLASS ServerContext	315
13.1.2	CLASS ServerManager	316
14	Package com.imagos.axilice.core	318
14.1	Classes	319
14.1.1	CLASS AxiliceException	319
14.1.2	CLASS HibernateEntityResolver	321

15 Package com.imagos.axilice.client.management	322
15.1 Classes	323
15.1.1 CLASS DocumentClient	323
15.1.2 CLASS DocumentLineClient	336
15.1.3 CLASS DocumentTypeClient	349
15.1.4 CLASS LotOutItemListener	361
15.1.5 CLASS LotOutListener	362
15.1.6 CLASS LotOutSelector	363
15.1.7 CLASS MovementClient	364
15.1.8 CLASS SerialNumberClient	377
16 Package com.imagos.axilice.server.management.bl	391
16.1 Classes	392
16.1.1 CLASS ManagementBLL	392
17 Package com.imagos.axilice.client.gui.common	393
17.1 Classes	394
17.1.1 CLASS BarcodeTypeClient	394
17.1.2 CLASS CommonClient	402
17.1.3 CLASS CurrencyClient	415
17.1.4 CLASS MapViewer	428
17.1.5 CLASS PriceListClient	429
17.1.6 CLASS PriceListLineClient	442
17.1.7 CLASS VatTypeClient	455
18 Package com.imagos.axilice.core.ssl	463
18.1 Classes	464
18.1.1 CLASS RMISSLClientSocketFactory	464
18.1.2 CLASS RMISSLServerSocketFactory	464
19 Package com.imagos.axilice.server.common.bl	465
19.1 Classes	466
19.1.1 CLASS CommonBLL	466
20 Package com.imagos.axilice.server.warehouse.bl	468
20.1 Classes	469
20.1.1 CLASS ProductBLL	469
21 Package com.imagos.axilice.core.util	471
21.1 Classes	473
21.1.1 CLASS AddressUtils	473
21.1.2 CLASS BarcodeGenerator	474
21.1.3 CLASS BaseVObyIdComparator	475
21.1.4 CLASS Converter	475
21.1.5 CLASS DateTime	477
21.1.6 CLASS FieldValue	480
21.1.7 CLASS GoogleMap	481
21.1.8 CLASS ImageConverter	482
21.1.9 CLASS InstanceLocator	483
21.1.10 CLASS LocalizedResultMessage	484
21.1.11 CLASS LoggerFactory	489

21.1.12	CLASS	MapUtils	491
21.1.13	CLASS	OperationResult	492
21.1.14	CLASS	ReflectionUtils	493
21.1.15	CLASS	ResultMessage	494
21.1.16	CLASS	ResultMessageCatalog	498
21.1.17	CLASS	SortUtils	504
21.1.18	CLASS	Validator	505
22	Package	com.imagos.axilice.client	513
22.1	Classes		514
22.1.1	CLASS	AutoUpdateInitProgressBarOperation	514
22.1.2	CLASS	BuildEntityViewOperation	514
22.1.3	CLASS	BuildGuiOperation	515
22.1.4	CLASS	BuildGUIThread	515
22.1.5	CLASS	ClientContext	517
22.1.6	CLASS	ClientManager	522
22.1.7	CLASS	CloseInitializingWindowOperation	523
22.1.8	CLASS	GetInitProgressBarOperation	523
22.1.9	CLASS	InitClientOperation	524
22.1.10	CLASS	UpdateInitProgressBarOperation	526
23	Package	com.imagos.axilice.server.management.dao	527
23.1	Classes		528
23.1.1	CLASS	ManagementDAO	528
24	Package	com.imagos.axilice.core.installer	529
24.1	Interfaces		531
24.1.1	INTERFACE	Installer	531
24.2	Classes		531
24.2.1	CLASS	AreaOT	531
24.2.2	CLASS	EntityLauncher	532
24.2.3	CLASS	EntityLauncherFactory	541
24.2.4	CLASS	EntityOT	541
24.2.5	CLASS	FieldOT	545
24.2.6	CLASS	ModuleEntityResolver	548
24.2.7	CLASS	ModuleLauncher	549
24.2.8	CLASS	ModuleOT	551
24.2.9	CLASS	ModuleParser	555
24.2.10	CLASS	ResourceException	559
24.2.11	CLASS	ResourceOT	561
24.2.12	CLASS	SubentityOT	563
25	Package	com.imagos.axilice.server.traceability	565
25.1	Classes		566
25.1.1	CLASS	TracerImpl	566
26	Package	com.imagos.axilice.server.installer	568
26.1	Classes		569
26.1.1	CLASS	InstallerImpl	569
26.1.2	CLASS	ResourceDAO	570
26.1.3	CLASS	ResourceLN	572

26.1.4	CLASS	UpdateManager	574
27	Package	com.imagos.axilice.client.partners	575
27.1	Classes		576
27.1.1	CLASS	AddressClient	576
27.1.2	CLASS	AddressListener	589
27.1.3	CLASS	BankAccountClient	590
27.1.4	CLASS	ContactClient	602
27.1.5	CLASS	DepartmentClient	615
27.1.6	CLASS	DiscountClient	628
27.1.7	CLASS	PartnerClient	641
27.1.8	CLASS	PartnerGroupClient	654
27.1.9	CLASS	PaymentMethodClient	666
27.1.10	CLASS	RappelIntervalClient	679
27.1.11	CLASS	RappelTypeClient	692
27.1.12	CLASS	RappelTypeListener	704
27.1.13	CLASS	VendorClient	705
28	Package	com.imagos.axilice.server.common	719
28.1	Classes		720
28.1.1	CLASS	BaseDAO	720
28.1.2	CLASS	CommonManagerBuilder	721
28.1.3	CLASS	CommonManagerImpl	722
29	Package	com.imagos.axilice.server.warehouse	726
29.1	Classes		727
29.1.1	CLASS	WarehouseManagerBuilder	727
29.1.2	CLASS	WarehouseManagerImpl	727
30	Package	com.imagos.axilice.core.traceability.tree.bl	730
30.1	Classes		731
30.1.1	CLASS	TraceabilityTreeBLL	731
31	Package	com.imagos.axilice.core.common	733
31.1	Interfaces		735
31.1.1	INTERFACE	CommonManager	735
31.2	Classes		737
31.2.1	CLASS	BarcodeTypeOT	737
31.2.2	CLASS	BaseVO	740
31.2.3	CLASS	CountryOT	741
31.2.4	CLASS	CurrencyOT	743
31.2.5	CLASS	DocumentTotal	746
31.2.6	CLASS	FlagOT	749
31.2.7	CLASS	ImageComparator	751
31.2.8	CLASS	ImageOT	752
31.2.9	CLASS	LineaDesgloseIVA	755
31.2.10	CLASS	MultiSelectionSubentityOT	756
31.2.11	CLASS	PriceListLineOT	758
31.2.12	CLASS	PriceListOT	761
31.2.13	CLASS	RateOT	763
31.2.14	CLASS	RegistryOT	765

31.2.15 CLASS SelectionSubentityOT	766
31.2.16 CLASS StateOT	768
31.2.17 CLASS ValueOT	770
31.2.18 CLASS VatTypeOT	772
32 Package com.imagos.axilice.server.warehouse.dao	777
32.1 Classes	778
32.1.1 CLASS MeasureUnitDAO	778
32.1.2 CLASS ProductDAO	780
32.1.3 CLASS ProductGroupDAO	782
33 Package com.imagos.axilice.client.traceability	785
33.1 Classes	786
33.1.1 CLASS TraceabilityTreeView	786
34 Package com.imagos.axilice.core.traceability	787
34.1 Interfaces	788
34.1.1 INTERFACE Tracer	788
34.2 Classes	788
34.2.1 CLASS TracerException	788
35 Package com.imagos.axilice.client.installer	791
35.1 Classes	792
35.1.1 CLASS ClientClassLoader	792
35.1.2 CLASS ClientClasspathUpdater	794
35.1.3 CLASS ClientInstaller	796

Chapter 1

Package

com.imagos.axilice.client.traceability.gui

Package Contents

Page

Classes

TraceabilityTreeItem	9
<i>GUI class that implements a Traceability Tree Item.</i>	
TraceabilityTreeView	15
<i>Helper class that implements the Traceability Tree Viewer GUI.</i>	

1.1 Classes

1.1.1 CLASS TraceabilityTreeItem

GUI class that implements a Traceability Tree Item. It extends the TreeItem GUI Object to add extra data as the Info object linked to the TreeItem or the TreeItem name.
Responsibilities:

- Define traceability item types.
- Store the tree item info.
- Build the GUI for any traceability tree item.
- Tree item conversion to String or XML format.

DECLARATION

```
public class TraceabilityTreeItem
extends java.lang.Object
```

FIELDS

- public static final int NODE_UNDEFINED
 - Constant definition with the item type UNDEFINED. It's the default value to point out that no item type has been selected.
- public static final int NODE_PRODUCT
 - Constant definition with the item type PRODUCT.
- public static final int NODE_PROCESS
 - Constant definition with the item type PROCESS.
- public static final int NODE_WORKER
 - Constant definition with the item type WORKER.
- public static final int NODE_WORKCENTER
 - Constant definition with the item type WORKCENTER.
- public static final int NODE_WAREHOUSE
 - Constant definition with the item type WAREHOUSE.
- public static final int NODE_PURCHASE
 - Constant definition with the item type PURCHASE.
- public static final int NODE_PROPERTY
 - Constant definition with the item type PROPERTY.
- public static final int NODE_APPCC
 - Constant definition with the item type APPCC.
- public static final int NODE_COMMENT
 - Constant definition with the item type COMMENT.

CONSTRUCTORS

• *TraceabilityTreeItem*

```
public TraceabilityTreeItem( SwtToolkit  swtToolkit,
com.imagos.axilice.client.traceability.gui.TraceabilityTreeItem  parent, int
style, java.lang.Object  info, java.lang.String  text, java.lang.String
iconFile )
```

– Usage

- * Constructor for a non-root traceability tree item. It's used when the item created is any but the main product item of the traceability file.

– Parameters

- * **swtToolkit** - SwtToolkit, toolkit instance used to build the GUI for the tree items.
- * **parent** - TraceabilityTreeItem, parent item.
- * **style** - int, SWT constant with the style configuration for the item. By default it's SWT.NONE.
- * **info** - Object, info object with the data linked to the tree item.
- * **text** - String, text corresponding to the tree item, that will be showed on screen.
- * **iconFile** - String, path of the icon for the tree item.

– See Also

- * org.eclipse.swt.SWT
-

• *TraceabilityTreeItem*

```
public TraceabilityTreeItem( SwtToolkit  swtToolkit,
com.imagos.axilice.client.traceability.gui.TraceabilityTreeItem  parent, int
style, java.lang.Object  info, java.lang.String  text, java.lang.String
iconFile, int  index )
```

– Usage

- * Constructor for a non-root traceability tree item. It's used when the item created is any but the main product item of the traceability file.

– Parameters

- * **swtToolkit** - SwtToolkit, toolkit instance used to build the GUI for the tree items.
- * **parent** - TraceabilityTreeItem, parent item.
- * **style** - int, SWT constant with the style configuration for the item. By default it's SWT.NONE.
- * **info** - Object, info object with the data linked to the tree item.
- * **text** - String, text corresponding to the tree item, that will be showed on screen.
- * **iconFile** - String, path of the icon for the tree item.
- * **index** - int, index position at which the item will be inserted.

– See Also

- * org.eclipse.swt.SWT
-

• *TraceabilityTreeItem*

```
public TraceabilityTreeItem( SwtToolkit  swtToolkit,
com.imagos.axilice.client.traceability.gui.TraceabilityTreeItem  parent, int
style, java.lang.Object  info, java.lang.String  text, java.lang.String
iconFile, com.imagos.axilice.client.traceability.gui.TraceabilityTreeView
viewer, java.lang.String  handlerName )
```

– Usage

- * Constructor for a non-root traceability tree item. It's used when the item created is any but the main product item of the traceability file.

– **Parameters**

- * **swtToolkit** - SwtToolkit, toolkit instance used to build the GUI for the tree items.
- * **parent** - TraceabilityTreeItem, parent item.
- * **style** - int, SWT constant with the style configuration for the item. By default it's SWT.NONE.
- * **info** - Object, info object with the data linked to the tree item.
- * **text** - String, text corresponding to the tree item, that will be showed on screen.
- * **iconFile** - String, path of the icon for the tree item.
- * **viewer** - TraceabilityTreeView, event handler object.
- * **handlerName** - String, event handler method.

– **See Also**

- * org.eclipse.swt.SWT

• *TraceabilityTreeItem*

```
public TraceabilityTreeItem( SwtToolkit swtToolkit,
com.imagos.axilice.client.traceability.gui.TraceabilityTreeItem parent, int
style, java.lang.Object info, java.lang.String text, java.lang.String
iconFile, com.imagos.axilice.client.traceability.gui.TraceabilityTreeView
viewer, java.lang.String handlerName, int index )
```

– **Usage**

- * Constructor for a non-root traceability tree item. It's used when the item created is any but the main product item of the traceability file.

– **Parameters**

- * **swtToolkit** - SwtToolkit, toolkit instance used to build the GUI for the tree items.
- * **parent** - TraceabilityTreeItem, parent item.
- * **style** - int, SWT constant with the style configuration for the item. By default it's SWT.NONE.
- * **info** - Object, info object with the data linked to the tree item.
- * **text** - String, text corresponding to the tree item, that will be showed on screen.
- * **iconFile** - String, path of the icon for the tree item.
- * **viewer** - TraceabilityTreeView, event handler object.
- * **handlerName** - String, event handler method.
- * **index** - int, index position at which the item will be inserted.

– **See Also**

- * org.eclipse.swt.SWT

• *TraceabilityTreeItem*

```
public TraceabilityTreeItem( SwtToolkit swtToolkit, Tree parent, int
style, java.lang.Object info, java.lang.String text, java.lang.String
iconFile )
```

– **Usage**

- * Constructor for a root traceability tree item. It's used when the item created is the main product of the traceability file.

– **Parameters**

- * **swtToolkit** - SwtToolkit, toolkit instance used to build the GUI for the tree items.
- * **parent** - Tree, Tree widget.
- * **style** - int, SWT constant with the style configuration for the item. By default it's SWT.NONE.

- * **info** - Object, info object with the data linked to the tree item.
- * **text** - String, text corresponding to the tree item, that will be showed on screen.
- * **iconFile** - String, path of the icon for the tree item.

– **See Also**

- * org.eclipse.swt.SWT

• *TraceabilityTreeItem*

```
public TraceabilityTreeItem( SwtToolkit swtToolkit, Tree parent, int
style, java.lang.Object info, java.lang.String text, java.lang.String
iconFile, int index )
```

– **Usage**

- * Constructor for a root traceability tree item. It's used when the item created is the main product of the traceability file.

– **Parameters**

- * **swtToolkit** - SwtToolkit, toolkit instance used to build the GUI for the tree items.
- * **parent** - Tree, Tree widget.
- * **style** - int, SWT constant with the style configuration for the item. By default it's SWT.NONE.
- * **info** - Object, info object with the data linked to the tree item.
- * **text** - String, text corresponding to the tree item, that will be showed on screen.
- * **iconFile** - String, path of the icon for the tree item.
- * **index** - int, index position at which the item will be inserted.

– **See Also**

- * org.eclipse.swt.SWT

• *TraceabilityTreeItem*

```
public TraceabilityTreeItem( SwtToolkit swtToolkit, Tree parent, int
style, java.lang.Object info, java.lang.String text, java.lang.String
iconFile, com.imagos.axilice.client.traceability.gui.TraceabilityTreeView
viewer, java.lang.String handlerName )
```

– **Usage**

- * Constructor for a root traceability tree item. It's used when the item created is the main product item of the traceability file. It also adds an item selection event handler.

– **Parameters**

- * **swtToolkit** - SwtToolkit, toolkit instance used to build the GUI for the tree items.
- * **parent** - Tree, Tree widget.
- * **style** - int, SWT constant with the style configuration for the item. By default it's SWT.NONE.
- * **info** - Object, info object with the data linked to the tree item.
- * **text** - String, text corresponding to the tree item, that will be showed on screen.
- * **iconFile** - String, path of the icon for the tree item.
- * **viewer** - TraceabilityTreeView, event handler object.
- * **handlerName** - String, event handler method.

– **See Also**

- * org.eclipse.swt.SWT

• *TraceabilityTreeItem*

```
public TraceabilityTreeItem( SwtToolkit swtToolkit, Tree parent, int
```

```
style, java.lang.Object info, java.lang.String text, java.lang.String
iconFile, com.imagos.axilice.client.traceability.gui.TraceabilityTreeView
viewer, java.lang.String handlerName, int index )
```

– **Usage**

- * Constructor for a root traceability tree item. It's used when the item created is the main product item of the traceability file. It also adds an item selection event handler.

– **Parameters**

- * **swtToolkit** - SwtToolkit, toolkit instance used to build the GUI for the tree items.
- * **parent** - Tree, Tree widget.
- * **style** - int, SWT constant with the style configuration for the item. By default it's SWT.NONE.
- * **info** - Object, info object with the data linked to the tree item.
- * **text** - String, text corresponding to the tree item, that will be showed on screen.
- * **iconFile** - String, path of the icon for the tree item.
- * **viewer** - TraceabilityTreeView, event handler object.
- * **handlerName** - String, event handler method.
- * **index** - int, index position at which the item will be inserted.

– **See Also**

- * org.eclipse.swt.SWT

METHODS

- *getIconFile*

```
public String getIconFile( )
```

– **Usage**

- * Gets the traceability tree item icon file path.

– **Returns** - String, item icon file path.

- *getInfo*

```
public Object getInfo( )
```

– **Usage**

- * Gets the traceability tree item data object.

– **Returns** - Object, item data object.

- *getItemWidget*

```
public TreeItem getItemWidget( )
```

– **Usage**

- * Gets the traceability tree item widget.

– **Returns** - TreeItem, item widget.

- *getName*

```
public String getName( )
```

– **Usage**

- * Gets the traceability tree item name.

– **Returns** - String, item name.

-
- *getRevision*
public static String **getRevision**()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
-
- *getType*
public int **getType**()
 - **Usage**
 - * Gets the traceability tree item type.
 - **Returns** - int, item type.
-
- *setIconFile*
public void **setIconFile**(java.lang.String **iconFile**)
 - **Usage**
 - * Sets up the traceability tree item icon file path.
 - **Parameters**
 - * **iconFile** - String, an icon file path.
-
- *setInfo*
public void **setInfo**(java.lang.Object **info**)
 - **Usage**
 - * Sets up the traceability tree item data object.
 - **Parameters**
 - * **info** - Object, a data object.
-
- *setItemWidget*
public void **setItemWidget**(TreeItem **itemWidget**)
 - **Usage**
 - * Sets up the traceability tree item widget.
 - **Parameters**
 - * **itemWidget** - TreeItem, an item widget.
-
- *setName*
public void **setName**(java.lang.String **name**)
 - **Usage**
 - * Sets up the traceability tree item name.
 - **Parameters**
 - * **name** - String, a name.
-
- *setType*
public void **setType**(int **type**)
 - **Usage**
 - * Sets up the traceability tree item type.

- **Parameters**
 - * **type** - int, an item type.
-
- *toString*
public String toString()
 - **Usage**
 - * Converts the traceability tree item to a formatted String that show all the item's data.
 - **Returns** - String, formatted String with item's data.
-
- *toXml*
public String toXml(java.lang.String prefix)
 - **Usage**
 - * Converts the traceability tree item to a formatted String in XML format, that will hold all the item's data.
 - **Parameters**
 - * **prefix** - String, prefix added to each of the XML lines resulting from the item conversion.
 - **Returns** - String, formatted String with the item's data in XML format.
-
- *updateInfo*
public void updateInfo(java.lang.Object info)
 - **Usage**
 - * Updates the traceability tree item data object. Sets up all the data (of the corresponding item type) kept in the tree item data object. That means that it will overwrite only the object data fields of the item type level.
 - **Parameters**
 - * **info** - Object, a data object.

1.1.2 CLASS TraceabilityTreeView

Helper class that implements the Traceability Tree Viewer GUI. It completes the glade file specification to add extra features. It also implements the event handlers for the Traceability Tree Viewer screens and dialogs. Manages the tree contents of the viewer, items, edition panels, and management of traceability files.
Responsibilities:

- Definition and inicialization of the Traceability Tree Viewer GUI.
- Management of traceability tree content (items of any kind, traceability files ...).
- Traceability Tree Viewer event handling.

DECLARATION

```
public class TraceabilityTreeView
extends GenericHandler
```

METHODS

• *buildTreeView*

```
public void buildTreeView( )
```

– Usage

- * Initializes the Traceability Tree Viewer engine and builds the main window of the Viewer. Searches automatically for the glade interface file with the application GUI configuration.
-

• *buildTreeView*

```
public void buildTreeView( java.lang.String gladeFilename )
```

– Usage

- * Initializes the Traceability Tree Viewer engine and builds the main window of the Viewer.

– Parameters

- * **gladeFilename** - String, file path of the Traceability Tree Viewer glade file that defines the GUI.
-

• *buildTreeView*

```
public void buildTreeView( java.lang.String gladeFilename, SwtWidget  
parentWidget, Gui parentGui )
```

– Usage

- * Initializes the Traceability Tree Viewer engine and builds the main window of the Viewer.

– Parameters

- * **gladeFilename**, - name of the traceability tree viewer .glade file
 - * **parentWidget** - SwtWidget, parent widget where the Traceability Tree Viewer will be embedded. Null in case a new window is opened.
 - * **parentGui** - Gui, parent Gui where the Traceability tree viewer component will be embedded. Null in case a new window is opened.
-

• *deleteTreeContents*

```
public void deleteTreeContents( )
```

– Usage

- * Delete the graphical traceability tree made of the traceability file. Delete all the items defined in the traceability tree.
-

• *deleteTreeContents*

```
public void deleteTreeContents( Tree t )
```

– Usage

- * Delete the graphical traceability tree made of the traceability file. Delete all the items defined in the traceability tree.

– Parameters

- * **t** - Tree,
-

- *getCompletePath*

```
public URL getCompletePath( java.lang.String path )
```

- *getInputStream*

```
public InputStream getInputStream( java.lang.String path )
```

- *getInsertType*

```
public int getInsertType( )
```

- **Usage**

- * Gets the insert item type chosen by the user.

- **Returns** - int, insert item type.

- **See Also**

- *

- com.imagos.traceabilityTreeView.gui.TraceabilityTreeItem.NODE_UNDEFINED

- *

- com.imagos.traceabilityTreeView.gui.TraceabilityTreeItem.NODE_PRODUCT

- *

- com.imagos.traceabilityTreeView.gui.TraceabilityTreeItem.NODE_PROCESS

- *

- com.imagos.traceabilityTreeView.gui.TraceabilityTreeItem.NODE_WORKER

- *

- com.imagos.traceabilityTreeView.gui.TraceabilityTreeItem.NODE_WORKCENTER

- *

- com.imagos.traceabilityTreeView.gui.TraceabilityTreeItem.NODE_WAREHOUSE

- *

- com.imagos.traceabilityTreeView.gui.TraceabilityTreeItem.NODE_PURCHASE

- *

- com.imagos.traceabilityTreeView.gui.TraceabilityTreeItem.NODE_PROPERTY

- *

- com.imagos.traceabilityTreeView.gui.TraceabilityTreeItem.NODE_APPCC

- *

- com.imagos.traceabilityTreeView.gui.TraceabilityTreeItem.NODE_COMMENT

- *getInstance*

```
public static final TraceabilityTreeView getInstance( )
```

- **Usage**

- * Returns the unique instance of the TraceabilityTreeView class. Implements the singleton design pattern.

- **Returns** - TraceabilityTreeView, unique instance of TraceabilityTreeView.

- *getItems*

```
public Hashtable getItems( )
```

- **Usage**

- * Gets the Traceability tree item's catalog (indexed by widget).

- **Returns** - Hashtable<Widget, TraceabilityTreeItem>, catalog of items.

- *getItemsByName*

```
public Hashtable getItemsByName( )
```

- **Usage**
 - * Gets the Traceability tree item's catalog (indexed by name).
 - **Returns** - Hashtable<String, TraceabilityTreeItem>, catalog of items.
-
- *getRevision*
 public static String **getRevision**()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
-
- *getTraceabilityFilename*
 public String **getTraceabilityFilename**()
 - **Usage**
 - * Gets the opened traceability file path.
 - **Returns** - String, item name.
-
- *isInsertMode*
 public boolean **isInsertMode**()
 - **Usage**
 - * Gets if the insert mode is enabled.
 - **Returns** - boolean, True when the insert mode is enabled; False when the insert mode is disabled.
-
- *on_accept_pressed*
 public void **on_accept_pressed**(Event e)
 - **Usage**
 - * Handler for the accept button pressed event. Updates the tree to keep the changes made to the traceability tree.
 - **Parameters**
 - * e - Event, accept button pressed event data.
-
- *on_buttonProcessDate_pressed*
 public void **on_buttonProcessDate_pressed**(Event e)
 - **Usage**
 - * Handler for the choose process date button pressed event. Shows the calendar to pick up a process date for the process and updates the edit panel with the selected date.
 - **Parameters**
 - * e - Event, Button pressed event data
-
- *on_buttonProductDateExpiry_pressed*
 public void **on_buttonProductDateExpiry_pressed**(Event e)
 - **Usage**
 - * Handler for the choose expiry date button pressed event. Shows the calendar to pick up an expiry date for the product and updates the edit panel with the selected date.

– **Parameters**

* **e** - Event, Button pressed event data

• *on_buttonProductDateIn_pressed*

public void **on_buttonProductDateIn_pressed**(Event **e**)

– **Usage**

* Handler for the choose in date button pressed event. Shows the calendar to pick up an in date for the product and updates the edit panel with the selected date.

– **Parameters**

* **e** - Event, Button pressed event data

• *on_buttonProductTraceabilityFile_pressed*

public void **on_buttonProductTraceabilityFile_pressed**(Event **e**)

– **Usage**

* Handler for the choose traceability file button pressed event. Opens a traceability file and appends the traceability tree to the process node selected.

– **Parameters**

* **e** - Event, Open event data

• *on_cancel_pressed*

public void **on_cancel_pressed**(Event **e**)

– **Usage**

* Handler for the cancel button pressed event. Discards the changes made to the tree item and reloads the previous item data.

– **Parameters**

* **e** - Event, cancel button pressed event data.

• *on_delete_pressed*

public void **on_delete_pressed**(Event **e**)

– **Usage**

* Handler for the Delete button pressed event. Deletes the selected tree item.

– **Parameters**

* **e** - Event, delete button pressed event data.

• *on_dialogChooseDate_close*

public void **on_dialogChooseDate_close**(Event **e**)

– **Usage**

* Handler for the closing of the choose date dialog event. Gets the date selected by the user as the result.

– **Parameters**

* **e** - Event, close choose date dialog event data.

• *on_dialogChooseNodeType_close*

public void **on_dialogChooseNodeType_close**(Event **e**)

– **Usage**

- * Handler for the closing of the choose item type dialog event. Gets the type of item selected by the user as the result.

– **Parameters**

- * **e** - Event, close choose item type dialog event data.
-

- *on_dialogResultMessage_close*

```
public void on_dialogResultMessage_close( Event e )
```

– **Usage**

- * Handler for the closing of the show result messages dialog event. Hides the result message dialog.

– **Parameters**

- * **e** - Event, close result message dialog event data.
-

- *on_editPanel_resized*

```
public void on_editPanel_resized( Event e )
```

- *on_entryProductBarcode_changed*

```
public void on_entryProductBarcode_changed( Event e )
```

– **Usage**

- * Handler for the barcode entry changed event. Updates the Product barcode's image item.

– **Parameters**

- * **e** - Event, barcode entry changed event data.
-

- *on_exit_pressed*

```
public void on_exit_pressed( Event e )
```

– **Usage**

- * Handler for the Exit button pressed event. Closes the application.

– **Parameters**

- * **e** - Event, Exit button pressed event data.
-

- *on_item_selected*

```
public void on_item_selected( Event e )
```

– **Usage**

- * Handler for the item selected event. Loads the edition screen corresponding to the selected item and shows the item information on screen.

– **Parameters**

- * **e** - Event, item selection event data.
-

- *on_newItem_pressed*

```
public void on_newItem_pressed( Event e )
```

– **Usage**

- * Handler for the new item button pressed event or for the new item menuitem selected event. Enables the insert mode to add a new tree item.

– **Parameters**

- * **e** - Event, new item event data.

- *on_newTree_pressed*

```
public void on_newTree_pressed( Event e )
```

- **Usage**

- * Handler for the New tree button pressed event or for the New Tree menuitem selection event. Clears the opened traceability tree and gets ready for starting the construction of a new tree.

- **Parameters**

- * e - Event, New tree event data
-

- *on_open_pressed*

```
public void on_open_pressed( Event e )
```

- **Usage**

- * Handler for the Open button pressed event or for the Open menuitem selected event. Opens a traceability file and builds the traceability tree. Shows the built tree on screen.

- **Parameters**

- * e - Event, Open event data
-

- *on_resultMessageText_modify*

```
public void on_resultMessageText_modify( Event e )
```

- **Usage**

- * Handler for the result message text modify event. Updates the scroll bars needed to see the result messages on screen.

- **Parameters**

- * e - Event, result message text modify event data.
-

- *on_save_pressed*

```
public void on_save_pressed( Event e )
```

- **Usage**

- * Handler for the Save button pressed event or for the Save menuitem selection event. Saves the tree content to the opened traceability file.

- **Parameters**

- * e - Event, Save event data.
-

- *on_saveAs_pressed*

```
public void on_saveAs_pressed( Event e )
```

- **Usage**

- * Handler for the SaveAs menuitem selection event. Selects the file that will hold the tree content, and then saves the tree content to the selected file.

- **Parameters**

- * e - Event, SaveAs menuitem selection event data.
-

- *openTraceabilityFile*

```
public void openTraceabilityFile( byte [] trData )
```

– **Usage**

- * Builds the graphical traceability tree made of the traceability file. Build all the items defined in the traceability tree and show them to the user.

– **Parameters**

- * **trData** - byte[], traceability data of the product. It's the traceability bytes read from the traceability file.
-

- *openTraceabilityFile*

```
public void openTraceabilityFile( java.lang.String  traceabilityFilename )
```

– **Usage**

- * Builds the graphical traceability tree made of the traceability file. Build all the items defined in the traceability tree and show them to the user.

– **Parameters**

- * **traceabilityFilename** - String, path to the traceability file.
-

- *saveTraceabilityTree*

```
public void saveTraceabilityTree( java.io.ByteArrayOutputStream  out )
```

– **Usage**

- * Saves the traceability tree content onto an output stream.

– **Parameters**

- * **out** - ByteArrayOutputStream out, output stream to save the traceability tree contents to.
-

- *setInsertMode*

```
public void setInsertMode( boolean  insertMode )
```

– **Usage**

- * Sets up the insert mode.

– **Parameters**

- * **insertMode** - boolean, True to enable the insert mode; False to disable the insert mode.
-

- *setInsertType*

```
public void setInsertType( int  insertType )
```

– **Usage**

- * Sets up the insert item type chosen by the user.

– **Parameters**

- * **insertType** - int, an item type.

– **See Also**

- *
com.imagos.traceabilityTreeView.gui.TraceabilityTreeItem.NODE_UNDEFINED
- *
com.imagos.traceabilityTreeView.gui.TraceabilityTreeItem.NODE_PRODUCT
- *
com.imagos.traceabilityTreeView.gui.TraceabilityTreeItem.NODE_PROCESS
- *
com.imagos.traceabilityTreeView.gui.TraceabilityTreeItem.NODE_WORKER

```

*
    com.imagos.traceabilityTreeView.gui.TraceabilityTreeItem.NODE_WORKCENTER
*
    com.imagos.traceabilityTreeView.gui.TraceabilityTreeItem.NODE_WAREHOUSE
*
    com.imagos.traceabilityTreeView.gui.TraceabilityTreeItem.NODE_PURCHASE
*
    com.imagos.traceabilityTreeView.gui.TraceabilityTreeItem.NODE_PROPERTY
* com.imagos.traceabilityTreeView.gui.TraceabilityTreeItem.NODE_APPCC
*
    com.imagos.traceabilityTreeView.gui.TraceabilityTreeItem.NODE_COMMENT

```

- *setItems*

```
public void setItems( java.util.Hashtable items )
```

- **Usage**

- * Sets up the Traceability tree item's catalog (indexed by widget).

- **Parameters**

- * *items* - Hashtable<Widget, TraceabilityTreeItem>, a catalog of items.

- *setItemsByName*

```
public void setItemsByName( java.util.Hashtable itemsByName )
```

- **Usage**

- * Sets up the Traceability tree item's catalog (indexed by name).

- **Parameters**

- * *itemsByName* - Hashtable<String, TraceabilityTreeItem>, a catalog of items.

- *setTraceabilityFilename*

```
public void setTraceabilityFilename( java.lang.String traceabilityFilename )
```

- **Usage**

- * Sets up the opened traceability file path.

- **Parameters**

- * *traceabilityFilename* - String, a traceability file path.

METHODS INHERITED FROM CLASS *GenericHandler*

Chapter 2

Package com.imagos.axilice.core.extTables

Package Contents

Page

Classes

ExtFieldBooleanValueOT	25
<i>Store class to hold the data of a Boolean type extension field's value.</i>	
ExtFieldDateValueOT	26
<i>Store class to hold the data of a Date type extension field's value.</i>	
ExtFieldLongValueOT	27
<i>Store class to hold the data of a Long (BLOB) type extension field's value.</i>	
ExtFieldNumberValueOT	29
<i>Store class to hold the data of a String type extension field's value.</i>	
ExtFieldOT	30
<i>Store class to hold the data of an extension field.</i>	
ExtFieldStringValueOT	34
<i>Store class to hold the data of a String type extension field's value.</i>	
ExtFieldValueOT	35
<i>Store class to hold the data of an extension field's value.</i>	
ExtFieldValuePK	36
<i>Store class to hold the primary key of any extension field value (of any type).</i>	
ExtTableDAO	37
<i>This class implements the table extension data access object.</i>	
ExtTableException	43
<i>This class encapsulates and holds all the errors that might have taken place during any of the extension table operations.</i>	
ExtTableLN	46
<i>This class implements the table extension model.</i>	
ExtTableOT	50
<i>Store class to hold the data of the extended table.</i>	

2.1 Classes

2.1.1 CLASS ExtFieldBooleanValueOT

Store class to hold the data of a Boolean type extension field's value. It keeps the value of the extension field defined for an extended item.
Responsibilities:

Store of extension data for exchanging data among presentation, business and data access layers.

DECLARATION

```
public class ExtFieldBooleanValueOT
extends com.imagos.axilice.core.extTables.ExtFieldValueOT
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private Boolean value
 - Boolean value of the extension field for the extended item.

CONSTRUCTORS

- *ExtFieldBooleanValueOT*
public **ExtFieldBooleanValueOT**()

METHODS

- *getRevision*
public static String **getRevision**()
 - Usage
 - * Returns tag Id assigned to CVS source file.
- *getValue*
public Boolean **getValue**()
- *setValue*
public void **setValue**(java.lang.Boolean value)

METHODS INHERITED FROM CLASS com.imagos.axilice.core.extTables.ExtFieldValueOT

(in 2.1.7, page 35)

- *equals*
public boolean **equals**(java.lang.Object obj)
- *getId*
public ExtFieldValuePK **getId**()
- *getIdExtField*
public long **getIdExtField**()

- *getIdValue*
public long getIdValue()
- *getRevision*
public static String getRevision()
 - Usage
 - * Returns tag Id assigned to CVS source file.
- *getVersion*
public Integer getVersion()
- *hashCode*
public int hashCode()
- *setId*
public void setId(com.imagos.axilice.core.extTables.ExtFieldValuePK id)
- *setIdExtField*
public void setIdExtField(long idExtField)
- *setIdValue*
public void setIdValue(long idValue)
- *setVersion*
public void setVersion(java.lang.Integer version)

2.1.2 CLASS ExtFieldDateValueOT

Store class to hold the data of a Date type extension field's value. It keeps the value of the extension field defined for an extended item.
Responsibilities:

- Store of extension data for exchanging data among presentation, business and data access layers.

DECLARATION

```
public class ExtFieldDateValueOT
extends com.imagos.axilice.core.extTables.ExtFieldValueOT
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private Date value
 - Date value of the extension field for the extended item.

CONSTRUCTORS

- *ExtFieldDateValueOT*
public ExtFieldDateValueOT()

METHODS

- *getRevision*
public static String getRevision()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
-
- *getValue*
public Date getValue()
-
- *setValue*
public void setValue(java.sql.Date value)

METHODS INHERITED FROM CLASS com.imagos.axilice.core.extTables.ExtFieldValueOT

(in 2.1.7, page 35)

- *equals*
public boolean equals(java.lang.Object obj)
 - *getId*
public ExtFieldValuePK getId()
 - *getIdExtField*
public long getIdExtField()
 - *getIdValue*
public long getIdValue()
 - *getRevision*
public static String getRevision()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
-
- *getVersion*
public Integer getVersion()
 - *hashCode*
public int hashCode()
 - *setId*
public void setId(com.imagos.axilice.core.extTables.ExtFieldValuePK id)
 - *setIdExtField*
public void setIdExtField(long idExtField)
 - *setIdValue*
public void setIdValue(long idValue)
 - *setVersion*
public void setVersion(java.lang.Integer version)

2.1.3 CLASS ExtFieldLongValueOT

Store class to hold the data of a Long (BLOB) type extension field's value. It keeps the value of the extension field defined for an extended item.
Responsibilities:

Store of extension data for exchanging data among presentation, business and data access layers.

DECLARATION

```
public class ExtFieldLongValueOT
extends com.imagos.axilice.core.extTables.ExtFieldValueOT
implements java.io.Serializable
```

SERIALIZABLE FIELDS

-
- private byte value
 - Long value of the extension field for the extended item.

CONSTRUCTORS

-
- *ExtFieldLongValueOT*
 public **ExtFieldLongValueOT**()

METHODS

-
- *getRevision*
 public static String **getRevision**()
 - **Usage**
 * Returns tag Id assigned to CVS source file.
 - *getValue*
 public byte **getValue**()
 - *setValue*
 public void **setValue**(byte [] value)

METHODS INHERITED FROM CLASS com.imagos.axilice.core.extTables.ExtFieldValueOT

(in 2.1.7, page 35)

- *equals*
 public boolean **equals**(java.lang.Object obj)
 - *getId*
 public ExtFieldValuePK **getId**()
 - *getIdExtField*
 public long **getIdExtField**()
 - *getIdValue*
 public long **getIdValue**()
 - *getRevision*
 public static String **getRevision**()
 - **Usage**
 * Returns tag Id assigned to CVS source file.
-

- *getVersion*
public Integer getVersion()
- *hashCode*
public int hashCode()
- *setId*
public void setId(com.imagos.axilice.core.extTables.ExtFieldValuePK id)
- *setIdExtField*
public void setIdExtField(long idExtField)
- *setIdValue*
public void setIdValue(long idValue)
- *setVersion*
public void setVersion(java.lang.Integer version)

2.1.4 CLASS ExtFieldValueOT

Store class to hold the data of a String type extension field's value. It keeps the value of the extension field defined for an extended item.
Responsibilities:

- Store of extension data for exchanging data among presentation, business and data access layers.

DECLARATION

```
public class ExtFieldValueOT
extends com.imagos.axilice.core.extTables.ExtFieldValueOT
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private BigDecimal value
 - BigDecimal (Numeric) value of the extension field for the extended item.

CONSTRUCTORS

- *ExtFieldValueOT*
public **ExtFieldValueOT**()

METHODS

- *getRevision*
public static String **getRevision**()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
-

- *getValue*
public BigDecimal getValue()
- *setValue*
public void setValue(java.math.BigDecimal value)

METHODS INHERITED FROM CLASS com.imagos.axilice.core.extTables.ExtFieldValueOT

(in 2.1.7, page 35)

- *equals*
public boolean equals(java.lang.Object obj)
- *getId*
public ExtFieldValuePK getId()
- *getIdExtField*
public long getIdExtField()
- *getIdValue*
public long getIdValue()
- *getRevision*
public static String getRevision()
 - Usage
* Returns tag Id assigned to CVS source file.
- *getVersion*
public Integer getVersion()
- *hashCode*
public int hashCode()
- *setId*
public void setId(com.imagos.axilice.core.extTables.ExtFieldValuePK id)
- *setIdExtField*
public void setIdExtField(long idExtField)
- *setIdValue*
public void setIdValue(long idValue)
- *setVersion*
public void setVersion(java.lang.Integer version)

2.1.5 CLASS ExtFieldOT

Store class to hold the data of an extension field. It keeps the metadata and the value of the extension field.
Responsibilities:

Store of extension field data for exchanging data among presentation, business and data access layers.

DECLARATION

```
public class ExtFieldOT
extends java.lang.Object
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idExtField
 - Internal Id of the extension field.
- private String codExtField
 - Code key of the extension field.
- private Integer fieldType
 - Type of the extension field.
- private BigDecimal fieldLength
 - Maximum length (number of characters) of the extension field values.
- private String fieldDescription
 - Description of the extension field.
- private long idValue
 - Value of the id field in the original table (extended table id). It's the id value of the extended item.
- private boolean nullable
 - Show if the field value can be NULL.
- private boolean historizable
 - Show if the field value has to be stored in the historic table.
- private ExtTableOT extTable
 - Extension table the extension field is part of.
- private Object value
 - Value of the extension field for an extended item
- private Integer version
 -

CONSTRUCTORS

- *ExtFieldOT*
public ExtFieldOT()

METHODS

- *equals*
public boolean equals(java.lang.Object other)
- *getCodExtField*
public String getCodExtField()
- *getExtTable*
public ExtTableOT getExtTable()
- *getFieldDescription*
public String getFieldDescription()
- *getFieldLength*
public BigDecimal getFieldLength()
- *getFieldType*
public Integer getFieldType()
- *getFieldTypeAsString*
public String getFieldTypeAsString()
- *getIdExtField*
public long getIdExtField()
- *getIdValue*
public long getIdValue()
- *getRevision*
public static String getRevision()

– Usage

* Returns tag Id assigned to CVS source file.

-
- *getStrVal*
public String getStrVal()
 - *getValue*
public Object getValue()
 - *getValueAsString*
public String getValueAsString()
 - *getVersion*
public Integer getVersion()
 - *hashCode*
public int hashCode()
 - *isHistorizable*
public boolean isHistorizable()
 - *isNullable*
public boolean isNullable()

- *setBooVal*
public void setBooVal(java.lang.Boolean value)
- *setCodExtField*
public void setCodExtField(java.lang.String codExtField)
- *setDatVal*
public void setDatVal(java.sql.Date value)
- *setExtTable*
public void setExtTable(com.imagos.axilice.core.extTables.ExtTableOT extTable)
- *setFieldDescription*
public void setFieldDescription(java.lang.String fieldDescription)
- *setFieldLength*
public void setFieldLength(java.math.BigDecimal fieldLength)
- *setFieldType*
public void setFieldType(java.lang.Integer fieldType)
- *setHistorizable*
public void setHistorizable(boolean historizable)
- *setIdExtField*
public void setIdExtField(java.math.BigDecimal idExtField)
- *setIdExtField*
public void setIdExtField(long idExtField)
- *setIdValue*
public void setIdValue(long idValue)
- *setLonVal*
public void setLonVal(byte [] value)
- *setLonVal*
public void setLonVal(java.lang.String value)
- *setNullable*
public void setNullable(boolean nullable)
- *setNumVal*
public void setNumVal(java.math.BigDecimal value)
- *setStrVal*
public void setStrVal(java.lang.String value)
- *setValue*
public void setValue(java.lang.Object value)
- *setVersion*
public void setVersion(java.lang.Integer version)
- *toString*
public String toString()

- * Implementation of the toString() method to show the data of the ExtFieldOT formatted.

-
- *updateValue*

```
public void updateValue( java.lang.String value )
```

2.1.6 CLASS ExtFieldStringValueOT

Store class to hold the data of a String type extension field's value. It keeps the value of the extension field defined for an extended item.
Responsibilities:

Store of extension data for exchanging data among presentation, business and data access layers.

DECLARATION

```
public class ExtFieldStringValueOT
extends com.imagos.axilice.core.extTables.ExtFieldValueOT
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private String value
 - String value of the extension field for the extended item.

CONSTRUCTORS

- *ExtFieldStringValueOT*

```
public ExtFieldStringValueOT( )
```

METHODS

- *getRevision*

```
public static String getRevision( )
```

 - **Usage**
 - * Returns tag Id assigned to CVS source file.
-
- *getValue*

```
public String getValue( )
```
-
- *setValue*

```
public void setValue( java.lang.String value )
```

METHODS INHERITED FROM CLASS com.imagos.axilice.core.extTables.ExtFieldValueOT

(in 2.1.7, page 35)

- *equals*
public boolean equals(java.lang.Object obj)
- *getId*
public ExtFieldValuePK getId()
- *getIdExtField*
public long getIdExtField()
- *getIdValue*
public long getIdValue()
- *getRevision*
public static String getRevision()
- Usage
* Returns tag Id assigned to CVS source file.
- *getVersion*
public Integer getVersion()
- *hashCode*
public int hashCode()
- *setId*
public void setId(com.imagos.axilice.core.extTables.ExtFieldValuePK id)
- *setIdExtField*
public void setIdExtField(long idExtField)
- *setIdValue*
public void setIdValue(long idValue)
- *setVersion*
public void setVersion(java.lang.Integer version)

2.1.7 CLASS ExtFieldValueOT

Store class to hold the data of an extension field's value. It's an abstract class to define a hierarchy of extension field's values for any type available. All extension field's value classes will implement this superclass.
Responsibilities:

Define the store hierarchy for extension field's values.

DECLARATION

```
public abstract class ExtFieldValueOT
extends java.lang.Object
implements java.io.Serializable
```

CONSTRUCTORS

- *ExtFieldValueOT*
public ExtFieldValueOT()

METHODS

- *equals*
public boolean equals(java.lang.Object obj)
 - *getId*
public ExtFieldValuePK getId()
 - *getIdExtField*
public long getIdExtField()
 - *getIdValue*
public long getIdValue()
 - *getRevision*
public static String **getRevision()**
- Usage
- * Returns tag Id assigned to CVS source file.
-
- *getVersion*
public Integer getVersion()
 - *hashCode*
public int hashCode()
 - *setId*
public void setId(com.imagos.axilice.core.extTables.ExtFieldValuePK id)
 - *setIdExtField*
public void setIdExtField(long idExtField)
 - *setIdValue*
public void setIdValue(long idValue)
 - *setVersion*
public void **setVersion(java.lang.Integer version)**

2.1.8 CLASS ExtFieldValuePK

Store class to hold the primary key of any extension field value (of any type).
Responsibilities:

Store of primary key of an extension field value for implementing the Hibernate mapping.

DECLARATION

```
public class ExtFieldValuePK
extends java.lang.Object
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idExtField
 - Internal Id of the extension field.
- private long idValue
 - Value of the id field in the original table (extended table id). It's the id value of the extended item.

CONSTRUCTORS

- *ExtFieldValuePK*
 public ExtFieldValuePK(long idExtField, long idValue)

METHODS

- *equals*
 public boolean equals(java.lang.Object obj)
- *getIdExtField*
 public long getIdExtField()
- *getIdValue*
 public long getIdValue()
- *getRevision*
 public static String getRevision()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
- *hashCode*
 public int hashCode()
- *setIdExtField*
 public void setIdExtField(long idExtField)
- *setIdValue*
 public void setIdValue(long idValue)

2.1.9 CLASS ExtTableDAO

This class implements the table extension data access object. It's a part of the data access layer and gives access to the database data. It knows the database structure and implementation.
This class is implemented to access a PostgreSQL database implementation.
Responsibilities:

Provide statements for inserting extension tables definitions and values into the database.

Provide statements for updating extension tables definitions and values into the database.

Provide statements for deleting extension tables definitions and values from the database.

Provide the database queries needed for extension tables. Those queries will return the extended table definition and the extension values for an extended item.

DECLARATION

```
public class ExtTableDAO
extends java.lang.Object
```

CONSTRUCTORS

- *ExtTableDAO*
public **ExtTableDAO**()

METHODS

- *deleteAllExtensionFieldValues*
public void **deleteAllExtensionFieldValues**(java.lang.String codExtField)
 - **Usage**
* Delete the extension field values (including all the extension field values for all the extended items).
 - **Parameters**
* codExtField - String, External code of the extension field.
 - **Exceptions**
* *ExtTableException*, - in case any error take place during the query operation.
- *deleteAllExtensionTableValues*
public void **deleteAllExtensionTableValues**(java.lang.String codExtTable)
 - **Usage**
* Delete the extension field values of all the extension fields belonging to the extension table (including all the values for all the extended items).
 - **Parameters**
* codExtTable - String, External code of the extension table.
 - **Exceptions**
* *ExtTableException*, - in case any error take place during the query operation.
- *deleteExtendedFieldValue*
public void **deleteExtendedFieldValue**(
com.imagos.axilice.core.extTables.ExtFieldBooleanValueOT extFieldValue,
Session session, Transaction tx)
 - **Usage**

- * Delete the numeric value of an extension field. This is the version used by a transaction with a lot of delete operations.

– **Parameters**

- * **extFieldValue** - ExtFieldBooleanValueOT, extended field boolean value to delete.
- * **session** - Session, Hibernate session opened for the whole set of delete operations.
- * **tx** - Transaction, Transaction that controls the whole set of delete operations.

– **Exceptions**

- * **ExtTableException**, - in case any error take place during the query operation.

• *deleteExtendedFieldValue*

```
public void deleteExtendedFieldValue(
com.imagos.axilice.core.extTables.ExtFieldDateValueOT extFieldValue,
Session session, Transaction tx )
```

– **Usage**

- * Delete the numeric value of an extension field. This is the version used by a transaction with a lot of delete operations.

– **Parameters**

- * **extFieldValue** - ExtFieldDateValueOT, extended field date value to delete.
- * **session** - Session, Hibernate session opened for the whole set of delete operations.
- * **tx** - Transaction, Transaction that controls the whole set of delete operations.

– **Exceptions**

- * **ExtTableException**, - in case any error take place during the query operation.

• *deleteExtendedFieldValue*

```
public void deleteExtendedFieldValue(
com.imagos.axilice.core.extTables.ExtFieldNumberValueOT extFieldValue,
Session session, Transaction tx )
```

– **Usage**

- * Delete the numeric value of an extension field. This is the version used by a transaction with a lot of delete operations.

– **Parameters**

- * **extFieldValue** - ExtFieldNumberValueOT, extended field numeric value to delete.
- * **session** - Session, Hibernate session opened for the whole set of delete operations.
- * **tx** - Transaction, Transaction that controls the whole set of delete operations.

– **Exceptions**

- * **ExtTableException**, - in case any error take place during the query operation.

• *deleteExtendedFieldValue*

```
public void deleteExtendedFieldValue(
com.imagos.axilice.core.extTables.ExtFieldStringValueOT extFieldValue,
Session session, Transaction tx )
```

– **Usage**

- * Delete the string value of an extension field. This is the version used by a transaction with a lot of delete operations.

– **Parameters**

- * **extFieldValue** - ExtFieldStringValueOT, extended field string value to delete.
- * **session** - Session, Hibernate session opened for the whole set of delete operations.
- * **tx** - Transaction, Transaction that controls the whole set of delete operations.

- **Exceptions**
 - * ExtTableException, - in case any error take place during the query operation.

- *deleteExtendedItemValues*

```
public void deleteExtendedItemValues( java.lang.String  codExtTable, long
idItem )
```

 - **Usage**
 - * Delete the extension field values of an extended item.
 - **Parameters**
 - * codExtTable - String, External code of the extension table.
 - * idItem - long, internal id of the extended item.
 - **Exceptions**
 - * ExtTableException, - in case any error take place during the query operation.

- *deleteExtensionField*

```
public void deleteExtensionField(
com.imagos.axilice.core.extTables.ExtFieldOT  extField )
```

 - **Usage**
 - * Delete the extension field definition and all the values stored for it.
 - **Parameters**
 - * extField - String, ExtFieldOT extension field to delete.
 - **Exceptions**
 - * ExtTableException, - in case any error take place during the delete operation.

- *deleteExtensionTableDefinition*

```
public void deleteExtensionTableDefinition(
com.imagos.axilice.core.extTables.ExtTableOT  extTable )
```

 - **Usage**
 - * Delete the extension table definition and all the fields and values stored for it.
 - **Parameters**
 - * extTable - ExtTableOT, extension table to delete.
 - **Exceptions**
 - * ExtTableException, - in case any error take place during the delete operation.

- *getExtendedItemValues*

```
public Map getExtendedItemValues( java.lang.Long  idValue,
java.lang.String  codExtTable )
```

 - **Usage**
 - * Query the database to get the extension values of an extended item.
 - **Parameters**
 - * idValue - long, Value of the internal id of the extended item. It's the primary key of the extended item.
 - * codExtTable - String, External code of the extension table.
 - **Returns** - Map, set of all the extension fields defined for the extension table containing all the values set for the extended item.
 - **Exceptions**

* **ExtTableException**, - in case any error take place during the query operation.

- *getExtensionField*

```
public ExtFieldOT getExtensionField( java.lang.Long idValue,
java.lang.String codExtField )
```

- **Usage**

- * Query the database to get the extension field (definition and value) of an extended item.

- **Parameters**

- * **idValue** - long, Value of the internal id of the extended item. It's the primary key of the extended item.

- * **codExtField** - String, External code of the extension field.

- **Returns** - ExtFieldOT, definition and value of the extension field queried. If there is no value assigned to the extended item for this field, then return only the extension field definition.

- **Exceptions**

- * **ExtTableException**, - in case any error take place during the query operation.

- *getExtensionTableDefinition*

```
public ExtTableOT getExtensionTableDefinition( long idExtTable )
```

- **Usage**

- * Query (by primary key) the database to get an extension table definition. This definition includes the extended table parameters and the extended fields defined for the extension table that has been queried.

- **Parameters**

- * **idExtTable** - long, Internal id of the extension table.

- **Returns** - ExtTableOT, extension table definition if it exists, null in any other case.

- **Exceptions**

- * **ExtTableException**, - in case any error take place during the query operation.

- *getExtensionTableDefinition*

```
public ExtTableOT getExtensionTableDefinition( java.lang.String
codExtTable )
```

- **Usage**

- * Query (by the extension table code) the database to get an extension table definition. This definition includes the extended table parameters and the extended fields defined for the extension table that has been queried.

- **Parameters**

- * **codExtTable** - String, External code of the extension table.

- **Returns** - ExtTableOT, extension table definition if it exists, null in any other case.

- **Exceptions**

- * **ExtTableException**, - in case any error take place during the query operation.

- *getRevision*

```
public static String getRevision( )
```

- **Usage**

* Returns tag Id assigned to CVS source file.

• *saveExtendedFieldValue*

```
public void saveExtendedFieldValue(
com.imagos.axilice.core.extTables.ExtFieldOT extField )
```

– **Usage**

* Update the value of an extension field.

– **Parameters**

* **extField** - ExtFieldOT, extended field to update.

– **Exceptions**

* **ExtTableException**, - in case any error take place during the query operation.

• *saveExtendedFieldValue*

```
public void saveExtendedFieldValue(
com.imagos.axilice.core.extTables.ExtFieldOT extField, Session session,
Transaction tx )
```

– **Usage**

* Update the value of an extension field. This is the version used by a transaction with a lot of update operations.

– **Parameters**

* **extField** - ExtFieldOT, extended field to update.

* **session** - Session, Hibernate session opened for the whole set of update operations.

* **tx** - Transaction, Transaction that controls the whole set of update operations.

– **Exceptions**

* **ExtTableException**, - in case any error take place during the query operation.

• *saveExtendedItemValues*

```
public void saveExtendedItemValues( java.util.ArrayList lExtFields, long
idItem )
```

– **Usage**

* Update the extension values of an extended item.

– **Parameters**

* **lExtFields** - ArrayList, list of the extension fields that will be updated for the extended item.

* **idItem** - long, internal id of the extended item.

– **Exceptions**

* **ExtTableException**, - in case any error take place during the query operation.

• *saveExtendedItemValues*

```
public void saveExtendedItemValues( java.util.Map mExtFields, long
idItem )
```

– **Usage**

* Update the extension values of an extended item.

– **Parameters**

* **mExtFields** - Map, set of the extension fields that will be updated for the extended item.

* **idItem** - long, internal id of the extended item.

– **Exceptions**

* ExtTableException, - in case any error take place during the query operation.

• *saveExtensionField*

```
public void saveExtensionField( com.imagos.axilice.core.extTables.ExtFieldOT
extField )
```

– **Usage**

* Update or add a new extension field definition.

– **Parameters**

* extField - ExtFieldOT, extended field to update/add.

– **Exceptions**

* ExtTableException, - in case any error take place during the update operation.

• *saveExtensionTableDefinition*

```
public void saveExtensionTableDefinition(
com.imagos.axilice.core.extTables.ExtTableOT extTable )
```

– **Usage**

* Update or add a new extension table definition.

– **Parameters**

* extTable - ExtTableOT, extended table to update/add.

– **Exceptions**

* ExtTableException, - in case any error take place during the update operation.

2.1.10 CLASS ExtTableException

This class encapsulates and holds all the errors that might have taken place during any of the extension table operations.
Responsibilities:

Encapsulate and pass on the error messages to the upper layers.

DECLARATION

```
public class ExtTableException
extends java.lang.Exception
```

SERIALIZABLE FIELDS

• private String codExtField

– Code of the extension field involved in the operation that originated the error.

• private String codExtTable

– Code of the extension table involved in the operation that originated the error.

• private long idItem

- Id of the extended item involved in the operation that originated the error.
- private int `errCode`
 - Internal error code to support error localization and classification.
- private String `errMessage`
 - Error message string that is showed to the end user.

CONSTRUCTORS

- *ExtTableException*
public ExtTableException(int errCode)
- *ExtTableException*
public ExtTableException(int errCode, java.lang.Throwable cause)
- *ExtTableException*
public ExtTableException(java.lang.String codExtTable, int errCode)
- *ExtTableException*
public ExtTableException(java.lang.String codExtTable, int errCode, java.lang.Throwable cause)
- *ExtTableException*
public ExtTableException(java.lang.String codExtTable, java.lang.String codExtField, int errCode)
- *ExtTableException*
public ExtTableException(java.lang.String codExtTable, java.lang.String codExtField, int errCode, java.lang.Throwable cause)
- *ExtTableException*
public ExtTableException(java.lang.String codExtTable, java.lang.String codExtField, long idItem, int errCode)
- *ExtTableException*
public ExtTableException(java.lang.String codExtTable, java.lang.String codExtField, long idItem, int errCode, java.lang.Throwable cause)

METHODS

- *getCodExtField*
public String getCodExtField()
- *getCodExtTable*
public String getCodExtTable()
- *getErrCode*
public int getErrCode()
- *getErrMessage*
public String getErrMessage()

- *getIdItem*
public long getIdItem()
- *getMessage*
public String getMessage()
- *getRevision*
public static String getRevision()
- Usage
 - * Returns tag Id assigned to CVS source file.
- *setCodExtField*
public void setCodExtField(java.lang.String codExtField)
- *setCodExtTable*
public void setCodExtTable(java.lang.String codExtTable)
- *setErrCode*
public void setErrCode(int errCode)
- *setErrMsg*
public void setErrMsg(java.lang.String errMsg)
- *setIdItem*
public void setIdItem(long idItem)
- *toString*
public String toString()

METHODS INHERITED FROM CLASS java.lang.Exception

METHODS INHERITED FROM CLASS java.lang.Throwable

- *fillInStackTrace*
public synchronized native Throwable fillInStackTrace()
- *getCause*
public Throwable getCause()
- *getLocalizedMessage*
public String getLocalizedMessage()
- *getMessage*
public String getMessage()
- *getStackTrace*
public StackTraceElement getStackTrace()
- *initCause*
public synchronized Throwable initCause(java.lang.Throwable arg0)
- *printStackTrace*
public void printStackTrace()
- *printStackTrace*
public void printStackTrace(java.io.PrintStream arg0)

- *printStackTrace*
public void printStackTrace(java.io.PrintWriter arg0)
- *setStackTrace*
public void setStackTrace(java.lang.StackTraceElement [] arg0)
- *toString*
public String toString()

2.1.11 CLASS ExtTableLN

This class implements the table extension model. It's a part of the business logic layer and encapsulates the access to the extended database tables, providing methods to query, update, insert or delete extended elements.
To implement these functionalities it uses a HashMap interface that stores all the values of the original table plus the extended values defined for the group/family the element belongs to. This HashMap interface is the most adaptable way to store an undefined and unlimited amount of data that can be stores into these extensible tables.
Responsibilities:

Provide access to extension fields definition metadata (format, type, length, nullable, etc.) for extended items.

Provide access to extension fields values for extended items.

Query the database by the primary key or by any other query criteria for extended items. These queries should return the whole items result of the query (including original values plus extended values).

Insert new extended elements.

Update extension values for extended items.

Delete extension values for extended items.

DECLARATION

```
public class ExtTableLN
extends java.lang.Object
```

METHODS

- *deleteExtendedItemValues*
public void **deleteExtendedItemValues**(java.lang.String codExtTable, long idItem)
 - **Usage**
 - * Delete the extension field values of an extended item.
 - **Parameters**
 - * **codExtTable** - String, External code of the extension table.

- * idItem - long, internal id of the extended item.
 - **Exceptions**
 - * ExtTableException, - in case any error take place during the query operation.

- *deleteExtensionField*

```
public void deleteExtensionField(
com.imagos.axilice.core.extTables.ExtFieldOT  extField )
```

 - **Usage**
 - * Delete the extension field definition and all the values stored for it.
 - **Parameters**
 - * extField - ExtFieldOT, the extension field to delete.
 - **Exceptions**
 - * ExtTableException, - in case any error take place during the delete operation.

- *deleteExtensionField*

```
public void deleteExtensionField( java.lang.String  codExtField )
```

 - **Usage**
 - * Delete the extension field definition and all the values stored for it.
 - **Parameters**
 - * codExtField - String, External code of the extension field.
 - **Exceptions**
 - * ExtTableException, - in case any error take place during the delete operation.

- *deleteExtensionTableDefinition*

```
public void deleteExtensionTableDefinition(
com.imagos.axilice.core.extTables.ExtTableOT  extTable )
```

 - **Usage**
 - * Delete the extension table definition and all the fields and values stored for it.
 - **Parameters**
 - * extTable - ExtTableOT, extension table to delete.
 - **Exceptions**
 - * ExtTableException, - in case any error take place during the delete operation.

- *deleteExtensionTableDefinition*

```
public void deleteExtensionTableDefinition( java.lang.String  codExtTable )
```

 - **Usage**
 - * Delete the extension table definition and all the fields and values stored for it.
 - **Parameters**
 - * codExtTable - String, external code of the extension table to delete.
 - **Exceptions**
 - * ExtTableException, - in case any error take place during the delete operation.

- *getExtendedItemValues*

```
public Map getExtendedItemValues( java.lang.Long  idValue,
java.lang.String  codExtTable )
```

 - **Usage**

- * Return all the values stored for an extended item. Include all the fields defined for the extension table and their correspondent value (if anyone has been defined).

– **Parameters**

- * `idValue` - Long, internal id of the extended item.
- * `codExtTable` - String, external code of the extension table.

– **Returns** - Map, set of all the extension fields and their values for the extended item.

– **Exceptions**

- * `ExtTableException`, - in case any error take place during the query operation.

• *getExtensionField*

```
public ExtFieldOT getExtensionField( java.lang.Long idValue,
java.lang.String codExtField )
```

– **Usage**

- * Query the database to get the extension field (definition and value) of an extended item.

– **Parameters**

- * `idValue` - long, Value of the internal id of the extended item. It's the primary key of the extended item.
- * `codExtField` - String, External code of the extension field.

– **Returns** - ExtFieldOT, definition and value of the extension field queried.

– **Exceptions**

- * `ExtTableException`, - in case any error take place during the update operation.

• *getExtensionTableDefinition*

```
public ExtTableOT getExtensionTableDefinition( long idExtTable )
```

– **Usage**

- * Return the extension table definition, including the list of all the extension fields defined for the table.

– **Parameters**

- * `idExtTable` - long, internal id of the extension table.

– **Returns** - ExtTableOT, extension table definition.

– **Exceptions**

- * `com.imagos.axilice.core.extTables.ExtTableException` - in case any error take place during the query operation.

• *getExtensionTableDefinition*

```
public ExtTableOT getExtensionTableDefinition( java.lang.String
codExtTable )
```

– **Usage**

- * Return the extension table definition, including the list of all the extension fields defined for the table.

– **Parameters**

- * `codExtTable` - String, external code of the extension table.

– **Returns** - ExtTableOT, extension table definition.

– **Exceptions**

- * `com.imagos.axilice.core.extTables.ExtTableException` - in case any error take place during the query operation.

- *getInstance*

```
public static ExtTableLN getInstance( )
```

- **Usage**

- * Returns the Singleton instance of class ExtTableLN.

- **Returns** - ExtTableLN, the Singleton instance of ExtTableLN.

- *getRevision*

```
public static String getRevision( )
```

- **Usage**

- * Returns tag Id assigned to CVS source file.

- *saveExtendedFieldValue*

```
public void saveExtendedFieldValue(
com.imagos.axilice.core.extTables.ExtFieldOT extField, long idItem )
```

- **Usage**

- * Update the value of an extension field.

- **Parameters**

- * **extField** - ExtFieldOT, extended field to update.
 - * **idItem** - long, internal id of the extended item.

- **Exceptions**

- * **ExtTableException**, - in case any error take place during the update operation.

- *saveExtendedItemValues*

```
public OperationResult saveExtendedItemValues( java.util.ArrayList
lExtFields, long idItem )
```

- **Usage**

- * Update the value of all the extension fields defined for an extended item.

- **Parameters**

- * **lExtFields** - ArrayList, list of all the extension fields whose values must be updated.
 - * **idItem** - long, internal id of the extended item.

- **Returns** - OperationResult, result of the save extended values operation.

- **Exceptions**

- * **ExtTableException**, - in case any error take place during the update operation.

- *saveExtendedItemValues*

```
public void saveExtendedItemValues( java.util.Map mExtFields, long
idItem )
```

- **Usage**

- * Update the value of all the extension fields defined for an extended item.

- **Parameters**

- * **mExtFields** - Map, set of all the extension fields whose values must be updated.
 - * **idItem** - long, internal id of the extended item.

- **Exceptions**

- * **ExtTableException**, - in case any error take place during the update operation.

- *saveExtensionField*

```
public void saveExtensionField( com.imagos.axilice.core.extTables.ExtFieldOT
extField )
```

- **Usage**

- * Update or add a new extension field definition.

- **Parameters**

- * **extField** - ExtFieldOT, extended field to update/add.

- **Exceptions**

- * **ExtTableException**, - in case any error take place during the update operation.

- *saveExtensionTableDefinition*

```
public void saveExtensionTableDefinition(
com.imagos.axilice.core.extTables.ExtTableOT extTable )
```

- **Usage**

- * Update or add a new extension table definition.

- **Parameters**

- * **extTable** - ExtTableOT, extended table to update/add.

- **Exceptions**

- * **ExtTableException**, - in case any error take place during the update operation.

2.1.12 CLASS ExtTableOT

Store class to hold the data of the extended table. It keeps the metadata and the values of the extension fields defined for an extended item.
Responsibilities:

- Store of extension data for exchanging data among presentation, business and data access layers.

DECLARATION

```
public class ExtTableOT
extends java.lang.Object
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private Map fieldValues

- Map to store an undefined number of extension field values.

- private long idExtTable

- Internal Id of the extension table.

- private String codExtTable

- Code key of the extension table.
- private String tableDescription
 - Name of the extension table.
- private String idFieldName
 - Name of the id field in the original table (extended table id).
- private long idItem
 - Value of the id field in the original table (extended table id). It's the id value of the extended item.
- private Integer version
 -

CONSTRUCTORS

- *ExtTableOT*
 public ExtTableOT()

METHODS

- *equals*
 public boolean equals(java.lang.Object other)
- *getCodExtTable*
 public String getCodExtTable()
- *getFieldValues*
 public Map getFieldValues()
- *getIdExtTable*
 public long getIdExtTable()
- *getIdFieldName*
 public String getIdFieldName()
- *getIdItem*
 public long getIdItem()
- *getRevision*
 public static String getRevision()
 - Usage
 - * Returns tag Id assigned to CVS source file.
- *getTableDescription*
 public String getTableDescription()
- *getVersion*
 public Integer getVersion()

- *hashCode*
public int hashCode()
 - *setCodExtTable*
public void setCodExtTable(java.lang.String codExtTable)
 - *setFieldValues*
public void setFieldValues(java.util.Map fieldValues)
 - *setIdExtTable*
public void setIdExtTable(long idExtTable)
 - *setIdFieldName*
public void setIdFieldName(java.lang.String idFieldName)
 - *setIdItem*
public void setIdItem(long idItem)
 - *setTableDescription*
public void setTableDescription(java.lang.String tableDescription)
 - *setVersion*
public void setVersion(java.lang.Integer version)
 - *toString*
public String toString()
- **Usage**
- * Implementation of the toString() method to show the data of the ExtFieldOT formatted.

Chapter 3

Package com.imagos.axilice.core.management

<i>Package Contents</i>	<i>Page</i>
<hr/>	
Interfaces	
ManagementManager <i>...no description...</i>	54
Classes	
DocumentLineOT <i>...no description...</i>	54
DocumentOT <i>...no description...</i>	58
DocumentTypeOT <i>...no description...</i>	65
MovementOT <i>...no description...</i>	68
SerialNumberOT <i>...no description...</i>	72
<hr/>	

3.1 Interfaces

3.1.1 INTERFACE ManagementManager

DECLARATION

```
public interface ManagementManager
implements java.rmi.Remote
```

METHODS

- *deleteMovements*

```
public OperationResult deleteMovements(
    com.imagos.axilice.core.management.DocumentLineOT entityData )
```
- *getProductStock*

```
public ArrayList getProductStock(
    com.imagos.axilice.core.warehouse.ProductOT product,
    com.imagos.axilice.core.warehouse.WarehouseOT warehouse )
```
- *nextSerialNumber*

```
public long nextSerialNumber(
    com.imagos.axilice.core.management.SerialNumberOT serial )
```

3.2 Classes

3.2.1 CLASS DocumentLineOT

DECLARATION

```
public class DocumentLineOT
extends com.imagos.axilice.core.common.BaseVO
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idDocumentLine
—
- private DocumentOT document
—
- private ProductOT product

-
- private String description
-
- private BigDecimal quantity
-
- private BigDecimal price
-
- private BigDecimal costPrice
-
- private BigDecimal discount
-
- private BigDecimal orderQuantity
-
- private BigDecimal vat
-
- private String productReference
-
- private int order
-
- private Integer version
-
- private BigDecimal re
-
- private Set movements
-
- private WarehouseOT warehouse
-

CONSTRUCTORS

- *DocumentLineOT*
public DocumentLineOT()

METHODS

- *equals*
public boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *getCostPrice*
public BigDecimal getCostPrice()
- *getDescription*
public String getDescription()
- *getDiscount*
public BigDecimal getDiscount()
- *getDiscountAmount*
public BigDecimal getDiscountAmount()
- *getDocument*
public DocumentOT getDocument()
- *getEntityId*
public String getEntityId()
- *getGrossAmount*
public BigDecimal getGrossAmount()
- *getIdDocumentLine*
public long getIdDocumentLine()
- *getInternalId*
public long getInternalId()
- *getInternalValue*
public String getInternalValue()
- *getMovements*
public Set getMovements()
- *getNetAmountBeforeTax*
public BigDecimal getNetAmountBeforeTax()
- *getOrder*
public int getOrder()
- *getOrderQuantity*
public BigDecimal getOrderQuantity()
- *getPrice*
public BigDecimal getPrice()
- *getProduct*
public ProductOT getProduct()
- *getProductReference*
public String getProductReference()

- *getQuantity*
public BigDecimal **getQuantity**()
- *getRe*
public BigDecimal **getRe**()
- *getVat*
public BigDecimal **getVat**()
- *getVersion*
public Integer **getVersion**()
- *getWarehouse*
public WarehouseOT **getWarehouse**()
- *hashCode*
public int **hashCode**()
- *setCostPrice*
public void **setCostPrice**(java.math.BigDecimal costPrice)
- *setDescription*
public void **setDescription**(java.lang.String description)
- *setDiscount*
public void **setDiscount**(java.math.BigDecimal discount)
- *setDocument*
public void **setDocument**(com.imagos.axilice.core.management.DocumentOT document)
- *setIdDocumentLine*
public void **setIdDocumentLine**(long idDocumentLine)
- *setMovements*
public void **setMovements**(java.util.Set movements)
- *setOrder*
public void **setOrder**(int order)
- *setOrderQuantity*
public void **setOrderQuantity**(java.math.BigDecimal orderQuantity)
- *setPrice*
public void **setPrice**(java.math.BigDecimal price)
- *setProduct*
public void **setProduct**(com.imagos.axilice.core.warehouse.ProductOT product)
- *setProductReference*
public void **setProductReference**(java.lang.String productReference)
- *setQuantity*
public void **setQuantity**(java.math.BigDecimal quantity)
- *setRe*
public void **setRe**(java.math.BigDecimal re)

- *setVat*
public void setVat(java.math.BigDecimal vat)
- *setVersion*
public void setVersion(java.lang.Integer version)
- *setWarehouse*
public void setWarehouse(com.imagos.axilice.core.warehouse.WarehouseOT warehouse)
- *toString*
public String toString()

METHODS INHERITED FROM CLASS com.imagos.axilice.core.common.BaseVO

(in 31.2.2, page 740)

- *equals*
public abstract boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *format*
public String format(java.lang.String fieldId, java.util.ArrayList lFields)
- *formatValue*
public String formatValue(java.lang.String fieldId)
- *getEntityId*
public abstract String getEntityId()
- *getInternalId*
public abstract long getInternalId()
- *getInternalValue*
public abstract String getInternalValue()
- *getVisibleFieldsCount*
public int getVisibleFieldsCount(java.util.ArrayList lFields)

– Usage

* Calculates the number of visible fields of the entity.

– **Returns** - int, visible fields number; 1 in case no visible fields have been defined.

3.2.2 CLASS DocumentOT

DECLARATION

```
public class DocumentOT
extends com.imagos.axilice.core.common.BaseVO
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idDocument
—
- private DocumentTypeOT documentType
—
- private Date dateDocument
—
- private Integer version
—
- private String serialNumberPrefix
—
- private long serialNumberValue
—
- private PartnerOT partner
—
- private VendorOT vendor
—
- private CurrencyOT currency
—
- private PaymentMethodOT paymentMethod
—
- private Date lastUpdate
—
- private int vatRegime
—
- private BigDecimal promptPaymentDiscount
—
- private BigDecimal fixedDiscount
—
- private String name
—
- private String brandName
—

- private String idNumber
—
- private String address_billing
—
- private String address_shipping
—
- private String postalCode_billing
—
- private String postalCode_shipping
—
- private String city_billing
—
- private String city_shipping
—
- private StateOT state_billing
—
- private StateOT state_shipping
—
- private CountryOT country_billing
—
- private CountryOT country_shipping
—
- private SerialNumberOT serial
—
- private DocumentTotal total
—
- private WarehouseOT warehouse
—
- private Set lines
—

CONSTRUCTORS

- *DocumentOT*
public DocumentOT()

METHODS

- *equals*
public boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *getAddress_billing*
public String getAddress_billing()
- *getAddress_shipping*
public String getAddress_shipping()
- *getBrandName*
public String getBrandName()
- *getCity_billing*
public String getCity_billing()
- *getCity_shipping*
public String getCity_shipping()
- *getCountry_billing*
public CountryOT getCountry_billing()
- *getCountry_shipping*
public CountryOT getCountry_shipping()
- *getCurrency*
public CurrencyOT getCurrency()
- *getDateDocument*
public Date getDateDocument()
- *getDocumentType*
public DocumentTypeOT getDocumentType()
- *getEntityId*
public String getEntityId()
- *getFixedDiscount*
public BigDecimal getFixedDiscount()
- *getIdDocument*
public long getIdDocument()
- *getIdNumber*
public String getIdNumber()
- *getInternalId*
public long getInternalId()
- *getInternalValue*
public String getInternalValue()
- *getLastUpdate*
public Date getLastUpdate()

- *getLines*
public Set **getLines**()
- *getName*
public String **getName**()
- *getPartner*
public PartnerOT **getPartner**()
- *getPaymentMethod*
public PaymentMethodOT **getPaymentMethod**()
- *getPostalCode_billing*
public String **getPostalCode_billing**()
- *getPostalCode_shipping*
public String **getPostalCode_shipping**()
- *getPromptPaymentDiscount*
public BigDecimal **getPromptPaymentDiscount**()
- *getSerial*
public SerialNumberOT **getSerial**()
- *getSerialNumber*
public String **getSerialNumber**()
- *getSerialNumberPrefix*
public String **getSerialNumberPrefix**()
- *getSerialNumberValue*
public long **getSerialNumberValue**()
- *getState_billing*
public StateOT **getState_billing**()
- *getState_shipping*
public StateOT **getState_shipping**()
- *getTotal*
public DocumentTotal **getTotal**()
- *getVatRegime*
public int **getVatRegime**()
- *getVendor*
public VendorOT **getVendor**()
- *getVersion*
public Integer **getVersion**()
- *getWarehouse*
public WarehouseOT **getWarehouse**()
- *hashCode*
public int **hashCode**()
- *nextLineNumber*
public int **nextLineNumber**()

- *setAddress_billing*
public void setAddress_billing(java.lang.String addressBilling)
- *setAddress_shipping*
public void setAddress_shipping(java.lang.String addressShipping)
- *setBrandName*
public void setBrandName(java.lang.String brandName)
- *setCity_billing*
public void setCity_billing(java.lang.String cityBilling)
- *setCity_shipping*
public void setCity_shipping(java.lang.String cityShipping)
- *setCountry_billing*
public void setCountry_billing(com.imagos.axilice.core.common.CountryOT countryBilling)
- *setCountry_shipping*
public void setCountry_shipping(com.imagos.axilice.core.common.CountryOT countryShipping)
- *setCurrency*
public void setCurrency(com.imagos.axilice.core.common.CurrencyOT currency)
- *setDateDocument*
public void setDateDocument(java.util.Date dateDocument)
- *setDocumentType*
public void setDocumentType(com.imagos.axilice.core.management.DocumentTypeOT documentType)
- *setFixedDiscount*
public void setFixedDiscount(java.math.BigDecimal fixedDiscount)
- *setIdDocument*
public void setIdDocument(long idDocument)
- *setIdNumber*
public void setIdNumber(java.lang.String idNumber)
- *setLastUpdate*
public void setLastUpdate(java.util.Date lastUpdate)
- *setLines*
public void setLines(java.util.Set lines)
- *setName*
public void setName(java.lang.String name)
- *setPartner*
public void setPartner(com.imagos.axilice.core.partners.PartnerOT partner)

- *setPaymentMethod*
public void setPaymentMethod(
com.imagos.axilice.core.partners.PaymentMethodOT paymentMethod)
- *setPostalCode_billing*
public void setPostalCode_billing(java.lang.String postalCodeBilling)
- *setPostalCode_shipping*
public void setPostalCode_shipping(java.lang.String postalCodeShipping)
- *setPromptPaymentDiscount*
public void setPromptPaymentDiscount(java.math.BigDecimal
promptPaymentDiscount)
- *setSerial*
public void setSerial(com.imagos.axilice.core.management.SerialNumberOT
serial)
- *setSerialNumberPrefix*
public void setSerialNumberPrefix(java.lang.String serialNumberPrefix)
- *setSerialNumberValue*
public void setSerialNumberValue(long serialNumberValue)
- *setState_billing*
public void setState_billing(com.imagos.axilice.core.common.StateOT
stateBilling)
- *setState_shipping*
public void setState_shipping(com.imagos.axilice.core.common.StateOT
stateShipping)
- *setVatRegime*
public void setVatRegime(int vatRegime)
- *setVendor*
public void setVendor(com.imagos.axilice.core.partners.VendorOT vendor)
- *setVersion*
public void setVersion(java.lang.Integer version)
- *setWarehouse*
public void setWarehouse(com.imagos.axilice.core.warehouse.WarehouseOT
warehouse)
- *toString*
public String toString()

METHODS INHERITED FROM CLASS com.imagos.axilice.core.common.BaseVO

(in 31.2.2, page 740)

- *equals*
public abstract boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)

- *format*
public String format(java.lang.String fieldId, java.util.ArrayList lFields)
- *formatValue*
public String formatValue(java.lang.String fieldId)
- *getEntityId*
public abstract String getEntityId()
- *getInternalId*
public abstract long getInternalId()
- *getInternalValue*
public abstract String getInternalValue()
- *getVisibleFieldsCount*
public int getVisibleFieldsCount(java.util.ArrayList lFields)
 - **Usage**
 - * Calculates the number of visible fields of the entity.
 - **Returns** - int, visible fields number; 1 in case no visible fields have been defined.

3.2.3 CLASS DocumentTypeOT

DECLARATION

```
public class DocumentTypeOT
extends com.imagos.axilice.core.common.BaseVO
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idDocumentType
—
- private String description
—
- private int documentClass
—
- private boolean modifyStock
—
- private boolean returnDocument
—
- private boolean incrementStock
—
- private boolean accountingDocument
—

- private Integer version
—
- private ExtTableOT extTable
—
- private ExtTableOT extTableLines
—
- private String reportName
—
- private String printerName
—
- private Boolean preview
—

CONSTRUCTORS

- *DocumentTypeOT*
public **DocumentTypeOT**()

METHODS

- *equals*
public boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *getAccountingDocument*
public boolean getAccountingDocument()
- *getDescription*
public String getDescription()
- *getDocumentClass*
public int getDocumentClass()
- *getEntityId*
public String getEntityId()
- *getExtTable*
public ExtTableOT getExtTable()
- *getExtTableLines*
public ExtTableOT getExtTableLines()
- *getIdDocumentType*
public long getIdDocumentType()

- *getIncrementStock*
public boolean **getIncrementStock**()
- *getInternalId*
public long **getInternalId**()
- *getInternalValue*
public String **getInternalValue**()
- *getModifyStock*
public boolean **getModifyStock**()
- *getPreview*
public Boolean **getPreview**()
- *getPrinterName*
public String **getPrinterName**()
- *getReportName*
public String **getReportName**()
- *getReturnDocument*
public boolean **getReturnDocument**()
- *getVersion*
public Integer **getVersion**()
- *hashCode*
public int **hashCode**()
- *isAccountingDocument*
public boolean **isAccountingDocument**()
- *isIncrementStock*
public boolean **isIncrementStock**()
- *isModifyStock*
public boolean **isModifyStock**()
- *isReturnDocument*
public boolean **isReturnDocument**()
- *setAccountingDocument*
public void **setAccountingDocument**(boolean accountingDocument)
- *setDescription*
public void **setDescription**(java.lang.String description)
- *setDocumentClass*
public void **setDocumentClass**(int documentClass)
- *setExtTable*
public void **setExtTable**(com.imagos.axilice.core.extTables.ExtTableOT extTable)
- *setExtTableLines*
public void **setExtTableLines**(com.imagos.axilice.core.extTables.ExtTableOT extTableLines)

- *setIdDocumentType*
public void setIdDocumentType(long idDocumentType)
- *setIncrementStock*
public void setIncrementStock(boolean incrementStock)
- *setModifyStock*
public void setModifyStock(boolean stockModifyDocument)
- *setPreview*
public void setPreview(java.lang.Boolean preview)
- *setPrinterName*
public void setPrinterName(java.lang.String printerName)
- *setReportName*
public void setReportName(java.lang.String reportName)
- *setReturnDocument*
public void setReturnDocument(boolean returnDocument)
- *setVersion*
public void setVersion(java.lang.Integer version)
- *toString*
public String toString()

METHODS INHERITED FROM CLASS com.imagos.axilice.core.common.BaseVO

(in 31.2.2, page 740)

- *equals*
public abstract boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *format*
public String format(java.lang.String fieldId, java.util.ArrayList lFields)
- *formatValue*
public String formatValue(java.lang.String fieldId)
- *getEntityId*
public abstract String getEntityId()
- *getInternalId*
public abstract long getInternalId()
- *getInternalValue*
public abstract String getInternalValue()
- *getVisibleFieldsCount*
public int getVisibleFieldsCount(java.util.ArrayList lFields)
 - Usage
 - * Calculates the number of visible fields of the entity.
 - Returns - int, visible fields number; 1 in case no visible fields have been defined.

3.2.4 CLASS MovementOT

DECLARATION

```
public class MovementOT
extends com.imagos.axilice.core.common.BaseVO
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idMovement
—
- private ProductOT product
—
- private WarehouseOT warehouse
—
- private PartnerOT partner
—
- private DocumentLineOT documentLine
—
- private Date date
—
- private BigDecimal costPrice
—
- private BigDecimal salesPrice
—
- private BigDecimal quantity
—
- private Integer version
—
- private Boolean incrementStock
—
- private LotOT lot
—

CONSTRUCTORS

- *MovementOT*
public MovementOT()

METHODS

- *equals*
public boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *getCostPrice*
public BigDecimal getCostPrice()
- *getDate*
public Date getDate()
- *getDocumentLine*
public DocumentLineOT getDocumentLine()
- *getEntityId*
public String getEntityId()
- *getIdMovement*
public long getIdMovement()
- *getIncrementStock*
public Boolean getIncrementStock()
- *getInternalId*
public long getInternalId()
- *getInternalValue*
public String getInternalValue()
- *getLot*
public LotOT getLot()
- *getPartner*
public PartnerOT getPartner()
- *getProduct*
public ProductOT getProduct()
- *getQuantity*
public BigDecimal getQuantity()
- *getSalesPrice*
public BigDecimal getSalesPrice()
- *getVersion*
public Integer getVersion()
- *getWarehouse*
public WarehouseOT getWarehouse()
- *hashCode*
public int hashCode()
- *setCostPrice*
public void setCostPrice(java.math.BigDecimal costPrice)

- *setDate*
public void setDate(java.util.Date date)
- *setDocumentLine*
public void setDocumentLine(com.imagos.axilice.core.management.DocumentLineOT documentLine)
- *setIdMovement*
public void setIdMovement(long idMovement)
- *setIncrementStock*
public void setIncrementStock(java.lang.Boolean incrementStock)
- *setLot*
public void setLot(com.imagos.axilice.core.warehouse.LotOT lot)
- *setPartner*
public void setPartner(com.imagos.axilice.core.partners.PartnerOT partner)
- *setProduct*
public void setProduct(com.imagos.axilice.core.warehouse.ProductOT product)
- *setQuantity*
public void setQuantity(java.math.BigDecimal quantity)
- *setSalesPrice*
public void setSalesPrice(java.math.BigDecimal salesPrice)
- *setVersion*
public void setVersion(java.lang.Integer version)
- *setWarehouse*
public void setWarehouse(com.imagos.axilice.core.warehouse.WarehouseOT warehouse)
- *toString*
public String toString()

METHODS INHERITED FROM CLASS com.imagos.axilice.core.common.BaseVO

(in 31.2.2, page 740)

- *equals*
public abstract boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *format*
public String format(java.lang.String fieldId, java.util.ArrayList lFields)
- *formatValue*
public String formatValue(java.lang.String fieldId)
- *getEntityId*
public abstract String getEntityId()
- *getInternalId*
public abstract long getInternalId()

- *getInternalValue*
`public abstract String getInternalValue()`
- *getVisibleFieldsCount*
`public int getVisibleFieldsCount(java.util.ArrayList IFields)`
 - **Usage**
 * Calculates the number of visible fields of the entity.
 - **Returns** - int, visible fields number; 1 in case no visible fields have been defined.

3.2.5 CLASS SerialNumberOT

DECLARATION

```
public class SerialNumberOT
extends com.imagos.axilice.core.common.BaseVO
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idSerialNumber
 —
- private String description
 —
- private long value
 —
- private String prefix
 —
- private DocumentTypeOT documentType
 —
- private Integer version
 —

CONSTRUCTORS

- *SerialNumberOT*
`public SerialNumberOT()`

METHODS

- *equals*
public boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *getDescription*
public String getDescription()
- *getDocumentType*
public DocumentTypeOT getDocumentType()
- *getEntityId*
public String getEntityId()
- *getIdSerialNumber*
public long getIdSerialNumber()
- *getInternalId*
public long getInternalId()
- *getInternalValue*
public String getInternalValue()
- *getPrefix*
public String getPrefix()
- *getValue*
public long getValue()
- *getVersion*
public Integer getVersion()
- *hashCode*
public int hashCode()
- *setDescription*
public void setDescription(java.lang.String description)
- *setDocumentType*
public void setDocumentType(com.imagos.axilice.core.management.DocumentTypeOT documentType)
- *setIdSerialNumber*
public void setIdSerialNumber(long idSerialNumber)
- *setPrefix*
public void setPrefix(java.lang.String prefix)
- *setValue*
public void setValue(long value)
- *setVersion*
public void setVersion(java.lang.Integer version)

METHODS INHERITED FROM CLASS com.imagos.axilice.core.common.BaseVO

(in 31.2.2, page 740)

- *equals*
public abstract boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *format*
public String format(java.lang.String fieldId, java.util.ArrayList IFields)
- *formatValue*
public String formatValue(java.lang.String fieldId)
- *getEntityId*
public abstract String getEntityId()
- *getInternalId*
public abstract long getInternalId()
- *getInternalValue*
public abstract String getInternalValue()
- *getVisibleFieldsCount*
public int **getVisibleFieldsCount**(java.util.ArrayList IFields)

– Usage

* Calculates the number of visible fields of the entity.

– **Returns** - int, visible fields number; 1 in case no visible fields have been defined.

Chapter 4

Package

com.imagos.axilice.core.traceability.tree.par

Package Contents

Page

Classes

TraceabilityTreeEntityResolver 76

This class implements a SAX Entity Resolver to update the DTD used in the Traceability Tree XML Parsing to the local version.

TraceabilityTreeParser 77

This class implements a SAX Parser to parse an XML traceability file.

4.1 Classes

4.1.1 CLASS *TraceabilityTreeEntityResolver*

This class implements a SAX Entity Resolver to update the DTD used in the Traceability Tree XML Parsing to the local version. That way the application can work even with no Internet connection and the parsing is faster.
Responsibilities:

- Translate the web URL of the DTD to a local URI.

DECLARATION

```
public class TraceabilityTreeEntityResolver
extends java.lang.Object
implements org.xml.sax.EntityResolver
```

CONSTRUCTORS

- *TraceabilityTreeEntityResolver*
public TraceabilityTreeEntityResolver()

METHODS

- *getRevision*
public static String getRevision()
– **Usage**
* Returns tag Id assigned to CVS source file.
- *resolveEntity*
**public InputSource resolveEntity(java.lang.String publicId,
java.lang.String systemId)**
– **Usage**
* Allow the application to resolve external entities. Its the implementation of the resolveEntity method of the EntityResolver interface. In this method the DTD external entity will be redirected to its local version.
– **Parameters**
* **publicId** - String, The public identifier of the external entity being referenced, or null if none was supplied.
* **systemId** - String, The system identifier of the external entity being referenced.
– **Returns** - InputSource An InputSource object describing the new input source, or null to request that the parser open a regular URI connection to the system identifier.

4.1.2 CLASS *TraceabilityTreeParser*

This class implements a SAX Parser to parse an XML traceability file.
Responsibilities:

- Parse XML traceability files.
- Validate XML traceability files.

DECLARATION

```
public class TraceabilityTreeParser
extends org.xml.sax.helpers.DefaultHandler
```

CONSTRUCTORS

- *TraceabilityTreeParser*
public TraceabilityTreeParser(byte [] content)
 - **Usage**
 - * Parser constructor. Creates a new instance of the parser and parses the content specified.
 - **Parameters**
 - * **content** - byte[], content of the traceability (xml) file that will be parsed.

- *TraceabilityTreeParser*
public TraceabilityTreeParser(java.io.InputStream is)
 - **Usage**
 - * Parser constructor. Creates a new instance of the parser and parses the file specified.
 - **Parameters**
 - * **is** - InputStream, input stream of the traceability (xml) file that will be parsed.

- *TraceabilityTreeParser*
public TraceabilityTreeParser(java.lang.String aPathFile)
 - **Usage**
 - * Parser constructor. Creates a new instance of the parser and parses the file specified.
 - **Parameters**
 - * **aPathFile** - String, path of the traceability (xml) file that will be parsed.

METHODS

• *appendContent*

```
public void appendContent( char [] ch, int start, int length )
```

– **Usage**

* Updates the content of a PCDATA XML element with the parsed characters.

– **Parameters**

* **ch** - char[], set of characters that is being parsed.
 * **start** - int, starting index of the first character parsed.
 * **length** - int, number of characters parsed.

• *characters*

```
public void characters( char [] ch, int start, int length )
```

– **Usage**

* Receive notification of character data inside an element.

– **Parameters**

* **ch** - char[], The characters.
 * **start** - int, The start position in the character array.
 * **length** - int, The number of characters to use from the character array.

– **Exceptions**

* **SAXException**, - if there is a problem processing the glade file (such as incorrect file format, format not compliant with the schema, unknown tag appeared, etc.).

• *endElement*

```
public void endElement( java.lang.String uri, java.lang.String  
localName, java.lang.String qName )
```

– **Usage**

* Receive notification of the end of an element.

– **Parameters**

* **uri** - String,
 * **localName** - String, the local name (without prefix), or the empty string if Namespace processing is not being performed.
 * **qName** - String, the qualified name (with prefix), or the empty string if qualified names are not available.

– **Exceptions**

* **SAXException**, - if there is a problem processing the glade file (such as incorrect file format, format not compliant with the schema, unknown tag appeared, etc.).

• *error*

```
public void error( org.xml.sax.SAXParseException e )
```

– **Usage**

* Receive notification of a recoverable parser error.

– **Parameters**

* **e** - SAXParseException, The warning information encoded as an exception.

• *fatalError*

```
public void fatalError( org.xml.sax.SAXParseException e )
```

- **Usage**
 - * Report a fatal XML parsing error.
 - **Parameters**
 - * **e** - SAXParseException, The warning information encoded as an exception.
-
- *getContent*
 public String **getContent**()
 - **Usage**
 - * Gets the PCDATA XML element content.
 - **Returns** - String, parsed content.
-
- *getFilename*
 public String **getFilename**()
 - **Usage**
 - * Gets the parsed traceability file path.
 - **Returns** - String, traceability file path.
-
- *getProcessDepth*
 public int **getProcessDepth**()
 - **Usage**
 - * Gets the number of processes being parsed at the time.
 - **Returns** - int, number of processes being parsed.
-
- *getProductDepth*
 public int **getProductDepth**()
 - **Usage**
 - * Gets the number of products being parsed at the time.
 - **Returns** - int, number of products being parsed.
-
- *getRevision*
 public static String **getRevision**()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
-
- *getSComments*
 public Stack **getSComments**()
 - **Usage**
 - * Gets the stack of comments being parsed at the time.
 - **Returns** - Stack<String>, stack of comments being parsed.
-
- *getSProcesses*
 public Stack **getSProcesses**()
 - **Usage**
 - * Gets the stack of processes being parsed at the time.

- **Returns** - Stack<ProcessInfo>, stack of processes being parsed.
-

- *getSProducts*

public Stack **getSProducts**()

- **Usage**
 - * Gets the stack of products being parsed at the time.
 - **Returns** - Stack<ProductInfo>, stack of products being parsed.
-

- *getTrAppcc*

public String **getTrAppcc**()

- **Usage**
 - * Gets the temporary appcc data
 - **Returns** - String, temporary appcc data.
-

- *getTrProcess*

public ProcessInfo **getTrProcess**()

- **Usage**
 - * Gets the temporary process data
 - **Returns** - ProcessInfo, temporary process data.
-

- *getTrProduct*

public ProductInfo **getTrProduct**()

- **Usage**
 - * Gets the temporary product data
 - **Returns** - ProductInfo, temporary product data.
-

- *getTrProperty*

public PropertyInfo **getTrProperty**()

- **Usage**
 - * Gets the temporary property data.
 - **Returns** - PropertyInfo, temporary property data.
-

- *getTrPurchase*

public PurchaseInfo **getTrPurchase**()

- **Usage**
 - * Gets the temporary purchase data
 - **Returns** - PurchaseInfo, temporary purchase data.
-

- *getTrTree*

public TraceabilityTree **getTrTree**()

- **Usage**
 - * Gets the temporary traceability tree data
 - **Returns** - TraceabilityTree, temporary traceability tree data.
-

- *getTrWarehouse*
public WarehouseInfo getTrWarehouse()
 - **Usage**
 - * Gets the temporary warehouse data
 - **Returns** - WarehouseInfo, temporary warehouse data.

- *getTrWorkcenter*
public WorkcenterInfo getTrWorkcenter()
 - **Usage**
 - * Gets the temporary workcenter data
 - **Returns** - WorkcenterInfo, temporary workcenter data.

- *getTrWorker*
public WorkerInfo getTrWorker()
 - **Usage**
 - * Gets the temporary worker data
 - **Returns** - WorkerInfo, temporary worker data.

- *setContent*
public void setContent(java.lang.String content)
 - **Usage**
 - * Sets up the PCDATA XML element content.
 - **Parameters**
 - * content - String, parsed content.

- *setFilename*
public void setFilename(java.lang.String filename)
 - **Usage**
 - * Sets up parsed XML traceability file path.
 - **Parameters**
 - * filename - String, traceability file path.

- *setProcessDepth*
public void setProcessDepth(int processDepth)
 - **Usage**
 - * Sets up the number of processes being parsed at the time.
 - **Parameters**
 - * processDepth - int, number of processes being parsed.

- *setProductDepth*
public void setProductDepth(int productDepth)
 - **Usage**
 - * Sets up the number of products being parsed at the time.
 - **Parameters**

* **productDepth** - int, number of products being parsed.

- *setSComments*

public void setSComments(java.util.Stack comments)

- **Usage**

- * Sets up the stack of comments being parsed at the time.

- **Parameters**

- * **comments** - Stack<String>, stack of comments being parsed.

- *setSProcesses*

public void setSProcesses(java.util.Stack processes)

- **Usage**

- * Sets up the stack of processes being parsed at the time.

- **Parameters**

- * **processes** - Stack<ProcessInfo>, stack of processes being parsed.

- *setSProducts*

public void setSProducts(java.util.Stack products)

- **Usage**

- * Sets up the stack of products being parsed at the time.

- **Parameters**

- * **products** - Stack<ProductInfo>, stack of products being parsed.

- *setTrAppcc*

public void setTrAppcc(java.lang.String trAppcc)

- **Usage**

- * Sets up temporary appcc data.

- **Parameters**

- * **trAppcc** - String, appcc data.

- *setTrProcess*

public void setTrProcess(com.imagos.axilice.core.traceability.tree.ProcessInfo trProcess)

- **Usage**

- * Sets up temporary process data.

- **Parameters**

- * **trProcess** - ProcessInfo, process data.

- *setTrProduct*

public void setTrProduct(com.imagos.axilice.core.traceability.tree.ProductInfo trProduct)

- **Usage**

- * Sets up temporary product data.

- **Parameters**

- * **trProduct** - ProductInfo, product data.

- *setTrProperty*

```
public void setTrProperty(  
com.imagos.axilice.core.traceability.tree.PropertyInfo  trProperty )
```

- **Usage**

- * Sets up temporary property data.

- **Parameters**

- * trProperty - PropertyInfo, property data.

- *setTrPurchase*

```
public void setTrPurchase(  
com.imagos.axilice.core.traceability.tree.PurchaseInfo  trPurchase )
```

- **Usage**

- * Sets up temporary purchase data.

- **Parameters**

- * trPurchase - PurchaseInfo, purchase data.

- *setTrTree*

```
public void setTrTree(  
com.imagos.axilice.core.traceability.tree.TraceabilityTree  trTree )
```

- **Usage**

- * Sets up temporary traceability tree data.

- **Parameters**

- * trTree - TraceabilityTree, traceability tree data.

- *setTrWarehouse*

```
public void setTrWarehouse(  
com.imagos.axilice.core.traceability.tree.WarehouseInfo  trWarehouse )
```

- **Usage**

- * Sets up temporary warehouse data.

- **Parameters**

- * trWarehouse - WarehouseInfo, warehouse data.

- *setTrWorkcenter*

```
public void setTrWorkcenter(  
com.imagos.axilice.core.traceability.tree.WorkcenterInfo  trWorkcenter )
```

- **Usage**

- * Sets up temporary workcenter data.

- **Parameters**

- * trWorkcenter - WorkcenterInfo, workcenter data.

- *setTrWorker*

```
public void setTrWorker(  
com.imagos.axilice.core.traceability.tree.WorkerInfo  trWorker )
```

- **Usage**

- * Sets up temporary worker data.
 - **Parameters**
 - * `trWorker` - WorkerInfo, worker data.
-
- *startDocument*
`public void startDocument()`
 - **Usage**
 - * Receive notification of the start of the traceability file to parse.
 - **Exceptions**
 - * `SAXException`, - if there is a problem processing the traceability file (such as incorrect file format, format not compliant with the schema, unknown tag appeared, etc.).
-
- *startElement*
`public void startElement(java.lang.String uri, java.lang.String
localName, java.lang.String qName, org.xml.sax.Attributes attributes)`
 - **Usage**
 - * Receive notification of the start of an element.
 - **Parameters**
 - * `uri` - String,
 - * `localName` - String, the local name (without prefix), or the empty string if Namespace processing is not being performed.
 - * `qName` - String, the qualified name (with prefix), or the empty string if qualified names are not available.
 - * `attributes` - Attributes, the specified or defaulted attributes.
 - **Exceptions**
 - * `SAXException`, - if there is a problem processing the traceability file (such as incorrect file format, format not compliant with the schema, unknown tag appeared, etc.).
-
- *warning*
`public void warning(org.xml.sax.SAXParseException e)`
 - **Usage**
 - * Receive notification of a parser warning.
 - **Parameters**
 - * `e` - `SAXParseException`, The warning information encoded as an exception.

METHODS INHERITED FROM CLASS `org.xml.sax.helpers.DefaultHandler`

- *characters*
`public void characters(char [] arg0, int arg1, int arg2)`
- *endDocument*
`public void endDocument()`
- *endElement*
`public void endElement(java.lang.String arg0, java.lang.String arg1,
java.lang.String arg2)`

- *endPrefixMapping*
public void endPrefixMapping(java.lang.String arg0)
- *error*
public void error(org.xml.sax.SAXParseException arg0)
- *fatalError*
public void fatalError(org.xml.sax.SAXParseException arg0)
- *ignorableWhitespace*
public void ignorableWhitespace(char [] arg0, int arg1, int arg2)
- *notationDecl*
public void notationDecl(java.lang.String arg0, java.lang.String arg1, java.lang.String arg2)
- *processingInstruction*
public void processingInstruction(java.lang.String arg0, java.lang.String arg1)
- *resolveEntity*
public InputSource resolveEntity(java.lang.String arg0, java.lang.String arg1)
- *setDocumentLocator*
public void setDocumentLocator(org.xml.sax Locator arg0)
- *skippedEntity*
public void skippedEntity(java.lang.String arg0)
- *startDocument*
public void startDocument()
- *startElement*
public void startElement(java.lang.String arg0, java.lang.String arg1, java.lang.String arg2, org.xml.sax.Attributes arg3)
- *startPrefixMapping*
public void startPrefixMapping(java.lang.String arg0, java.lang.String arg1)
- *unparsedEntityDecl*
public void unparsedEntityDecl(java.lang.String arg0, java.lang.String arg1, java.lang.String arg2, java.lang.String arg3)
- *warning*
public void warning(org.xml.sax.SAXParseException arg0)

Chapter 5

Package com.imagos.axilice.core.partners

<i>Package Contents</i>	<i>Page</i>
<hr/>	
Interfaces	
PartnersManager87	
<i>...no description...</i>	
Classes	
AddressOT87	
<i>Store class to hold the data of an address (client address, supplier address, warehouse address, etc.).</i>	
BankAccountOT91	
<i>...no description...</i>	
ContactOT93	
<i>...no description...</i>	
DepartmentOT96	
<i>...no description...</i>	
DiscountOT98	
<i>...no description...</i>	
PartnerGroupOT101	
<i>...no description...</i>	
PartnerOT102	
<i>...no description...</i>	
PaymentMethodOT108	
<i>...no description...</i>	
RappelIntervalOT111	
<i>...no description...</i>	
RappelTypeOT113	
<i>...no description...</i>	
VendorOT115	
<i>...no description...</i>	

5.1 Interfaces

5.1.1 INTERFACE PartnersManager

DECLARATION

```
public interface PartnersManager
implements java.rmi.Remote
```

METHODS

- *delEntities*

```
public OperationResult delEntities( java.util.ArrayList lSel, java.lang.Class myClass )
```

- *getEntity*

```
public BaseVO getEntity( java.lang.Long id, java.lang.Class myClass )
```

- *queryEntities*

```
public List queryEntities( java.lang.String fieldId,
com.imagos.axilice.core.common.BaseVO value, java.lang.Class myClass )
```

- *queryEntities*

```
public List queryEntities( java.lang.String fieldId, java.lang.String value,
java.lang.Class myClass )
```

- *refreshEntity*

```
public OperationResult refreshEntity( com.imagos.axilice.core.common.BaseVO
selectedEntity )
```

- *saveEntity*

```
public OperationResult saveEntity( com.imagos.axilice.core.common.BaseVO
entity )
```

5.2 Classes

5.2.1 CLASS AddressOT

Store class to hold the data of an address (client address, supplier address, warehouse address, etc.).

Responsibilities:

Store of address data for exchanging data among presentation, business and data access layers.

.

DECLARATION

```
public class AddressOT
extends com.imagos.axilice.core.common.BaseVO
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idAddress
 - Internal id of the address.
- private String description
 -
- private String address
 -
- private String poBox
 -
- private String postalCode
 -
- private String city
 -
- private boolean billingAddress
 -
- private boolean shippingAddress
 -
- private CountryOT country
 -
- private StateOT state
 -
- private Integer version
 - Version of the address used by Hibernate for optimistic concurrency control.

CONSTRUCTORS

- *AddressOT*
public **AddressOT**()

METHODS

- *equals*
public boolean equals(java.lang.Long id)
 - *equals*
public boolean equals(java.lang.Object obj)
 - *getAddress*
public String getAddress()
 - *getCity*
public String getCity()
 - *getCountry*
public CountryOT getCountry()
 - *getDescription*
public String getDescription()
 - *getEntityId*
public String getEntityId()
 - *getIdAddress*
public long getIdAddress()
 - *getInternalId*
public long getInternalId()
 - *getInternalValue*
public String getInternalValue()
 - *getMapImage*
public Image getMapImage()
 - *getPoBox*
public String getPoBox()
 - *getPostalCode*
public String getPostalCode()
 - *getState*
public StateOT getState()
 - *getVersion*
public Integer getVersion()
 - **Usage**
 - * Gets the version of the address used by Hibernate for optimistic concurrency control.
 - **Returns** - Integer, version of the group.
-
- *hashCode*
public int hashCode()
 - *isBillingAddress*
public boolean isBillingAddress()

- *isShippingAddress*
public boolean isShippingAddress()
- *setAddress*
public void setAddress(java.lang.String address)
- *setBillingAddress*
public void setBillingAddress(boolean billingAddress)
- *setCity*
public void setCity(java.lang.String city)
- *setCountry*
public void setCountry(com.imagos.axilice.core.common.CountryOT country)
- *setDescription*
public void setDescription(java.lang.String description)
- *setIdAddress*
public void setIdAddress(long idAddress)
- *setPoBox*
public void setPoBox(java.lang.String poBox)
- *setPostalCode*
public void setPostalCode(java.lang.String postalCode)
- *setShippingAddress*
public void setShippingAddress(boolean shippingAddress)
- *setState*
public void setState(com.imagos.axilice.core.common.StateOT state)
- *setVersion*
public void setVersion(java.lang.Integer version)
 - Usage
 - * Sets up the version of the address used by Hibernate for optimistic concurrency control.
 - Parameters
 - * **version** - Integer, version of the group.
- *toString*
public String toString()

METHODS INHERITED FROM CLASS com.imagos.axilice.core.common.BaseVO

(in 31.2.2, page 740)

- *equals*
public abstract boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *format*
public String format(java.lang.String fieldId, java.util.ArrayList lFields)

- *formatValue*
public String formatValue(java.lang.String fieldId)
- *getEntityId*
public abstract String getEntityId()
- *getInternalId*
public abstract long getInternalId()
- *getInternalValue*
public abstract String getInternalValue()
- *getVisibleFieldsCount*
public int getVisibleFieldsCount(java.util.ArrayList IFields)
 - **Usage**
 - * Calculates the number of visible fields of the entity.
 - **Returns** - int, visible fields number; 1 in case no visible fields have been defined.

5.2.2 CLASS BankAccountOT

DECLARATION

```
public class BankAccountOT
extends com.imagos.axilice.core.common.BaseVO
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idBankAccount
—
- private String description
—
- private long idBank
—
- private long idBranch
—
- private long accountNumber
—
- private long controlNumber
—
- private PartnerOT partner
—
- private Integer version
—

CONSTRUCTORS

- *BankAccountOT*
`public BankAccountOT()`

METHODS

- *equals*
`public boolean equals(java.lang.Long id)`
- *equals*
`public boolean equals(java.lang.Object obj)`
- *getAccountNumber*
`public long getAccountNumber()`
- *getControlNumber*
`public long getControlNumber()`
- *getDescription*
`public String getDescription()`
- *getEntityId*
`public String getEntityId()`
- *getIdBank*
`public long getIdBank()`
- *getIdBankAccount*
`public long getIdBankAccount()`
- *getIdBranch*
`public long getIdBranch()`
- *getInternalId*
`public long getInternalId()`
- *getInternalValue*
`public String getInternalValue()`
- *getPartner*
`public PartnerOT getPartner()`
- *getVersion*
`public Integer getVersion()`
- *hashCode*
`public int hashCode()`
- *setAccountNumber*
`public void setAccountNumber(long accountNumber)`
- *setControlNumber*
`public void setControlNumber(long controlNumber)`

- *setDescription*
public void setDescription(java.lang.String description)
- *setIdBank*
public void setIdBank(long idBank)
- *setIdBankAccount*
public void setIdBankAccount(long idBankAccount)
- *setIdBranch*
public void setIdBranch(long idBranch)
- *setPartner*
public void setPartner(com.imagos.axilice.core.partners.PartnerOT partner)
- *setVersion*
public void setVersion(java.lang.Integer version)
- *toString*
public String toString()

METHODS INHERITED FROM CLASS com.imagos.axilice.core.common.BaseVO

(in 31.2.2, page 740)

- *equals*
public abstract boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *format*
public String format(java.lang.String fieldId, java.util.ArrayList IFields)
- *formatValue*
public String formatValue(java.lang.String fieldId)
- *getEntityId*
public abstract String getEntityId()
- *getInternalId*
public abstract long getInternalId()
- *getInternalValue*
public abstract String getInternalValue()
- *getVisibleFieldsCount*
public int getVisibleFieldsCount(java.util.ArrayList IFields)
 - **Usage**
 - * Calculates the number of visible fields of the entity.
 - **Returns** - int, visible fields number; 1 in case no visible fields have been defined.

5.2.3 CLASS ContactOT

DECLARATION

```
public class ContactOT
extends com.imagos.axilice.core.common.BaseVO
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idContact
—
- private String name
—
- private String position
—
- private String phone
—
- private String fax
—
- private String email
—
- private PartnerOT partner
—
- private Integer version
—

CONSTRUCTORS

- *ContactOT*
public **ContactOT**()

METHODS

- *equals*
public boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)

- *getEmail*
public String getEmail()
- *getEntityId*
public String getEntityId()
- *getFax*
public String getFax()
- *getIdContact*
public long getIdContact()
- *getInternalId*
public long getInternalId()
- *getInternalValue*
public String getInternalValue()
- *getName*
public String getName()
- *getPartner*
public PartnerOT getPartner()
- *getPhone*
public String getPhone()
- *getPosition*
public String getPosition()
- *getVersion*
public Integer getVersion()
- *hashCode*
public int hashCode()
- *setEmail*
public void setEmail(java.lang.String email)
- *setFax*
public void setFax(java.lang.String fax)
- *setIdContact*
public void setIdContact(long idContact)
- *setName*
public void setName(java.lang.String name)
- *setPartner*
public void setPartner(com.imagos.axilice.core.partners.PartnerOT partner)
- *setPhone*
public void setPhone(java.lang.String phone)
- *setPosition*
public void setPosition(java.lang.String position)
- *setVersion*
public void setVersion(java.lang.Integer version)
- *toString*
public String toString()

METHODS INHERITED FROM CLASS com.imagos.axilice.core.common.BaseVO

(in 31.2.2, page 740)

- *equals*
public abstract boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *format*
public String format(java.lang.String fieldId, java.util.ArrayList lFields)
- *formatValue*
public String formatValue(java.lang.String fieldId)
- *getEntityId*
public abstract String getEntityId()
- *getInternalId*
public abstract long getInternalId()
- *getInternalValue*
public abstract String getInternalValue()
- *getVisibleFieldsCount*
public int getVisibleFieldsCount(java.util.ArrayList lFields)
 - **Usage**
 - * Calculates the number of visible fields of the entity.
 - **Returns** - int, visible fields number; 1 in case no visible fields have been defined.

5.2.4 CLASS DepartmentOT

DECLARATION

```
public class DepartmentOT
extends com.imagos.axilice.core.common.BaseVO
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idDepartment
 -
- private String name
 -
- private Integer version
 -

CONSTRUCTORS

- *DepartmentOT*
public DepartmentOT()

METHODS

- *equals*
public boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *getEntityId*
public String getEntityId()
- *getIdDepartment*
public long getIdDepartment()
- *getInternalId*
public long getInternalId()
- *getInternalValue*
public String getInternalValue()
- *getName*
public String getName()
- *getVersion*
public Integer getVersion()
- *hashCode*
public int hashCode()
- *setIdDepartment*
public void setIdDepartment(long idDepartment)
- *setName*
public void setName(java.lang.String name)
- *setVersion*
public void setVersion(java.lang.Integer version)
- *toString*
public String toString()

METHODS INHERITED FROM CLASS com.imagos.axilice.core.common.BaseVO

(in 31.2.2, page 740)

- *equals*
public abstract boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *format*
public String format(java.lang.String fieldId, java.util.ArrayList lFields)
- *formatValue*
public String formatValue(java.lang.String fieldId)
- *getEntityId*
public abstract String getEntityId()

- *getInternalId*
`public abstract long getInternalId()`
- *getInternalValue*
`public abstract String getInternalValue()`
- *getVisibleFieldsCount*
`public int getVisibleFieldsCount(java.util.ArrayList IFields)`
 - **Usage**
 - * Calculates the number of visible fields of the entity.
 - **Returns** - int, visible fields number; 1 in case no visible fields have been defined.

5.2.5 CLASS DiscountOT

DECLARATION

```
public class DiscountOT
extends com.imagos.axilice.core.common.BaseVO
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idDiscount
 -
- private String description
 -
- private double lowerLimit
 -
- private double upperLimit
 -
- private double percentage
 -
- private PartnerGroupOT partnerGroup
 -
- private ProductGroupOT productGroup
 -
- private Integer version
 -

CONSTRUCTORS

- *DiscountOT*
`public DiscountOT()`

METHODS

- *equals*
`public boolean equals(java.lang.Long id)`
- *equals*
`public boolean equals(java.lang.Object obj)`
- *getDescription*
`public String getDescription()`
- *getEntityId*
`public String getEntityId()`
- *getIdDiscount*
`public long getIdDiscount()`
- *getInternalId*
`public long getInternalId()`
- *getInternalValue*
`public String getInternalValue()`
- *getLowerLimit*
`public double getLowerLimit()`
- *getPartnerGroup*
`public PartnerGroupOT getPartnerGroup()`
- *getPartnerGroupName*
`public String getPartnerGroupName()`
- *getPercentage*
`public double getPercentage()`
- *getProductGroup*
`public ProductGroupOT getProductGroup()`
- *getProductGroupName*
`public String getProductGroupName()`
- *getUpperLimit*
`public double getUpperLimit()`
- *getVersion*
`public Integer getVersion()`
- *hashCode*
`public int hashCode()`

- *setDescription*
public void setDescription(java.lang.String description)
- *setIdDiscount*
public void setIdDiscount(long idDiscount)
- *setLowerLimit*
public void setLowerLimit(double lowerLimit)
- *setPartnerGroup*
public void setPartnerGroup(com.imagos.axilice.core.partners.PartnerGroupOT partnerGroup)
- *setPercentage*
public void setPercentage(double percentage)
- *setProductGroup*
public void setProductGroup(com.imagos.axilice.core.warehouse.ProductGroupOT productGroup)
- *setUpperLimit*
public void setUpperLimit(double upperLimit)
- *setVersion*
public void setVersion(java.lang.Integer version)
- *toString*
public String toString()

METHODS INHERITED FROM CLASS com.imagos.axilice.core.common.BaseVO

(in 31.2.2, page 740)

- *equals*
public abstract boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *format*
public String format(java.lang.String fieldId, java.util.ArrayList lFields)
- *formatValue*
public String formatValue(java.lang.String fieldId)
- *getEntityId*
public abstract String getEntityId()
- *getInternalId*
public abstract long getInternalId()
- *getInternalValue*
public abstract String getInternalValue()
- *getVisibleFieldsCount*
public int getVisibleFieldsCount(java.util.ArrayList lFields)

– **Usage**

* Calculates the number of visible fields of the entity.

– **Returns** - int, visible fields number; 1 in case no visible fields have been defined.

5.2.6 CLASS PartnerGroupOT

DECLARATION

```
public class PartnerGroupOT
extends com.imagos.axilice.core.common.BaseVO
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idPartnerGroup
—
- private String description
—
- private Integer version
—
- private ExtTableOT extTable
—

CONSTRUCTORS

- *PartnerGroupOT*
public **PartnerGroupOT**()

METHODS

- *equals*
public boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *getDescription*
public String getDescription()
- *getEntityId*
public String getEntityId()
- *getExtTable*
public ExtTableOT getExtTable()
- *getIdPartnerGroup*
public long getIdPartnerGroup()

- *getInternalId*
public long getInternalId()
- *getInternalValue*
public String getInternalValue()
- *getVersion*
public Integer getVersion()
- *hashCode*
public int hashCode()
- *setDescription*
public void setDescription(java.lang.String description)
- *setExtTable*
public void setExtTable(com.imagos.axilice.core.extTables.ExtTableOT extTable)
- *setIdPartnerGroup*
public void setIdPartnerGroup(long idPartnerGroup)
- *setVersion*
public void setVersion(java.lang.Integer version)
- *toString*
public String toString()

METHODS INHERITED FROM CLASS com.imagos.axilice.core.common.BaseVO

(in 31.2.2, page 740)

- *equals*
public abstract boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *format*
public String format(java.lang.String fieldId, java.util.ArrayList lFields)
- *formatValue*
public String formatValue(java.lang.String fieldId)
- *getEntityId*
public abstract String getEntityId()
- *getInternalId*
public abstract long getInternalId()
- *getInternalValue*
public abstract String getInternalValue()
- *getVisibleFieldsCount*
public int getVisibleFieldsCount(java.util.ArrayList lFields)
 - **Usage**
 - * Calculates the number of visible fields of the entity.
 - **Returns** - int, visible fields number; 1 in case no visible fields have been defined.

5.2.7 CLASS PartnerOT

DECLARATION

```
public class PartnerOT
extends com.imagos.axilice.core.common.BaseVO
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idPartner
—
- private String name
—
- private String brandName
—
- private String idNumber
—
- private String codeEDI
—
- private String mainContact
—
- private String phone1
—
- private String phone2
—
- private String fax
—
- private String email
—
- private Integer version
—
- private Integer vatRegime
—
- private Integer numCopies
—
- private BigDecimal maxRisk

-
- private BigDecimal currentRisk
 -
- private Double promptPaymentDiscount
 -
- private Double fixedDiscount
 -
- private PartnerGroupOT partnerGroup
 - Relations
- private RappelTypeOT rappelType
 -
- private CurrencyOT currency
 -
- private PaymentMethodOT paymentMethod
 -
- private Set addressSet
 -
- private Set bankAccountSet
 -
- private VendorOT vendor
 -
- private Set contactSet
 -
- private PriceListOT priceList
 -

CONSTRUCTORS

- *PartnerOT*
public **PartnerOT**()

METHODS

- *equals*
public boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *getAddressSet*
public Set getAddressSet()
- *getBankAccountSet*
public Set getBankAccountSet()
- *getBillingAddress*
public ArrayList getBillingAddress()
- *getBrandName*
public String getBrandName()
- *getCodeEDI*
public String getCodeEDI()
- *getContactSet*
public Set getContactSet()
- *getCurrency*
public CurrencyOT getCurrency()
- *getCurrentRisk*
public BigDecimal getCurrentRisk()
- *getEmail*
public String getEmail()
- *getEntityId*
public String getEntityId()
- *getFax*
public String getFax()
- *getFixedDiscount*
public Double getFixedDiscount()
- *getIdNumber*
public String getIdNumber()
- *getIdPartner*
public long getIdPartner()
- *getInternalId*
public long getInternalId()
- *getInternalValue*
public String getInternalValue()
- *getMainContact*
public String getMainContact()

- *getMaxRisk*
public BigDecimal getMaxRisk()
- *getName*
public String getName()
- *getNumCopies*
public Integer getNumCopies()
- *getPartnerGroup*
public PartnerGroupOT getPartnerGroup()
- *getPartnerGroupName*
public String getPartnerGroupName()
- *getPaymentMethod*
public PaymentMethodOT getPaymentMethod()
- *getPhone1*
public String getPhone1()
- *getPhone2*
public String getPhone2()
- *getPriceList*
public PriceListOT getPriceList()
- *getPromptPaymentDiscount*
public Double getPromptPaymentDiscount()
- *getRappelType*
public RappelTypeOT getRappelType()
- *getShippingAddress*
public ArrayList getShippingAddress()
- *getVatRegime*
public Integer getVatRegime()
- *getVendor*
public VendorOT getVendor()
- *getVersion*
public Integer getVersion()
- *hashCode*
public int hashCode()
- *setAddressSet*
public void setAddressSet(java.util.Set addressSet)
- *setBankAccountSet*
public void setBankAccountSet(java.util.Set bankAccountSet)
- *setBrandName*
public void setBrandName(java.lang.String brandName)
- *setCodeEDI*
public void setCodeEDI(java.lang.String codeEDI)

- *setContactSet*
public void setContactSet(java.util.Set contactSet)
- *setCurrency*
public void setCurrency(com.imagos.axilice.core.common.CurrencyOT currency)
- *setCurrentRisk*
public void setCurrentRisk(java.math.BigDecimal currentRisk)
- *setEmail*
public void setEmail(java.lang.String email)
- *setFax*
public void setFax(java.lang.String fax)
- *setFixedDiscount*
public void setFixedDiscount(java.lang.Double fixedDiscount)
- *setIdNumber*
public void setIdNumber(java.lang.String idNumber)
- *setIdPartner*
public void setIdPartner(long idPartner)
- *setMainContact*
public void setMainContact(java.lang.String mainContact)
- *setMaxRisk*
public void setMaxRisk(java.math.BigDecimal maxRisk)
- *setName*
public void setName(java.lang.String name)
- *setNumCopies*
public void setNumCopies(java.lang.Integer numCopies)
- *setPartnerGroup*
public void setPartnerGroup(com.imagos.axilice.core.partners.PartnerGroupOT partnerGroup)
- *setPaymentMethod*
public void setPaymentMethod(com.imagos.axilice.core.partners.PaymentMethodOT paymentMethod)
- *setPhone1*
public void setPhone1(java.lang.String phone1)
- *setPhone2*
public void setPhone2(java.lang.String phone2)
- *setPriceList*
public void setPriceList(com.imagos.axilice.core.common.PriceListOT priceList)
- *setPromptPaymentDiscount*
public void setPromptPaymentDiscount(java.lang.Double promptPaymentDiscount)

- *setRappelType*
public void setRappelType(com.imagos.axilice.core.partners.RappelTypeOT
rappelType)
- *setVatRegime*
public void setVatRegime(java.lang.Integer vatRegime)
- *setVendor*
public void setVendor(com.imagos.axilice.core.partners.VendorOT agent)
- *setVersion*
public void setVersion(java.lang.Integer version)
- *toString*
public String toString()

METHODS INHERITED FROM CLASS com.imagos.axilice.core.common.BaseVO

(in 31.2.2, page 740)

- *equals*
public abstract boolean equals(java.lang.Long id)
 - *equals*
public boolean equals(java.lang.Object obj)
 - *format*
public String format(java.lang.String fieldId, java.util.ArrayList lFields)
 - *formatValue*
public String formatValue(java.lang.String fieldId)
 - *getEntityId*
public abstract String getEntityId()
 - *getInternalId*
public abstract long getInternalId()
 - *getInternalValue*
public abstract String getInternalValue()
 - *getVisibleFieldsCount*
public int getVisibleFieldsCount(java.util.ArrayList lFields)
- **Usage**
 - * Calculates the number of visible fields of the entity.
 - **Returns** - int, visible fields number; 1 in case no visible fields have been defined.

5.2.8 CLASS PaymentMethodOT

DECLARATION

```
public class PaymentMethodOT
extends com.imagos.axilice.core.common.BaseVO
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idPaymentMethod
—
- private String description
—
- private boolean cash
—
- private boolean batch
—
- private long daysFirstPayment
—
- private long numberPayments
—
- private long daysBetweenPayments
—
- private Integer version
—

CONSTRUCTORS

- *PaymentMethodOT*
public **PaymentMethodOT**()

METHODS

- *equals*
public boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *getDaysBetweenPayments*
public long getDaysBetweenPayments()
- *getDaysFirstPayment*
public long getDaysFirstPayment()
- *getDescription*
public String getDescription()

- *getEntityId*
public String getEntityId()
- *getIdPaymentMethod*
public long getIdPaymentMethod()
- *getInternalId*
public long getInternalId()
- *getInternalValue*
public String getInternalValue()
- *getNumberPayments*
public long getNumberPayments()
- *getVersion*
public Integer getVersion()
- *hashCode*
public int hashCode()
- *isBatch*
public boolean isBatch()
- *isCash*
public boolean isCash()
- *setBatch*
public void setBatch(boolean batch)
- *setCash*
public void setCash(boolean isCash)
- *setDaysBetweenPayments*
public void setDaysBetweenPayments(long daysBetweenPayments)
- *setDaysFirstPayment*
public void setDaysFirstPayment(long daysFirstPayment)
- *setDescription*
public void setDescription(java.lang.String description)
- *setIdPaymentMethod*
public void setIdPaymentMethod(long idPaymentMethod)
- *setNumberPayments*
public void setNumberPayments(long numberPayments)
- *setVersion*
public void setVersion(java.lang.Integer version)

METHODS INHERITED FROM CLASS `com.imagos.axilice.core.common.BaseVO`

(in 31.2.2, page 740)

- *equals*
`public abstract boolean equals(java.lang.Long id)`
- *equals*
`public boolean equals(java.lang.Object obj)`
- *format*
`public String format(java.lang.String fieldId, java.util.ArrayList lFields)`
- *formatValue*
`public String formatValue(java.lang.String fieldId)`
- *getEntityId*
`public abstract String getEntityId()`
- *getInternalId*
`public abstract long getInternalId()`
- *getInternalValue*
`public abstract String getInternalValue()`
- *getVisibleFieldsCount*
`public int getVisibleFieldsCount(java.util.ArrayList lFields)`
 - **Usage**
 - * Calculates the number of visible fields of the entity.
 - **Returns** - int, visible fields number; 1 in case no visible fields have been defined.

5.2.9 CLASS *RappelIntervalOT*

DECLARATION

```
public class RappelIntervalOT
extends com.imagos.axilice.core.common.BaseVO
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idRappelInterval
–
- private RappelTypeOT rappelType
–
- private BigDecimal lowerLimit
–
- private BigDecimal upperLimit
–
- private double discountRate
–

- private Integer version

CONSTRUCTORS

- *RappelIntervalOT*
public **RappelIntervalOT**()

METHODS

- *equals*
public boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *getDisccountRate*
public double getDisccountRate()
- *getEntityId*
public String getEntityId()
- *getIdRappelInterval*
public long getIdRappelInterval()
- *getInternalId*
public long getInternalId()
- *getInternalValue*
public String getInternalValue()
- *getLowerLimit*
public BigDecimal getLowerLimit()
- *getRappelType*
public RappelTypeOT getRappelType()
- *getRappelTypeName*
public String getRappelTypeName()
- *getUpperLimit*
public BigDecimal getUpperLimit()
- *getVersion*
public Integer getVersion()
- *hashCode*
public int hashCode()
- *setDisccountRate*
public void setDisccountRate(double disccountRate)

- *setIdRappelInterval*
public void setIdRappelInterval(long idRappelInterval)
- *setLowerLimit*
public void setLowerLimit(java.math.BigDecimal lowerLimit)
- *setRappelType*
public void setRappelType(com.imagos.axilice.core.partners.RappelTypeOT rappelType)
- *setUpperLimit*
public void setUpperLimit(java.math.BigDecimal upperLimit)
- *setVersion*
public void setVersion(java.lang.Integer version)
- *toString*
public String toString()

METHODS INHERITED FROM CLASS com.imagos.axilice.core.common.BaseVO

(in 31.2.2, page 740)

- *equals*
public abstract boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *format*
public String format(java.lang.String fieldId, java.util.ArrayList lFields)
- *formatValue*
public String formatValue(java.lang.String fieldId)
- *getEntityId*
public abstract String getEntityId()
- *getInternalId*
public abstract long getInternalId()
- *getInternalValue*
public abstract String getInternalValue()
- *getVisibleFieldsCount*
public int getVisibleFieldsCount(java.util.ArrayList lFields)
 - **Usage**
 - * Calculates the number of visible fields of the entity.
 - **Returns** - int, visible fields number; 1 in case no visible fields have been defined.

5.2.10 CLASS *RappelTypeOT*

DECLARATION

```
public class RappelTypeOT
extends com.imagos.axilice.core.common.BaseVO
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idRappelType
—
- private String description
—
- private Integer version
—
- private Set rappelIntervals
—

CONSTRUCTORS

- *RappelTypeOT*
public **RappelTypeOT**()

METHODS

- *equals*
public boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *getDescription*
public String getDescription()
- *getEntityId*
public String getEntityId()
- *getIdRappelType*
public long getIdRappelType()
- *getInternalId*
public long getInternalId()
- *getInternalValue*
public String getInternalValue()
- *getRappelIntervals*
public Set getRappelIntervals()
- *getVersion*
public Integer getVersion()
- *hashCode*
public int hashCode()

- *setDescription*
public void setDescription(java.lang.String description)
- *setIdRappelType*
public void setIdRappelType(long idRappelType)
- *setRappelIntervals*
public void setRappelIntervals(java.util.Set rappelIntervals)
- *setVersion*
public void setVersion(java.lang.Integer version)
- *toString*
public String toString()

METHODS INHERITED FROM CLASS com.imagos.axilice.core.common.BaseVO

(in 31.2.2, page 740)

- *equals*
public abstract boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *format*
public String format(java.lang.String fieldId, java.util.ArrayList IFields)
- *formatValue*
public String formatValue(java.lang.String fieldId)
- *getEntityId*
public abstract String getEntityId()
- *getInternalId*
public abstract long getInternalId()
- *getInternalValue*
public abstract String getInternalValue()
- *getVisibleFieldsCount*
public int getVisibleFieldsCount(java.util.ArrayList IFields)

– **Usage**

* Calculates the number of visible fields of the entity.

– **Returns** - int, visible fields number; 1 in case no visible fields have been defined.

5.2.11 CLASS VendorOT

DECLARATION

```
public class VendorOT
extends com.imagos.axilice.core.common.BaseVO
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idVendor
—
- private String name
—
- private String lastName
—
- private String idNumber
—
- private Float incomeTax
—
- private Set addressSet
—
- private Integer version
—

CONSTRUCTORS

- *VendorOT*
public **VendorOT**()

METHODS

- *addAddress*
public void **addAddress**(java.util.Collection toAdd)
- *clearAddress*
public void **clearAddress**()
- *equals*
public boolean **equals**(java.lang.Long id)
- *equals*
public boolean **equals**(java.lang.Object obj)
- *getAddressSet*
public Set **getAddressSet**()
- *getEntityId*
public String **getEntityId**()
- *getIdNumber*
public String **getIdNumber**()

- *getIdVendor*
public long getIdVendor()
- *getIncomeTax*
public Float getIncomeTax()
- *getInternalId*
public long getInternalId()
- *getInternalValue*
public String getInternalValue()
- *getLastName*
public String getLastName()
- *getName*
public String getName()
- *getVersion*
public Integer getVersion()
- *hashCode*
public int hashCode()
- *setAddressSet*
public void setAddressSet(java.util.Set address)
- *setIdNumber*
public void setIdNumber(java.lang.String idNumber)
- *setIdVendor*
public void setIdVendor(long idVendor)
- *setIncomeTax*
public void setIncomeTax(java.lang.Float incomeTax)
- *setLastName*
public void setLastName(java.lang.String lastName)
- *setName*
public void setName(java.lang.String name)
- *setVersion*
public void setVersion(java.lang.Integer version)
- *toString*
public String toString()

METHODS INHERITED FROM CLASS com.imagos.axilice.core.common.BaseVO

(in 31.2.2, page 740)

- *equals*
public abstract boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *format*
public String format(java.lang.String fieldId, java.util.ArrayList lFields)

- *formatValue*
public String formatValue(java.lang.String fieldId)
- *getEntityId*
public abstract String getEntityId()
- *getInternalId*
public abstract long getInternalId()
- *getInternalValue*
public abstract String getInternalValue()
- *getVisibleFieldsCount*
public int **getVisibleFieldsCount**(java.util.ArrayList IFields)
 - **Usage**
 - * Calculates the number of visible fields of the entity.
 - **Returns** - int, visible fields number; 1 in case no visible fields have been defined.

Chapter 6

Package

com.imagos.axilice.server.common.dao

Package Contents

Page

Classes

BarcodeTypeDAO	120
<i>This class implements the barcode type data access object.</i>	
CommonDAO	121
<i>...no description...</i>	
VatTypeDAO	122
<i>This class implements the VAT type data access object.</i>	

6.1 Classes

6.1.1 CLASS BarcodeTypeDAO

This class implements the barcode type data access object. It's a part of the data access layer and gives access to the database data. It knows the database structure and implementation.
This class is implemented to access a PostgreSQL database implementation.
Responsibilities:

Provide statements for inserting new barcode types into the database.

Provide statements for updating barcode types into the database.

Provide statements for deleting barcode types from the database.

Provide the database queries needed for barcode types operations.

DECLARATION

```
public class BarcodeTypeDAO
extends java.lang.Object
```

CONSTRUCTORS

- *BarcodeTypeDAO*
public BarcodeTypeDAO()
 - **Usage**
 - * Default constructor of the BarcodeTypeDAO class.

METHODS

- *delBarcodeType*
public OperationResult delBarcodeType(java.lang.Long idBarcodeType)

- *delBarcodeTypes*
public OperationResult delBarcodeTypes(java.util.ArrayList lSelBarcodeTypes)

- *getAllBarcodeTypes*
public List getAllBarcodeTypes()
 - **Usage**
 - * Query the database to get all the available barcode types.
 - **Returns** - ArrayList, list of all the barcode types.
 - **Exceptions**
 - * **Exception**, - in case any error take place during the query operation.

- *getBarcodeType*
`public BarcodeTypeOT getBarcodeType(long idBarcodeType)`
 - **Usage**
 - * Query (by primary key) the database to get a barcode type data.
 - **Parameters**
 - * idBarcodeType - long, Internal id of the barcode type.
 - **Returns** - BarcodeTypeOT, barcode type if it exists, null in any other case.
 - **Exceptions**
 - * AxiliceException, - in case any error take place during the query operation.

- *getBarcodeType*
`public BarcodeTypeOT getBarcodeType(java.lang.String description)`
 - **Usage**
 - * Query the database to get a barcode type data.
 - **Parameters**
 - * description - String, barcode type description.
 - **Returns** - BarcodeTypeOT, barcode type if it exists, null in any other case.
 - **Exceptions**
 - * AxiliceException, - in case any error take place during the query operation.

- *getBarcodeTypes*
`public ArrayList getBarcodeTypes(java.util.ArrayList lCriteria)`
 - **Usage**
 - * Query the database to get a list of barcode types that math the query criteria specified.
 - **Parameters**
 - * lCriteria - ArrayList<CriteriaBean>, list of query criteria.
 - **Returns** - ArrayList<BarcodeTypeOT>, list of barcode types matching the searching criteria.
 - **Exceptions**
 - * AxiliceException, - in case any error take place during the query operation.

- *getRevision*
`public static String getRevision()`
 - **Usage**
 - * Returns tag Id assigned to CVS source file.

- *saveBarcodeType*
`public OperationResult saveBarcodeType(com.imagos.axilice.core.common.BarcodeTypeOT barcodeType)`

6.1.2 CLASS CommonDAO

DECLARATION

```
public class CommonDAO
extends java.lang.Object
```

CONSTRUCTORS

- *CommonDAO*
public **CommonDAO**()

METHODS

-
- *delEntities*
public **OperationResult delEntities**(java.util.ArrayList **ISelEntities**,
java.lang.Class **myClass**)
 - *delEntitiesByCriteria*
public **OperationResult delEntitiesByCriteria**(java.util.ArrayList **ICriteria**,
java.lang.Class **myClass**)
 - *delEntity*
public **OperationResult delEntity**(com.imagos.axilice.core.common.BaseVO **b**)
-
- *getEntities*
public **ArrayList getEntities**(java.util.ArrayList **ICriteria**, java.lang.Class
myClass)
 - *getEntities*
public **ArrayList getEntities**(java.lang.String **hql**)
 - *getEntity*
public **BaseVO getEntity**(long **id**, java.lang.Class **myClass**)
 - *saveEntity*
public **OperationResult saveEntity**(com.imagos.axilice.core.common.BaseVO
entity)

6.1.3 CLASS VatTypeDAO

This class implements the VAT type data access object. It's a part of the data access layer and gives access to the database data. It knows the database structure and implementation.
This class is implemented to access a PostgreSQL database implementation.
Responsibilities:

Provide statements for inserting new VAT types into the database.

Provide statements for updating VAT types into the database.

Provide statements for deleting VAT types from the database.

Provide the database queries needed for VAT types operations.

DECLARATION

```
public class VatTypeDAO
extends java.lang.Object
```

CONSTRUCTORS

- *VatTypeDAO*
 public **VatTypeDAO**()
 - **Usage**
 - * Default constructor of the VatTypeDAO class.

METHODS

- *delVatType*
 public **OperationResult delVatType**(java.lang.Long **idVatType**)
- *delVatTypes*
 public **OperationResult delVatTypes**(java.util.ArrayList **lSelVatTypes**)
- *getAllVatTypes*
 public **List getAllVatTypes**()
 - **Usage**
 - * Query the database to get all the available VAT types.
 - **Returns** - ArrayList, list of all the VAT types.
 - **Exceptions**
 - * **Exception**, - in case any error take place during the query operation.
- *getRevision*
 public **static String getRevision**()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
- *getVatType*
 public **VatTypeOT getVatType**(long **idVatType**)
 - **Usage**
 - * Query (by primary key) the database to get a VAT type data.
 - **Parameters**
 - * **idVatType** - long, Internal id of the VAT type.
 - **Returns** - VatTypeOT, VAT type if it exists, null in any other case.
 - **Exceptions**
 - * **AxiliceException**, - in case any error take place during the query operation.
- *getVatType*
 public **VatTypeOT getVatType**(java.lang.String **description**)

- **Usage**
 - * Query the database to get a VAT type data.
 - **Parameters**
 - * **description** - String, VAT type description.
 - **Returns** - VatTypeOT, VAT type if it exists, null in any other case.
 - **Exceptions**
 - * **AxiliceException**, - in case any error take place during the query operation.
-

- *getVatTypes*

```
public ArrayList getVatTypes( java.util.ArrayList lCriteria )
```

- **Usage**
 - * Query the database to get a list of VAT types that math the query criteria specified.
 - **Parameters**
 - * **lCriteria** - ArrayList<CriteriaBean>, list of query criteria.
 - **Returns** - ArrayList<VatTypeOT>, list of VAT types matching the searching criteria.
 - **Exceptions**
 - * **AxiliceException**, - in case any error take place during the query operation.
-

- *saveVatType*

```
public OperationResult saveVatType(
com.imagos.axilice.core.common.VatTypeOT vatType )
```

Chapter 7

Package com.imagos.axilice.client.warehouse

<i>Package Contents</i>	<i>Page</i>
<hr/>	
Classes	
LotClient	126
<i>...no description...</i>	
MeasureUnitClient	139
<i>...no description...</i>	
ProductClient	152
<i>...no description...</i>	
ProductClientListener	166
<i>...no description...</i>	
ProductGroupClient	167
<i>...no description...</i>	
ProductStockClient	175
<i>...no description...</i>	
StockClient	188
<i>...no description...</i>	
TestClient	201
<i>...no description...</i>	
WarehouseClient	202
<i>...no description...</i>	
<hr/>	

7.1 Classes

7.1.1 CLASS *LotClient*

DECLARATION

```
public class LotClient
extends com.imagos.axilice.client.gui.common.CommonClient
```

FIELDS

- public static final String AXILICE_LOG_EDITLOT_GLADEPATH
– Message for logging the entity glade configuration file path.

CONSTRUCTORS

- *LotClient*
public **LotClient**()

METHODS

- *build_extension_fields_view*
public void **build_extension_fields_view**()
 - *buildSelectionSets*
public void **buildSelectionSets**()
 - *clearEntityData*
public void **clearEntityData**()
 - *fillEntityData*
public void **fillEntityData**()
 - *getEntityData*
public BaseVO **getEntityData**()
 - *getEntityFields*
public ArrayList **getEntityFields**()
 - *getProductName*
public String **getProductName**()
 - *getRevision*
public static String **getRevision**()
– **Usage**
* Returns tag Id assigned to CVS source file.
-

- *getWidgets*
public void **getWidgets**()
- *init_specific_components*
public void **init_specific_components**()
- *on_creationdate_selected*
public void **on_creationdate_selected**(Event e)
- *on_expirationdate_selected*
public void **on_expirationdate_selected**(Event e)
- *on_select_creationdate_clicked*
public void **on_select_creationdate_clicked**(Event e)
- *on_select_expirationdate_clicked*
public void **on_select_expirationdate_clicked**(Event e)
- *setEditionMode*
public void **setEditionMode**(boolean enabled)
- *setWidgetsValues*
public void **setWidgetsValues**(com.imagos.axilice.core.common.BaseVO entity)
- *updateEntityData*
public void **updateEntityData**()
- *validateEntity*
public **OperationResult** **validateEntity**()

METHODS INHERITED FROM CLASS com.imagos.axilice.client.gui.common.CommonClient

(in 17.1.2, page 402)

- *addAxiliceEventListener*
public void **addAxiliceEventListener**(
com.imagos.axilice.client.gui.AxiliceEventListener l)
- *addEntity*
public **ArrayList** **addEntity**()
- *addExistingSubentity*
public void **addExistingSubentity**(com.imagos.axilice.core.installer.SubentityOT
ent)
 - **Usage**
 - * Shows a dialog for selection of a new subentity
 - **Parameters**
 - * ent - SubentityOT. Encapsulate data type of subentity
- *addSubentity*
public void **addSubentity**(com.imagos.axilice.core.common.BaseVO subentity,
com.imagos.axilice.core.common.MultiSelectionSubentityOT subE)
 - **Usage**
 - * Adds a existent entity to a multiselection list
 - **Parameters**
 - * subentity - Subentity to be added

* **subE** - Object that encapsulates information about the subentity

- *addTrazabilityTree*

public void addTrazabilityTree(TabFolder tabFolder)

- *build_extension_fields_view*

public abstract void build_extension_fields_view()

– **Usage**

* Build the view of the extension fields.

It must implement only those clients who require it

- *buildEntity*

public AxiliceEntityView buildEntity()

- *buildEntityDetail*

**public AxiliceEntityView buildEntityDetail(
com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData)**

- *buildEntitySelectionView*

public AxiliceEntitySelectionView buildEntitySelectionView()

– **Usage**

* Build view of selection entities

– **Returns** - AxilicePartnerSelectionView. SelectionView created

- *buildSelectionSets*

public abstract void buildSelectionSets()

– **Usage**

* Build 2 sets of SubentityOT: lSelectionSubentities and lMultiSelectionSubentities

These sets are used to connect the widgets with the handlers for the selection buttons and the multiselection panels

It must implement only those clients who require it

- *buildSubentityMultiSelection*

public void buildSubentityMultiSelection()

- *clearEntityData*

public void clearEntityData()

- *clearParentField*

**public OperationResult clearParentField(java.util.ArrayList selRegistries,
java.lang.String parentField)**

– **Usage**

* Clears the value of the field indicated

– **Parameters**

* **selRegistries** - ArrayList. Index list of records to process

* **parentField** - Name of the field to clear

– **Returns** - OperationResult. Result of operation

- *delEntity*

public OperationResult delEntity()

- *delEntity*

public OperationResult delEntity(java.util.ArrayList selRegistries)

- *delEntity*

public OperationResult delEntity(com.imagos.axilice.core.common.BaseVO b)

– **Usage**

- * Delete a entity from database
 - **Parameters**
 - * **b** - BaseVO. Entity to delete
 - **Returns** - OperationResult. Result of the operation

- *delSubentities*
public OperationResult delSubentities(java.util.ArrayList selRegistries)
 - **Usage**
 - * Delete a set of subentities from record list of controller, not from database
 - Set the parent field of subentity to null
 - **Parameters**
 - * **selRegistries** - List of index to delete
 - **Returns** - OperationResult. Result of the operation

- *editEntity*
public OperationResult editEntity()
- *fillEntityData*
public void fillEntityData()
- *fillEntityExtensionTable*
public void fillEntityExtensionTable(java.lang.Long id)
 - **Usage**
 - * Retrieve extension field values from the server for a given entity and displays them in the view
 - **Parameters**
 - * **id** - Long. Key of the entity whose values are to be displayed

- *fillEntityTable*
public void fillEntityTable()
 - **Usage**
 - * Load records from database and fill the entity view table and controller record list

- *fillEntityTable*
public void fillEntityTable(java.util.ArrayList lR)
 - **Usage**
 - * Fill the entity view table with the records given
 - **Parameters**
 - * **lR** - ArrayList. Records to be displayed

- *getClassEntity*
public Class getClassEntity()
 - **Returns** - Class of the entity for which the controller is built

- *getDialogName*
public String getDialogName()
- *getEditGladePath*
public String getEditGladePath()
- *getEditionGui*
public Gui getEditionGui()
 - **Usage**
 - * Returns the edition Gui instance.

- **Returns** - Gui, entity edition Gui.
- - *getEditWindow*
public SwtTopLevelWindow **getEditWindow**()
 - *getEntityData*
public abstract BaseVO **getEntityData**()
 - *getEntityFields*
public ArrayList **getEntityFields**()
 - *getLogger*
public Logger **getLogger**()
 - *getParentEntityLauncher*
public EntityLauncher **getParentEntityLauncher**()
 - **Returns** - EntityLauncher of parent entity if is a embedded entity or null otherwise
- - *getRegistries*
public Collection **getRegistries**()
 - **Returns** - lRegistries
- - *getRegistries*
public List **getRegistries**(java.util.ArrayList lIds)
 - **Usage**
 - * Get a list of entities stored by the controller
 - **Parameters**
 - * lIds - Index list
 - **Returns** - List of entities
- - *getView*
public AxiliceEntityView **getView**()
 - **Returns** - AxiliceEntityView. Main view or Detail view if is a embedded entity.
- - *getWidgets*
public abstract void **getWidgets**()
 - **Usage**
 - * Get the widgets from the glade and stored in local variables.
 - Must be overwritten
- - *init_specific_components*
public abstract void **init_specific_components**()
 - **Usage**
 - * Initializes the specific components of the client
- - *printEntity*
public OperationResult **printEntity**()
- - *printEntity*
public OperationResult **printEntity**(java.lang.String reportPath, java.util.HashMap parameters)
- - *queryEntity*
public ArrayList **queryEntity**()
- - *queryEntity*
public ArrayList **queryEntity**(java.lang.String fieldId, java.lang.String value)

- *queryEntityWithFormat*
public ArrayList queryEntityWithFormat(java.lang.String fieldId, java.lang.String value)
- *queryFreeEntities*
public ArrayList queryFreeEntities(com.imagos.axilice.core.common.MultiSelectionSubentityOT mSubE)
 - **Usage**
 - * Search entities not associated with a parent entity (null parent field)
 - **Parameters**
 - * mSubE - SubentityOT for the parent entity that must be null
 - **Returns** - ArrayList. List of entities found
- *querySubentities*
public ArrayList querySubentities(com.imagos.axilice.core.installer.SubentityOT entData)
- *refreshEntity*
public OperationResult refreshEntity()
- *refreshEntity*
public BaseVO refreshEntity(com.imagos.axilice.core.common.BaseVO entity)
- *refreshParentSelectedEntity*
public OperationResult refreshParentSelectedEntity()
 - **Usage**
 - * Reload from database the value of the parent entity of the selected entity
 - **Returns** - OperationResult. The result of the operation.
- *removeRegistries*
public OperationResult removeRegistries(java.util.ArrayList selRegistries, java.lang.String parentField)
 - **Usage**
 - * Deletes a list of the records stored by the controller. Also clean your parent fields and updates the database
 - **Parameters**
 - * selRegistries - Index list to remove
 - * parentField - Field name that links to the parent entity
 - **Returns** - OperationResult. Result of the operation
- *saveEntity*
public OperationResult saveEntity()
- *saveEntity*
public OperationResult saveEntity(java.util.ArrayList lIds)
- *saveEntity*
public OperationResult saveEntity(com.imagos.axilice.core.common.BaseVO b)
 - **Usage**
 - * Save or update a entity to database
 - **Parameters**
 - * b - BaseVO. Entity to save or update
 - **Returns** - OperationResult. Result of the operation
- *selectEntity*
public void selectEntity(java.lang.Long idEntity)

- *selectSubentity*
 public void **selectSubentity**(java.lang.String parentFieldId,
 com.imagos.axilice.core.common.BaseVO subentityOT)

- *selectTreeItem*
 public void **selectTreeItem**(java.lang.Long id)
 – **Usage**
 * Select a item in the entity tree. No me gusta esto aqui
 – **Parameters**
 * id - Long. Index of item to select

- *setDialogName*
 public void **setDialogName**(java.lang.String dialogName)

- *setEditGladePath*
 public void **setEditGladePath**(java.lang.String editGladePath)

- *setEditionMode*
 public abstract void **setEditionMode**(boolean enabled)

- *setRegistries*
 public void **setRegistries**(java.util.Collection collection)
 – **Usage**
 * Setter for IRegistries

- *setWidgetsValues*
 public abstract void **setWidgetsValues**(com.imagos.axilice.core.common.BaseVO
 entity)
 – **Usage**
 * Set the values of the widgets on the edit dialog.

 Must be overwritten.
 – **Parameters**
 * entity - BaseVO. Entity that contains the values to be displayed

- *showErrorMessages*
 public void **showErrorMessages**(com.imagos.axilice.core.util.OperationResult result
)

- *updateEntityData*
 public void **updateEntityData**()

- *updateRegistry*
 public void **updateRegistry**(com.imagos.axilice.core.common.BaseVO ent)
 – **Usage**
 * Add or update an entity (to the controller records)
 – **Parameters**
 * ent - Entity to add or update

- *updateViews*
 public void **updateViews**(boolean updateFromParent)
 – **Usage**
 * Force the update of the views associated
 – **Parameters**
 * updateFromParent - Indicates whether the view must be updated by reading the values
 from the parent entity. takes effect only when the entity is detail (is embedded in another)

- *validateEntity*
 public abstract **OperationResult validateEntity()**
 - **Usage**
 - * Validate the values of the widgets in the edit dialog.
 - **Must be overwritten.**
 - **Returns** - **OperationResult**. Result of the validation.
-
- *viewEntity*
 public **OperationResult viewEntity()**

METHODS INHERITED FROM CLASS com.imagos.axilice.core.installer.EntityLauncher

(in 24.2.2, page 532)

- *addEntity*
 public abstract **ArrayList addEntity()**
 - **Usage**
 - * Add entity button clicked handler. Show the entity content screen ready to start inserting data for a new instance of the entity (new registry).
 - **Returns** - **ArrayList<ResultMessage>**, list of errors if any has taken place, null in any other case.
-
- *buildEditEntityGUI*
 public void **buildEditEntityGUI()**
 - **Usage**
 - * Builds the edit entity dialog window from the corresponding glade interface file. Stores the entity dialog instance for posterior accesses.
-
- *buildEntity*
 public **AxiliceEntityView buildEntity()**
 - **Usage**
 - * Builds the entity GUI and show it on screen to the user.
-
- *buildEntity*
 public **AxiliceDetailEntityView buildEntity(SwtWidget parent, com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData, java.lang.String rootName)**
 - **Usage**
 - * Builds the entity GUI and show it on screen to the user. The entity GUI will be embedded into the parent composite widget.
 - **Parameters**
 - * **parent** - Composite, parent holder for embedding the entity view.
 - * **subentityData** - SubentityOT, data of the subentity to build if the entity launcher will be used to handle a multi-selection detail entity view.
 - * **rootName** - String, name of the root widget that will be embedded into the master entity edition panel.
 - **Returns** - **AxiliceDetailEntityView**, the entity detail view built.
-
- *cancelEntityEdition*
 public void **cancelEntityEdition()**
 - *clearEntityData*
 public abstract void **clearEntityData()**

- **Usage**
 - * Clears the entity content panel by resetting the content widgets.

- *delEntity*

```
public abstract OperationResult delEntity( )
```

 - **Usage**
 - * Delete entity button clicked handler. Deletes the selected instance of the entity (selected registry) from the database.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *delEntity*

```
public abstract OperationResult delEntity( java.util.ArrayList lSelRegistries )
```

 - **Usage**
 - * Delete entity button clicked handler. Deletes the selected instances of the entity (selected registries) from the database. Its a multiple delete method.
 - **Parameters**
 - * lSelRegistries - ArrayList<Long>, list of selected instances of the entity (selected registries). The list will contain only the internal id of the selected registries.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *deselectSubentity*

```
public void deselectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

 - **Usage**
 - * Updates an entity instance by removing the selected value for a subentity field.
 - **Parameters**
 - * parentFieldId - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * subentityOT - BaseVO, subentity's deselected instance (registry) value.

- *editEntity*

```
public abstract OperationResult editEntity( )
```

 - **Usage**
 - * Edit entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), and allows the user to modify any content of the entity registry.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *fillEntityData*

```
public abstract void fillEntityData( )
```

 - **Usage**
 - * Fills the entity content panel with the data of the selected instance (registry) of the entity.

- *fireExceptionAndExit*

```
public void fireExceptionAndExit( java.lang.String msg )
```

 - **Usage**
 - * Fire a exception and stop the program
 - **Parameters**
 - * msg -

- *formatRegistries*
public ArrayList formatRegistries()
- *formatRegistries*
public ArrayList formatRegistries(java.util.ArrayList lRegistries)
- *getDialog*
public SwtDialog getDialog(java.lang.String dialogName)
- *getEntity*
public EntityOT getEntity()
 - **Usage**
 - * Gets the entity data.
 - **Returns** - EntityOT, entity data.

- *getEntityData*
public abstract BaseVO getEntityData()
 - **Usage**
 - * Gets the content panel's data typed by the user formatted as an entity VO object.

- *getEntityEditViewName*
public String getEntityEditViewName(java.lang.String parentGuiName)
- *getEntityFields*
public abstract ArrayList getEntityFields()
 - **Usage**
 - * Gets the list of defined fields for an entity
 - **Returns** - ArrayList, list of entity fields.

- *getEntityViewName*
public String getEntityViewName(java.lang.String parentGuiName)
- *getLRegistries*
public ArrayList getLRegistries()
 - **Usage**
 - * Getter for lRegistries

- *getRegistry*
public BaseVO getRegistry(long id)
- *getRevision*
public static String getRevision()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.

- *getSelectedEntity*
public BaseVO getSelectedEntity()
 - **Usage**
 - * Gets the selected instance (registry) of the entity.
 - **Returns** - BaseVO, selected instance (registry) of the entity.

- *getSelectedId*
public Long getSelectedId()
 - **Usage**
 - * Gets the internal id of the selected entity registry.
 - **Returns** - Long, internal id of the selected entity registry, null in case no entity registry has been selected yet.

- *getSwtWidget*
public SwtWidget getSwtWidget(java.lang.String widgetName)
- *getWidget*
public Widget getWidget(java.lang.String widgetName)
- *initEntityLauncher*
public void initEntityLauncher(Logger logger, Gui parentGui, SwtWidget parentPanel)
 - **Usage**
 - * Initiates the entity launcher context and prepares the launcher to be started.
 - **Parameters**
 - * **logger** - Logger, logger instance for printing logging messages.
 - * **parentGuiName** - String, name of the JavaGlade instance that built the parent where the entity will be embedded.
 - * **parentPanel** - Composite, parent panel where the entity contents will be embedded.
- *isDetail*
public boolean isDetail()
 - **Usage**
 - * Indicates whether is a main entity or is embedded in another
- *on_entitySelectionButton_clicked*
public void on_entitySelectionButton_clicked(Event e)
 - **Usage**
 - * Handler for the subentity selection buttons (magnifying glass buttons) clicked. Shows the selection dialog for selecting a valid instance of a subentity.
 - **Parameters**
 - * **e** - Event, magnifying glass button clicked event data.
- *printEntity*
public abstract OperationResult printEntity()
 - **Usage**
 - * Print entity button clicked handler. Builds the report with the data of the selected instance of the entity (selected registry), and allows the user to print the elaborated report.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.
- *printWidgets*
public void printWidgets()
- *queryEntity*
public abstract ArrayList queryEntity()
 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) available.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.
- *queryEntity*
public abstract ArrayList queryEntity(java.lang.String fieldId, java.lang.String value)
 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria.
 - **Parameters**

- * **fieldId** - String, name of the entity field that will be queried.
 - * **value** - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.

- *queryEntityWithFormat*

```
public abstract ArrayList queryEntityWithFormat( java.lang.String fieldId,
java.lang.String value )
```

 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria and show them (formatted as a list) to the user in the entity's screen.
 - **Parameters**
 - * **fieldId** - String, name of the entity field that will be queried.
 - * **value** - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList<RegistryOT>, list of entity instances (registries) taken as the result of the query operation.

- *querySubentities*

```
public abstract ArrayList querySubentities(
com.imagos.axilice.core.installer.SubentityOT entData )
```

 - **Usage**
 - * Queries the selected entity for all the detail subentities associated with it.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to query for.
 - **Returns** - ArrayList, list of detail subentity registries assigned to the master detail registry.

- *saveEntity*

```
public abstract OperationResult saveEntity( )
```

 - **Usage**
 - * Save entity button clicked handler. Update the system with the data of the edited entity registry. It's called for saving new entity registries (add a new registry) or for updating the contents of existing entity registries.
 - **Returns** - OperationResult, Result of the operation and list of messages generated through the operation.

- *saveEntityRegistry*

```
public void saveEntityRegistry( )
```

- *selectEntity*

```
public abstract void selectEntity( java.lang.Long idEntity )
```

 - **Usage**
 - * Marks an instance (registry) of the entity as the selected.
 - **Parameters**
 - * **idEntity** - Long, internal id of the selected instance (registry).

- *selectSubentity*

```
public void selectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

 - **Usage**
 - * Updates an entity instance by setting up the selected value for a subentity field.
 - **Parameters**
 - * **parentFieldId** - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.

-
- * subentityOT - BaseVO, subentity's selected instance (registry) value.
-
- *setEntity*
 public void **setEntity**(com.imagos.axilice.core.installer.EntityOT entity)
 – **Usage**
 * Sets up the entity name.
 – **Parameters**
 * name - String, entity name.
-
- *setRegistries*
 public void **setRegistries**(java.util.ArrayList lRegistries)
 – **Usage**
 * Sets the list of registries that are showed in the entity view controlled by the Entity Launcher instance. It's used for detail entities where queries are made in the master entity launcher.
 – **Parameters**
 * lRegistries - ArrayList<BaseVO>, list of shown registries of the entity.
-
- *showSelectionDialog*
 public void **showSelectionDialog**(com.imagos.axilice.core.installer.SubentityOT entData)
 – **Usage**
 * Opens the selection dialog that enables the user to select an instance of a subentity.
 – **Parameters**
 * entData - SubentityOT, data of the detail subentity to select.
-
- *showSubentityDialog*
 public void **showSubentityDialog**(com.imagos.axilice.core.installer.SubentityOT entData, long selEntity, boolean readonly)
 – **Usage**
 * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.
 – **Parameters**
 * entData - SubentityOT, data of the detail subentity to select.
 * selEntity - long, id of the selected instance of the subentity.
 * readonly - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.
-
- *showSubentityDialog*
 public void **showSubentityDialog**(com.imagos.axilice.core.installer.SubentityOT entData, java.lang.String parentGuiName, long selEntity, boolean readonly)
 – **Usage**
 * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.
 – **Parameters**
 * entData - SubentityOT, data of the detail subentity to select.
 * parentGuiName - String,
 * selEntity - long, id of the selected instance of the subentity.
 * readonly - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.
-
- *showSubentitySelectionDialog*
 public void **showSubentitySelectionDialog**(
 com.imagos.axilice.core.installer.SubentityOT entData)

- **Usage**
 - * Opens the selection dialog that enables the user to select an instance of a multiple selection detail subentity.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to select.
-
- *undoEntityEdition*
public void undoEntityEdition()
 - *updateEntityData*
public abstract void updateEntityData()
- **Usage**
 - * Updates the selected instance (registry) data with the content panel's data typed by the user.
-
- *viewEntity*
public abstract OperationResult viewEntity()
- **Usage**
 - * View entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), in read-only mode.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

7.1.2 CLASS MeasureUnitClient

DECLARATION

```
public class MeasureUnitClient
extends com.imagos.axilice.client.gui.common.CommonClient
```

FIELDS

- public static final String AXILICE_LOG_EDITMEASUREUNIT_GLADEPATH
 - Message for logging the entity glade configuration file path.

CONSTRUCTORS

- *MeasureUnitClient*
public MeasureUnitClient()

METHODS

- *build_extension_fields_view*
public void build_extension_fields_view()

- *buildSelectionSets*
public void buildSelectionSets()

- *clearEntityData*
public void clearEntityData()

- *fillEntityData*
public void fillEntityData()

- *getEntityData*
public BaseVO getEntityData()

- *getEntityFields*
public ArrayList getEntityFields()

- *getRevision*
public static String getRevision()

– Usage

* Returns tag Id assigned to CVS source file.

- *getWidgets*
public void getWidgets()

- *init_specific_components*
public void init_specific_components()

- *setEditionMode*
public void setEditionMode(boolean enabled)

- *setWidgetsValues*
public void setWidgetsValues(com.imagos.axilice.core.common.BaseVO entity)

- *updateEntityData*
public void updateEntityData()

- *validateEntity*
public OperationResult validateEntity()

METHODS INHERITED FROM CLASS com.imagos.axilice.client.gui.common.CommonClient

(in 17.1.2, page 402)

- *addAxiliceEventListener*
public void addAxiliceEventListener(com.imagos.axilice.client.gui.AxiliceEventListener l)
- *addEntity*
public ArrayList addEntity()

- *addExistingSubentity*

```
public void addExistingSubentity( com.imagos.axilice.core.installer.SubentityOT
ent )
```

 - **Usage**
 - * Shows a dialog for selection of a new subentity
 - **Parameters**
 - * **ent** - SubentityOT. Encapsulate data type of subentity

-
- *addSubentity*

```
public void addSubentity( com.imagos.axilice.core.common.BaseVO subentity,
com.imagos.axilice.core.common.MultiSelectionSubentityOT subE )
```

 - **Usage**
 - * Adds a existent entity to a multiselection list
 - **Parameters**
 - * **subentity** - Subentity to be added
 - * **subE** - Object that encapsulates information about the subentity

-
- *addTrazabilityTree*

```
public void addTrazabilityTree( TabFolder tabFolder )
```

- *build_extension_fields_view*

```
public abstract void build_extension_fields_view( )
```

 - **Usage**
 - * Build the view of the extension fields.

It must implement only those clients who require it

-
- *buildEntity*

```
public AxiliceEntityView buildEntity( )
```
 - *buildEntityDetail*

```
public AxiliceEntityView buildEntityDetail(
com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData )
```
 - *buildEntitySelectionView*

```
public AxiliceEntitySelectionView buildEntitySelectionView( )
```

 - **Usage**
 - * Build view of selection entities
 - **Returns** - AxilicePartnerSelectionView. SelectionView created

-
- *buildSelectionSets*

```
public abstract void buildSelectionSets( )
```

 - **Usage**
 - * Build 2 sets of SubentityOT: lSelectionSubentities and lMultiSelectionSubentities

These sets are used to connect the widgets with the handlers for the selection buttons and the multiseleccion panels

It must implement only those clients who require it

-
- *buildSubentityMultiSelection*

```
public void buildSubentityMultiSelection( )
```
 - *clearEntityData*

```
public void clearEntityData( )
```

- *clearParentField*

```
public OperationResult clearParentField( java.util.ArrayList selRegistries,
java.lang.String parentField )
```

 - **Usage**
 - * Clears the value of the field indicated
 - **Parameters**
 - * selRegistries - ArrayList. Index list of records to process
 - * parentField - Name of the field to clear
 - **Returns** - OperationResult. Result of operation

- *delEntity*

```
public OperationResult delEntity( )
```

- *delEntity*

```
public OperationResult delEntity( java.util.ArrayList selRegistries )
```

- *delEntity*

```
public OperationResult delEntity( com.imagos.axilice.core.common.BaseVO b )
```

 - **Usage**
 - * Delete a entity from database
 - **Parameters**
 - * b - BaseVO. Entity to delete
 - **Returns** - OperationResult. Result of the operation

- *delSubentities*

```
public OperationResult delSubentities( java.util.ArrayList selRegistries )
```

 - **Usage**
 - * Delete a set of subentities from record list of controller, not from database
 - Set the parent field of subentity to null
 - **Parameters**
 - * selRegistries - List of index to delete
 - **Returns** - OperationResult. Result of the operation

- *editEntity*

```
public OperationResult editEntity( )
```

- *fillEntityData*

```
public void fillEntityData( )
```

- *fillEntityExtensionTable*

```
public void fillEntityExtensionTable( java.lang.Long id )
```

 - **Usage**
 - * Retrieve extension field values from the server for a given entity and displays them in the view
 - **Parameters**
 - * id - Long. Key of the entity whose values are to be displayed

- *fillEntityTable*

```
public void fillEntityTable( )
```

 - **Usage**
 - * Load records from database and fill the entity view table and controller record list

- *fillEntityTable*

```
public void fillEntityTable( java.util.ArrayList IR )
```

- **Usage**
 - * Fill the entity view table with the records given
- **Parameters**
 - * lR - ArrayList. Records to be displayed

- *getClassEntity*
public Class getClassEntity()
 - **Returns** - Class of the entity for which the controller is built

- *getDialogName*
public String getDialogName()
- *getEditGladePath*
public String getEditGladePath()
- *getEditionGui*
public Gui getEditionGui()
 - **Usage**
 - * Returns the edition Gui instance.
 - **Returns** - Gui, entity edition Gui.

- *getEditWindow*
public SwtTopLevelWindow getEditWindow()
- *getEntityData*
public abstract BaseVO getEntityData()
- *getEntityFields*
public ArrayList getEntityFields()
- *getLogger*
public Logger getLogger()
- *getParentEntityLauncher*
public EntityLauncher getParentEntityLauncher()
 - **Returns** - EntityLauncher of parent entity if is a embedded entity or null otherwise

- *getRegistries*
public Collection getRegistries()
 - **Returns** - lRegistries

- *getRegistries*
public List getRegistries(java.util.ArrayList lIds)
 - **Usage**
 - * Get a list of entities stored by the controller
 - **Parameters**
 - * lIds - Index list
 - **Returns** - List of entities

- *getView*
public AxiliceEntityView getView()
 - **Returns** - AxiliceEntityView. Main view or Detail view if is a embedded entity.

- *getWidgets*
public abstract void getWidgets()
 - **Usage**
 - * Get the widgets from the glade and stored in local variables.

Must be overwritten

-
- *init_specific_components*
 public abstract void **init_specific_components**()
 – **Usage**
 * Initializes the specific components of the client

 - *printEntity*
 public OperationResult **printEntity**()

 - *printEntity*
 public OperationResult **printEntity**(java.lang.String reportPath, java.util.HashMap parameters)

 - *queryEntity*
 public ArrayList **queryEntity**()

 - *queryEntity*
 public ArrayList **queryEntity**(java.lang.String fieldId, java.lang.String value)

 - *queryEntityWithFormat*
 public ArrayList **queryEntityWithFormat**(java.lang.String fieldId, java.lang.String value)

 - *queryFreeEntities*
 public ArrayList **queryFreeEntities**(com.imagos.axilice.core.common.MultiSelectionSubentityOT mSubE)
 – **Usage**
 * Search entities not associated with a parent entity (null parent field)
 – **Parameters**
 * mSubE - SubentityOT for the parent entity that must be null
 – **Returns** - ArrayList. List of entities found

 - *querySubentities*
 public ArrayList **querySubentities**(com.imagos.axilice.core.installer.SubentityOT entData)

 - *refreshEntity*
 public OperationResult **refreshEntity**()

 - *refreshEntity*
 public BaseVO **refreshEntity**(com.imagos.axilice.core.common.BaseVO entity)

 - *refreshParentSelectedEntity*
 public OperationResult **refreshParentSelectedEntity**()
 – **Usage**
 * Reload from database the value of the parent entity of the selected entity
 – **Returns** - OperationResult. The result of the operation.

 - *removeRegistries*
 public OperationResult **removeRegistries**(java.util.ArrayList selRegistries, java.lang.String parentField)
 – **Usage**
 * Deletes a list of the records stored by the controller. Also clean your parent fields and updates the database
 – **Parameters**
 * selRegistries - Index list to remove
 * parentField - Field name that links to the parent entity
 – **Returns** - OperationResult. Result of the operation

- *saveEntity*
public OperationResult saveEntity()
- *saveEntity*
public OperationResult saveEntity(java.util.ArrayList lIds)
- *saveEntity*
public OperationResult saveEntity(com.imagos.axilice.core.common.BaseVO b)
 - **Usage**
 - * Save or update a entity to database
 - **Parameters**
 - * b - BaseVO. Entity to save or update
 - **Returns** - OperationResult. Result of the operation
- *selectEntity*
public void selectEntity(java.lang.Long idEntity)
- *selectSubentity*
public void selectSubentity(java.lang.String parentFieldId, com.imagos.axilice.core.common.BaseVO subentityOT)
- *selectTreeItem*
public void selectTreeItem(java.lang.Long id)
 - **Usage**
 - * Select a item in the entity tree. No me gusta esto aqui
 - **Parameters**
 - * id - Long. Index of item to select
- *setDialogName*
public void setDialogName(java.lang.String dialogName)
- *setEditGladePath*
public void setEditGladePath(java.lang.String editGladePath)
- *setEditionMode*
public abstract void setEditionMode(boolean enabled)
- *setRegistries*
public void setRegistries(java.util.Collection collection)
 - **Usage**
 - * Setter for lRegistries
- *setWidgetsValues*
public abstract void setWidgetsValues(com.imagos.axilice.core.common.BaseVO entity)
 - **Usage**
 - * Set the values of the widgets on the edit dialog.
 - Must be overwritten.
 - **Parameters**
 - * entity - BaseVO. Entity that contains the values to be displayed
- *showErrorMessages*
public void showErrorMessages(com.imagos.axilice.core.util.OperationResult result)
- *updateEntityData*
public void updateEntityData()
- *updateRegistry*
public void updateRegistry(com.imagos.axilice.core.common.BaseVO ent)

- **Usage**
 - * Add or update an entity (to the controller records)
 - **Parameters**
 - * **ent** - Entity to add or update
-
- *updateViews*
public void updateViews(boolean updateFromParent)
 - **Usage**
 - * Force the update of the views associated
 - **Parameters**
 - * **updateFromParent** - Indicates whether the view must be updated by reading the values from the parent entity. takes effect only when the entity is detail (is embedded in another)
-
- *validateEntity*
public abstract OperationResult validateEntity()
 - **Usage**
 - * Validate the values of the widgets in the edit dialog.
 - **Returns** - OperationResult. Result of the validation.
-
- *viewEntity*
public OperationResult viewEntity()

METHODS INHERITED FROM CLASS com.imagos.axilice.core.installer.EntityLauncher

(in 24.2.2, page 532)

- *addEntity*
public abstract ArrayList addEntity()
 - **Usage**
 - * Add entity button clicked handler. Show the entity content screen ready to start inserting data for a new instance of the entity (new registry).
 - **Returns** - ArrayList<ResultMessage>, list of errors if any has taken place, null in any other case.
-
- *buildEditEntityGUI*
public void buildEditEntityGUI()
 - **Usage**
 - * Builds the edit entity dialog window from the corresponding glade interface file. Stores the entity dialog instance for posterior accesses.
-
- *buildEntity*
public AxiliceEntityView buildEntity()
 - **Usage**
 - * Builds the entity GUI and show it on screen to the user.
-
- *buildEntity*
public AxiliceDetailEntityView buildEntity(SwtWidget parent, com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData, java.lang.String rootName)
 - **Usage**

- * Builds the entity GUI and show it on screen to the user. The entity GUI will be embedded into the parent composite widget.

– **Parameters**

- * **parent** - Composite, parent holder for embedding the entity view.
- * **subentityData** - SubentityOT, data of the subentity to build if the entity launcher will be used to handle a multi-selection detail entity view.
- * **rootName** - String, name of the root widget that will be embedded into the master entity edition panel.

– **Returns** - AxiliceDetailView, the entity detail view built.

• *cancelEntityEdition*

public void cancelEntityEdition()

• *clearEntityData*

public abstract void clearEntityData()

– **Usage**

- * Clears the entity content panel by resetting the content widgets.

• *delEntity*

public abstract OperationResult delEntity()

– **Usage**

- * Delete entity button clicked handler. Deletes the selected instance of the entity (selected registry) from the database.

– **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

• *delEntity*

public abstract OperationResult delEntity(java.util.ArrayList lSelRegistries)

– **Usage**

- * Delete entity button clicked handler. Deletes the selected instances of the entity (selected registries) from the database. Its a multiple delete method.

– **Parameters**

- * **lSelRegistries** - ArrayList<Long>, list of selected instances of the entity (selected registries). The list will contain only the internal id of the selected registries.

– **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

• *deselectSubentity*

**public void deselectSubentity(java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT)**

– **Usage**

- * Updates an entity instance by removing the selected value for a subentity field.

– **Parameters**

- * **parentFieldId** - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
- * **subentityOT** - BaseVO, subentity's deselected instance (registry) value.

• *editEntity*

public abstract OperationResult editEntity()

– **Usage**

- * Edit entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), and allows the user to modify any content of the entity registry.

– **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

-
- *fillEntityData*
 public abstract void **fillEntityData**()
 – **Usage**
 * Fills the entity content panel with the data of the selected instance (registry) of the entity.

 - *fireExceptionAndExit*
 public void **fireExceptionAndExit**(java.lang.String msg)
 – **Usage**
 * Fire a exception and stop the program
 – **Parameters**
 * msg -

 - *formatRegistries*
 public ArrayList **formatRegistries**()
 - *formatRegistries*
 public ArrayList **formatRegistries**(java.util.ArrayList lRegistries)
 - *getDialog*
 public SwtDialog **getDialog**(java.lang.String dialogName)
 - *getEntity*
 public EntityOT **getEntity**()
 – **Usage**
 * Gets the entity data.
 – **Returns** - EntityOT, entity data.

 - *getEntityData*
 public abstract BaseVO **getEntityData**()
 – **Usage**
 * Gets the content panel's data typed by the user formatted as an entity VO object.

 - *getEntityEditViewName*
 public String **getEntityEditViewName**(java.lang.String parentGuiName)
 - *getEntityFields*
 public abstract ArrayList **getEntityFields**()
 – **Usage**
 * Gets the list of defined fields for an entity
 – **Returns** - ArrayList, list of entity fields.

 - *getEntityViewName*
 public String **getEntityViewName**(java.lang.String parentGuiName)
 - *getLRegistries*
 public ArrayList **getLRegistries**()
 – **Usage**
 * Getter for lRegistries

 - *getRegistry*
 public BaseVO **getRegistry**(long id)
 - *getRevision*
 public static String **getRevision**()
 – **Usage**
 * Returns tag Id assigned to CVS source file.

- *getSelectedEntity*
public BaseVO getSelectedEntity()
 - **Usage**
 - * Gets the selected instance (registry) of the entity.
 - **Returns** - BaseVO, selected instance (registry) of the entity.

- *getSelectedId*
public Long getSelectedId()
 - **Usage**
 - * Gets the internal id of the selected entity registry.
 - **Returns** - Long, internal id of the selected entity registry, null in case no entity registry has been selected yet.

- *getSwtWidget*
public SwtWidget getSwtWidget(java.lang.String widgetName)

- *getWidget*
public Widget getWidget(java.lang.String widgetName)

- *initEntityLauncher*
public void initEntityLauncher(Logger logger, Gui parentGui, SwtWidget parentPanel)
 - **Usage**
 - * Inits the entity launcher context and prepares the launcher to be started.
 - **Parameters**
 - * **logger** - Logger, logger instance for printing logging messages.
 - * **parentGuiName** - String, name of the JavaGlade instance that built the parent where the entity will be embedded.
 - * **parentPanel** - Composite, parent panel where the entity contents will be embedded.

- *isDetail*
public boolean isDetail()
 - **Usage**
 - * Indicates whether is a main entity or is embedded in another

- *on_entitySelectionButton_clicked*
public void on_entitySelectionButton_clicked(Event e)
 - **Usage**
 - * Handler for the subentity selection buttons (magnifying glass buttons) clicked. Shows the selection dialog for selecting a valid instance of a subentity.
 - **Parameters**
 - * **e** - Event, magnifying glass button clicked event data.

- *printEntity*
public abstract OperationResult printEntity()
 - **Usage**
 - * Print entity button clicked handler. Builds the report with the data of the selected instance of the entity (selected registry), and allows the user to print the elaborated report.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *printWidgets*
public void printWidgets()

- *queryEntity*
public abstract ArrayList queryEntity()

- **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) available.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.
-
- *queryEntity*
 public abstract ArrayList **queryEntity**(java.lang.String fieldId, java.lang.String value)
 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria.
 - **Parameters**
 - * fieldId - String, name of the entity field that will be queried.
 - * value - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.
-
- *queryEntityWithFormat*
 public abstract ArrayList **queryEntityWithFormat**(java.lang.String fieldId, java.lang.String value)
 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria and show them (formatted as a list) to the user in the entity's screen.
 - **Parameters**
 - * fieldId - String, name of the entity field that will be queried.
 - * value - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList<RegistryOT>, list of entity instances (registries) taken as the result of the query operation.
-
- *querySubentities*
 public abstract ArrayList **querySubentities**(com.imagos.axilice.core.installer.SubentityOT entData)
 - **Usage**
 - * Queries the selected entity for all the detail subentities associated with it.
 - **Parameters**
 - * entData - SubentityOT, data of the detail subentity to query for.
 - **Returns** - ArrayList, list of detail subentity registries assigned to the master detail registry.
-
- *saveEntity*
 public abstract OperationResult **saveEntity**()
 - **Usage**
 - * Save entity button clicked handler. Update the system with the data of the edited entity registry. It's called for saving new entity registries (add a new registry) or for updating the contents of existing entity registries.
 - **Returns** - OperationResult, Result of the operation and list of messages generated through the operation.
-
- *saveEntityRegistry*
 public void **saveEntityRegistry**()
 - *selectEntity*
 public abstract void **selectEntity**(java.lang.Long idEntity)

- **Usage**
 - * Marks an instance (registry) of the entity as the selected.
 - **Parameters**
 - * `idEntity` - Long, internal id of the selected instance (registry).
-
- *selectSubentity*

```
public void selectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

 - **Usage**
 - * Updates an entity instance by setting up the selected value for a subentity field.
 - **Parameters**
 - * `parentFieldId` - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * `subentityOT` - BaseVO, subentity's selected instance (registry) value.
-
- *setEntity*

```
public void setEntity( com.imagos.axilice.core.installer.EntityOT entity )
```

 - **Usage**
 - * Sets up the entity name.
 - **Parameters**
 - * `name` - String, entity name.
-
- *setRegistries*

```
public void setRegistries( java.util.ArrayList lRegistries )
```

 - **Usage**
 - * Sets the list of registries that are showed in the entity view controlled by the Entity Launcher instance. It's used for detail entities where queries are made in the master entity launcher.
 - **Parameters**
 - * `lRegistries` - ArrayList<BaseVO>, list of shown registries of the entity.
-
- *showSelectionDialog*

```
public void showSelectionDialog( com.imagos.axilice.core.installer.SubentityOT
entData )
```

 - **Usage**
 - * Opens the selection dialog that enables the user to select an instance of a subentity.
 - **Parameters**
 - * `entData` - SubentityOT, data of the detail subentity to select.
-
- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT
entData, long selEntity, boolean readonly )
```

 - **Usage**
 - * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.
 - **Parameters**
 - * `entData` - SubentityOT, data of the detail subentity to select.
 - * `selEntity` - long, id of the selected instance of the subentity.
 - * `readonly` - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.
-
- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT
entData, java.lang.String parentGuiName, long selEntity, boolean readonly )
```

- **Usage**
 - * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to select.
 - * **parentGuiName** - String,
 - * **selEntity** - long, id of the selected instance of the subentity.
 - * **readonly** - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.
-
- *showSubentitySelectionDialog*

```
public void showSubentitySelectionDialog(
com.imagos.axilice.core.installer.SubentityOT entData )
```

 - **Usage**
 - * Opens the selection dialog that enables the user to select an instance of a multiple selection detail subentity.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to select.
-
- *undoEntityEdition*

```
public void undoEntityEdition( )
```
 - *updateEntityData*

```
public abstract void updateEntityData( )
```

 - **Usage**
 - * Updates the selected instance (registry) data with the content panel's data typed by the user.
-
- *viewEntity*

```
public abstract OperationResult viewEntity( )
```

 - **Usage**
 - * View entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), in read-only mode.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

7.1.3 CLASS ProductClient

DECLARATION

```
public class ProductClient
extends com.imagos.axilice.client.gui.common.CommonClient
```

FIELDS

- public static final String AXILICE_LOG_EDITPRODUCT_GLADEPATH
 - Message for logging the entity glade configuration file path.

CONSTRUCTORS

- *ProductClient*
 public **ProductClient**()
 – **Usage**
 * Default constructor for ProductClient objects.

METHODS

- *addProductImage*
 public void **addProductImage**()
- *build_extension_fields_view*
 public void **build_extension_fields_view**()
- *buildSelectionSets*
 public void **buildSelectionSets**()
- *clearEntityData*
 public void **clearEntityData**()
- *delProductImage*
 public void **delProductImage**()
- *fillEntityData*
 public void **fillEntityData**()
- *getEntityData*
 public BaseVO **getEntityData**()
- *getEntityFields*
 public ArrayList **getEntityFields**()
- *getRevision*
 public static String **getRevision**()
 – **Usage**
 * Returns tag Id assigned to CVS source file.
- *getWidgets*
 public void **getWidgets**()
- *init_specific_components*
 public void **init_specific_components**()
- *queryEntity*
 public ArrayList **queryEntity**(java.lang.String fieldId, java.lang.String value)
- *saveEntity*
 public **OperationResult** **saveEntity**()

- *saveEntity*
public OperationResult saveEntity(com.imagos.axilice.core.common.BaseVO entity)
- *selectEntity*
public void selectEntity(java.lang.Long idEntity)
- *setEditionMode*
public void setEditionMode(boolean enabled)
- *setWidgetsValues*
public void setWidgetsValues(com.imagos.axilice.core.common.BaseVO entity)
- *showFirstProductImage*
public void showFirstProductImage()
- *showLastProductImage*
public void showLastProductImage()
- *showNextProductImage*
public void showNextProductImage()
- *showPreviousProductImage*
public void showPreviousProductImage()
- *showSelectedProductImage*
public void showSelectedProductImage()
- *updateEntityData*
public void updateEntityData()
- *validateEntity*
public OperationResult validateEntity()

METHODS INHERITED FROM CLASS com.imagos.axilice.client.gui.common.CommonClient

(in 17.1.2, page 402)

- *addAxiliceEventListener*
public void addAxiliceEventListener(com.imagos.axilice.client.gui.AxiliceEventListener l)
- *addEntity*
public ArrayList addEntity()
- *addExistingSubentity*
public void addExistingSubentity(com.imagos.axilice.core.installer.SubentityOT ent)
 - **Usage**
 - * Shows a dialog for selection of a new subentity
 - **Parameters**
 - * ent - SubentityOT. Encapsulate data type of subentity
- *addSubentity*
public void addSubentity(com.imagos.axilice.core.common.BaseVO subentity, com.imagos.axilice.core.common.MultiSelectionSubentityOT subE)
 - **Usage**

- * Adds a existent entity to a multiselection list
 - **Parameters**
 - * **subentity** - Subentity to be added
 - * **subE** - Object that encapsulates information about the subentity

 - *addTrazabilityTree*
public void **addTrazabilityTree**(TabFolder tabFolder)
 - *build_extension_fields_view*
public abstract void **build_extension_fields_view**()
 - **Usage**
 - * Build the view of the extension fields.
- It must implement only those clients who require it
-
- *buildEntity*
public AxiliceEntityView **buildEntity**()
- *buildEntityDetail*
public AxiliceEntityView **buildEntityDetail**(
com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData)
- *buildEntitySelectionView*
public AxiliceEntitySelectionView **buildEntitySelectionView**()
- **Usage**
 - * Build view of selection entities
- **Returns** - AxilicePartnerSelectionView. SelectionView created
-
- *buildSelectionSets*
public abstract void **buildSelectionSets**()
- **Usage**
 - * Build 2 sets of SubentityOT: lSelectionSubentities and lMultiSelectionSubentities
- These sets are used to connect the widgets with the handlers for the selection buttons and the multiselection panels
- It must implement only those clients who require it
-
- *buildSubentityMultiSelection*
public void **buildSubentityMultiSelection**()
- *clearEntityData*
public void **clearEntityData**()
- *clearParentField*
public OperationResult **clearParentField**(java.util.ArrayList selRegistries,
java.lang.String parentField)
- **Usage**
 - * Clears the value of the field indicated
- **Parameters**
 - * **selRegistries** - ArrayList. Index list of records to process
 - * **parentField** - Name of the field to clear
- **Returns** - OperationResult. Result of operation
-
- *delEntity*
public OperationResult **delEntity**()
- *delEntity*
public OperationResult **delEntity**(java.util.ArrayList selRegistries)

- *delEntity*
 public OperationResult **delEntity**(com.imagos.axilice.core.common.BaseVO **b**)

 - **Usage**
 * Delete a entity from database
 - **Parameters**
 * **b** - BaseVO. Entity to delete
 - **Returns** - OperationResult. Result of the operation

- *delSubentities*
 public OperationResult **delSubentities**(java.util.ArrayList **selRegistries**)

 - **Usage**
 * Delete a set of subentities from record list of controller, not from database

 Set the parent field of subentity to null
 - **Parameters**
 * **selRegistries** - List of index to delete
 - **Returns** - OperationResult. Result of the operation

- *editEntity*
 public OperationResult **editEntity**()

- *fillEntityData*
 public void **fillEntityData**()

- *fillEntityExtensionTable*
 public void **fillEntityExtensionTable**(java.lang.Long **id**)

 - **Usage**
 * Retrieve extension field values from the server for a given entity and displays them in the view
 - **Parameters**
 * **id** - Long. Key of the entity whose values are to be displayed

- *fillEntityTable*
 public void **fillEntityTable**()

 - **Usage**
 * Load records from database and fill the entity view table and controller record list

- *fillEntityTable*
 public void **fillEntityTable**(java.util.ArrayList **lR**)

 - **Usage**
 * Fill the entity view table with the records given
 - **Parameters**
 * **lR** - ArrayList. Records to be displayed

- *getClassEntity*
 public Class **getClassEntity**()

 - **Returns** - Class of the entity for which the controller is built

- *getDialogName*
 public String **getDialogName**()

- *getEditGladePath*
 public String **getEditGladePath**()

- *getEditionGui*
public Gui **getEditionGui**()
 - **Usage**
 - * Returns the edition Gui instance.
 - **Returns** - Gui, entity edition Gui.

- *getEditWindow*
public SwtTopLevelWindow **getEditWindow**()
- *getEntityData*
public abstract BaseVO **getEntityData**()
- *getEntityFields*
public ArrayList **getEntityFields**()
- *getLogger*
public Logger **getLogger**()
- *getParentEntityLauncher*
public EntityLauncher **getParentEntityLauncher**()
 - **Returns** - EntityLauncher of parent entity if is a embedded entity or null otherwise

- *getRegistries*
public Collection **getRegistries**()
 - **Returns** - lRegistries

- *getRegistries*
public List **getRegistries**(java.util.ArrayList lIds)
 - **Usage**
 - * Get a list of entities stored by the controller
 - **Parameters**
 - * lIds - Index list
 - **Returns** - List of entities

- *getView*
public AxiliceEntityView **getView**()
 - **Returns** - AxiliceEntityView. Main view or Detail view if is a embedded entity.

- *getWidgets*
public abstract void **getWidgets**()
 - **Usage**
 - * Get the widgets from the glade and stored in local variables.
 - Must be overwritten

- *init_specific_components*
public abstract void **init_specific_components**()
 - **Usage**
 - * Initializes the specific components of the client

- *printEntity*
public OperationResult **printEntity**()
- *printEntity*
public OperationResult **printEntity**(java.lang.String reportPath, java.util.HashMap parameters)

- *queryEntity*
public ArrayList **queryEntity**()
- *queryEntity*
public ArrayList **queryEntity**(java.lang.String fieldId, java.lang.String value)
- *queryEntityWithFormat*
public ArrayList **queryEntityWithFormat**(java.lang.String fieldId, java.lang.String value)
- *queryFreeEntities*
public ArrayList **queryFreeEntities**(com.imagos.axilice.core.common.MultiSelectionSubentityOT mSubE)
 - **Usage**
 - * Search entities not associated with a parent entity (null parent field)
 - **Parameters**
 - * mSubE - SubentityOT for the parent entity that must be null
 - **Returns** - ArrayList. List of entities found
- *querySubentities*
public ArrayList **querySubentities**(com.imagos.axilice.core.installer.SubentityOT entData)
- *refreshEntity*
public OperationResult **refreshEntity**()
- *refreshEntity*
public BaseVO **refreshEntity**(com.imagos.axilice.core.common.BaseVO entity)
- *refreshParentSelectedEntity*
public OperationResult **refreshParentSelectedEntity**()
 - **Usage**
 - * Reload from database the value of the parent entity of the selected entity
 - **Returns** - OperationResult. The result of the operation.
- *removeRegistries*
public OperationResult **removeRegistries**(java.util.ArrayList selRegistries, java.lang.String parentField)
 - **Usage**
 - * Deletes a list of the records stored by the controller. Also clean your parent fields and updates the database
 - **Parameters**
 - * selRegistries - Index list to remove
 - * parentField - Field name that links to the parent entity
 - **Returns** - OperationResult. Result of the operation
- *saveEntity*
public OperationResult **saveEntity**()
- *saveEntity*
public OperationResult **saveEntity**(java.util.ArrayList lIds)
- *saveEntity*
public OperationResult **saveEntity**(com.imagos.axilice.core.common.BaseVO b)
 - **Usage**
 - * Save or update a entity to database
 - **Parameters**
 - * b - BaseVO. Entity to save or update
 - **Returns** - OperationResult. Result of the operation

- *selectEntity*
public void selectEntity(java.lang.Long idEntity)
- *selectSubentity*
public void selectSubentity(java.lang.String parentFieldId, com.imagos.axilice.core.common.BaseVO subentityOT)
- *selectTreeItem*
public void selectTreeItem(java.lang.Long id)
 - Usage
 - * Select a item in the entity tree. No me gusta esto aqui
 - Parameters
 - * id - Long. Index of item to select
- *setDialogName*
public void setDialogName(java.lang.String dialogName)
- *setEditGladePath*
public void setEditGladePath(java.lang.String editGladePath)
- *setEditionMode*
public abstract void setEditionMode(boolean enabled)
- *setRegistries*
public void setRegistries(java.util.Collection collection)
 - Usage
 - * Setter for lRegistries
- *setWidgetsValues*
public abstract void setWidgetsValues(com.imagos.axilice.core.common.BaseVO entity)
 - Usage
 - * Set the values of the widgets on the edit dialog.
 - Must be overwritten.
 - Parameters
 - * entity - BaseVO. Entity that contains the values to be displayed
- *showErrorMessages*
public void showErrorMessages(com.imagos.axilice.core.util.OperationResult result)
- *updateEntityData*
public void updateEntityData()
- *updateRegistry*
public void updateRegistry(com.imagos.axilice.core.common.BaseVO ent)
 - Usage
 - * Add or update an entity (to the controller records)
 - Parameters
 - * ent - Entity to add or update
- *updateViews*
public void updateViews(boolean updateFromParent)
 - Usage
 - * Force the update of the views associated
 - Parameters
 - * updateFromParent - Indicates whether the view must be updated by reading the values from the parent entity. takes effect only when the entity is detail (is embedded in another)

-
- *validateEntity*
 public abstract **OperationResult validateEntity()**
 – **Usage**
 * Validate the values of the widgets in the edit dialog.
 Must be overwritten.
 – **Returns** - **OperationResult**. Result of the validation.
-
- *viewEntity*
 public **OperationResult viewEntity()**

METHODS INHERITED FROM CLASS `com.imagos.axilice.core.installer.EntityLauncher`

(in 24.2.2, page 532)

- *addEntity*
 public abstract **ArrayList addEntity()**
 – **Usage**
 * Add entity button clicked handler. Show the entity content screen ready to start inserting data for a new instance of the entity (new registry).
 – **Returns** - **ArrayList<ResultMessage>**, list of errors if any has taken place, null in any other case.
-
- *buildEditEntityGUI*
 public void **buildEditEntityGUI()**
 – **Usage**
 * Builds the edit entity dialog window from the corresponding glade interface file. Stores the entity dialog instance for posterior accesses.
-
- *buildEntity*
 public **AxiliceEntityView buildEntity()**
 – **Usage**
 * Builds the entity GUI and show it on screen to the user.
-
- *buildEntity*
 public **AxiliceDetailEntityView buildEntity(SwtWidget parent, com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData, java.lang.String rootName)**
 – **Usage**
 * Builds the entity GUI and show it on screen to the user. The entity GUI will be embedded into the parent composite widget.
 – **Parameters**
 * **parent** - Composite, parent holder for embedding the entity view.
 * **subentityData** - SubentityOT, data of the subentity to build if the entity launcher will be used to handle a multi-selection detail entity view.
 * **rootName** - String, name of the root widget that will be embedded into the master entity edition panel.
 – **Returns** - **AxiliceDetailEntityView**, the entity detail view built.
-
- *cancelEntityEdition*
 public void **cancelEntityEdition()**
-
- *clearEntityData*
 public abstract void **clearEntityData()**

- **Usage**
 - * Clears the entity content panel by resetting the content widgets.

- *delEntity*

```
public abstract OperationResult delEntity( )
```

 - **Usage**
 - * Delete entity button clicked handler. Deletes the selected instance of the entity (selected registry) from the database.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *delEntity*

```
public abstract OperationResult delEntity( java.util.ArrayList lSelRegistries )
```

 - **Usage**
 - * Delete entity button clicked handler. Deletes the selected instances of the entity (selected registries) from the database. Its a multiple delete method.
 - **Parameters**
 - * lSelRegistries - ArrayList<Long>, list of selected instances of the entity (selected registries). The list will contain only the internal id of the selected registries.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *deselectSubentity*

```
public void deselectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

 - **Usage**
 - * Updates an entity instance by removing the selected value for a subentity field.
 - **Parameters**
 - * parentFieldId - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * subentityOT - BaseVO, subentity's deselected instance (registry) value.

- *editEntity*

```
public abstract OperationResult editEntity( )
```

 - **Usage**
 - * Edit entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), and allows the user to modify any content of the entity registry.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *fillEntityData*

```
public abstract void fillEntityData( )
```

 - **Usage**
 - * Fills the entity content panel with the data of the selected instance (registry) of the entity.

- *fireExceptionAndExit*

```
public void fireExceptionAndExit( java.lang.String msg )
```

 - **Usage**
 - * Fire a exception and stop the program
 - **Parameters**
 - * msg -

- *formatRegistries*
public ArrayList formatRegistries()
- *formatRegistries*
public ArrayList formatRegistries(java.util.ArrayList lRegistries)
- *getDialog*
public SwtDialog getDialog(java.lang.String dialogName)
- *getEntity*
public EntityOT getEntity()
 - **Usage**
 - * Gets the entity data.
 - **Returns** - EntityOT, entity data.

- *getEntityData*
public abstract BaseVO getEntityData()
 - **Usage**
 - * Gets the content panel's data typed by the user formatted as an entity VO object.

- *getEntityEditViewName*
public String getEntityEditViewName(java.lang.String parentGuiName)
- *getEntityFields*
public abstract ArrayList getEntityFields()
 - **Usage**
 - * Gets the list of defined fields for an entity
 - **Returns** - ArrayList, list of entity fields.

- *getEntityViewName*
public String getEntityViewName(java.lang.String parentGuiName)
- *getLRegistries*
public ArrayList getLRegistries()
 - **Usage**
 - * Getter for lRegistries

- *getRegistry*
public BaseVO getRegistry(long id)
- *getRevision*
public static String getRevision()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.

- *getSelectedEntity*
public BaseVO getSelectedEntity()
 - **Usage**
 - * Gets the selected instance (registry) of the entity.
 - **Returns** - BaseVO, selected instance (registry) of the entity.

- *getSelectedId*
public Long getSelectedId()
 - **Usage**
 - * Gets the internal id of the selected entity registry.
 - **Returns** - Long, internal id of the selected entity registry, null in case no entity registry has been selected yet.

- *getSwtWidget*
public SwtWidget getSwtWidget(java.lang.String widgetName)
- *getWidget*
public Widget getWidget(java.lang.String widgetName)
- *initEntityLauncher*
public void initEntityLauncher(Logger logger, Gui parentGui, SwtWidget parentPanel)
 - **Usage**
 - * Initiates the entity launcher context and prepares the launcher to be started.
 - **Parameters**
 - * **logger** - Logger, logger instance for printing logging messages.
 - * **parentGui** - String, name of the JavaGlade instance that built the parent where the entity will be embedded.
 - * **parentPanel** - Composite, parent panel where the entity contents will be embedded.
- *isDetail*
public boolean isDetail()
 - **Usage**
 - * Indicates whether is a main entity or is embedded in another
- *on_entitySelectionButton_clicked*
public void on_entitySelectionButton_clicked(Event e)
 - **Usage**
 - * Handler for the subentity selection buttons (magnifying glass buttons) clicked. Shows the selection dialog for selecting a valid instance of a subentity.
 - **Parameters**
 - * **e** - Event, magnifying glass button clicked event data.
- *printEntity*
public abstract OperationResult printEntity()
 - **Usage**
 - * Print entity button clicked handler. Builds the report with the data of the selected instance of the entity (selected registry), and allows the user to print the elaborated report.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.
- *printWidgets*
public void printWidgets()
- *queryEntity*
public abstract ArrayList queryEntity()
 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) available.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.
- *queryEntity*
public abstract ArrayList queryEntity(java.lang.String fieldId, java.lang.String value)
 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria.
 - **Parameters**

- * **fieldId** - String, name of the entity field that will be queried.
 - * **value** - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.

- *queryEntityWithFormat*

```
public abstract ArrayList queryEntityWithFormat( java.lang.String fieldId,
java.lang.String value )
```

 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria and show them (formatted as a list) to the user in the entity's screen.
 - **Parameters**
 - * **fieldId** - String, name of the entity field that will be queried.
 - * **value** - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList<RegistryOT>, list of entity instances (registries) taken as the result of the query operation.

- *querySubentities*

```
public abstract ArrayList querySubentities(
com.imagos.axilice.core.installer.SubentityOT entData )
```

 - **Usage**
 - * Queries the selected entity for all the detail subentities associated with it.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to query for.
 - **Returns** - ArrayList, list of detail subentity registries assigned to the master detail registry.

- *saveEntity*

```
public abstract OperationResult saveEntity( )
```

 - **Usage**
 - * Save entity button clicked handler. Update the system with the data of the edited entity registry. It's called for saving new entity registries (add a new registry) or for updating the contents of existing entity registries.
 - **Returns** - OperationResult, Result of the operation and list of messages generated through the operation.

- *saveEntityRegistry*

```
public void saveEntityRegistry( )
```

- *selectEntity*

```
public abstract void selectEntity( java.lang.Long idEntity )
```

 - **Usage**
 - * Marks an instance (registry) of the entity as the selected.
 - **Parameters**
 - * **idEntity** - Long, internal id of the selected instance (registry).

- *selectSubentity*

```
public void selectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

 - **Usage**
 - * Updates an entity instance by setting up the selected value for a subentity field.
 - **Parameters**
 - * **parentFieldId** - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.

-
- * subentityOT - BaseVO, subentity's selected instance (registry) value.
-
- *setEntity*
 public void **setEntity**(com.imagos.axilice.core.installer.EntityOT entity)
 - Usage
 - * Sets up the entity name.
 - Parameters
 - * name - String, entity name.
-
- *setRegistries*
 public void **setRegistries**(java.util.ArrayList lRegistries)
 - Usage
 - * Sets the list of registries that are showed in the entity view controlled by the Entity Launcher instance. It's used for detail entities where queries are made in the master entity launcher.
 - Parameters
 - * lRegistries - ArrayList<BaseVO>, list of shown registries of the entity.
-
- *showSelectionDialog*
 public void **showSelectionDialog**(com.imagos.axilice.core.installer.SubentityOT entData)
 - Usage
 - * Opens the selection dialog that enables the user to select an instance of a subentity.
 - Parameters
 - * entData - SubentityOT, data of the detail subentity to select.
-
- *showSubentityDialog*
 public void **showSubentityDialog**(com.imagos.axilice.core.installer.SubentityOT entData, long selEntity, boolean readonly)
 - Usage
 - * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.
 - Parameters
 - * entData - SubentityOT, data of the detail subentity to select.
 - * selEntity - long, id of the selected instance of the subentity.
 - * readonly - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.
-
- *showSubentityDialog*
 public void **showSubentityDialog**(com.imagos.axilice.core.installer.SubentityOT entData, java.lang.String parentGuiName, long selEntity, boolean readonly)
 - Usage
 - * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.
 - Parameters
 - * entData - SubentityOT, data of the detail subentity to select.
 - * parentGuiName - String,
 - * selEntity - long, id of the selected instance of the subentity.
 - * readonly - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.
-
- *showSubentitySelectionDialog*
 public void **showSubentitySelectionDialog**(
 com.imagos.axilice.core.installer.SubentityOT entData)

- **Usage**
 - * Opens the selection dialog that enables the user to select an instance of a multiple selection detail subentity.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to select.
-
- *undoEntityEdition*
public void undoEntityEdition()
 - *updateEntityData*
public abstract void updateEntityData()
- **Usage**
 - * Updates the selected instance (registry) data with the content panel's data typed by the user.
-
- *viewEntity*
public abstract OperationResult viewEntity()
- **Usage**
 - * View entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), in read-only mode.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

7.1.4 CLASS ProductClientListener

DECLARATION

```
public class ProductClientListener
extends GenericHandler
```

CONSTRUCTORS

- *ProductClientListener*
public ProductClientListener(Logger logger,
com.imagos.axilice.client.warehouse.ProductClient prodClient)

METHODS

- *on_product_resized*
public void on_product_resized(Event e)
- *on_productBarcode_changed*
public void on_productBarcode_changed(Event e)
- *on_productImageAdd_clicked*
public void on_productImageAdd_clicked(Event e)
- *on_productImageDelete_clicked*
public void on_productImageDelete_clicked(Event e)

- *on_productImageFirst_clicked*
public void on_productImageFirst_clicked(Event e)
- *on_productImageLast_clicked*
public void on_productImageLast_clicked(Event e)
- *on_productImageNext_clicked*
public void on_productImageNext_clicked(Event e)
- *on_productImagePrevious_clicked*
public void on_productImagePrevious_clicked(Event e)

METHODS INHERITED FROM CLASS GenericHandler

7.1.5 CLASS ProductGroupClient

DECLARATION

```
public class ProductGroupClient
extends com.imagos.axilice.core.installer.EntityLauncher
```

FIELDS

- public static final String AXILICE_LOG_EDITPRODUCTGROUP_GLADEPATH
– Message for logging the entity glade configuration file path.

CONSTRUCTORS

- *ProductGroupClient*
public ProductGroupClient()

METHODS

- *addEntity*
public ArrayList addEntity()
- *clearEntityData*
public void clearEntityData()
- *delEntity*
public OperationResult delEntity()
- *delEntity*
public OperationResult delEntity(java.util.ArrayList lSelfRegistries)

- *editEntity*
public **OperationResult** **editEntity**()
- *fillEntityData*
public void **fillEntityData**()
- *getEntityData*
public **BaseVO** **getEntityData**()
- *getEntityFields*
public **ArrayList** **getEntityFields**()
- *getSelectedEntity*
public **BaseVO** **getSelectedEntity**()
- *getSelectedId*
public **Long** **getSelectedId**()
- *printEntity*
public **OperationResult** **printEntity**()
- *queryEntity*
public **ArrayList** **queryEntity**()
- *queryEntity*
public **ArrayList** **queryEntity**(java.lang.String fieldId, java.lang.String value)
- *queryEntityWithFormat*
public **ArrayList** **queryEntityWithFormat**(java.lang.String fieldId, java.lang.String value)
- *querySubentities*
public **ArrayList** **querySubentities**(
com.imagos.axilice.core.installer.SubentityOT entData)
 - **Usage**
 - * Queries the selected entity for all the detail subentities associated with it.
 - **Parameters**
 - * entData - SubentityOT, data of the detail subentity to query for.
 - **Returns** - ArrayList, list of detail subentity registries assigned to the master detail registry.
- *saveEntity*
public **OperationResult** **saveEntity**()
- *selectEntity*
public void **selectEntity**(java.lang.Long idEntity)
- *selectSubentity*
public void **selectSubentity**(java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT)
- *updateEntityData*
public void **updateEntityData**()
- *viewEntity*
public **OperationResult** **viewEntity**()

METHODS INHERITED FROM CLASS com.imagos.axilice.core.installer.EntityLauncher

(in 24.2.2, page 532)

- *addEntity*
 public abstract ArrayList **addEntity**()
 — **Usage**
 * Add entity button clicked handler. Show the entity content screen ready to start inserting data for a new instance of the entity (new registry).
 — **Returns** - ArrayList<ResultMessage>, list of errors if any has taken place, null in any other case.

- *buildEditEntityGUI*
 public void **buildEditEntityGUI**()
 — **Usage**
 * Builds the edit entity dialog window from the corresponding glade interface file. Stores the entity dialog instance for posterior accesses.

- *buildEntity*
 public AxiliceEntityView **buildEntity**()
 — **Usage**
 * Builds the entity GUI and show it on screen to the user.

- *buildEntity*
 public AxiliceDetailEntityView **buildEntity**(SwtWidget **parent**,
 com.imagos.axilice.core.common.MultiSelectionSubentityOT **subentityData**,
 java.lang.String **rootName**)
 — **Usage**
 * Builds the entity GUI and show it on screen to the user. The entity GUI will be embedded into the parent composite widget.
 — **Parameters**
 * **parent** - Composite, parent holder for embedding the entity view.
 * **subentityData** - SubentityOT, data of the subentity to build if the entity launcher will be used to handle a multi-selection detail entity view.
 * **rootName** - String, name of the root widget that will be embedded into the master entity edition panel.
 — **Returns** - AxiliceDetailEntityView, the entity detail view built.

- *cancelEntityEdition*
 public void **cancelEntityEdition**()

- *clearEntityData*
 public abstract void **clearEntityData**()
 — **Usage**
 * Clears the entity content panel by resetting the content widgets.

- *delEntity*
 public abstract OperationResult **delEntity**()
 — **Usage**
 * Delete entity button clicked handler. Deletes the selected instance of the entity (selected registry) from the database.
 — **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *delEntity*

```
public abstract OperationResult delEntity( java.util.ArrayList lSelRegistries )
```

- **Usage**

- * Delete entity button clicked handler. Deletes the selected instances of the entity (selected registries) from the database. Its a multiple delete method.

- **Parameters**

- * lSelRegistries - ArrayList<Long>, list of selected instances of the entity (selected registries). The list will contain only the internal id of the selected registries.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *deselectSubentity*

```
public void deselectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

- **Usage**

- * Updates an entity instance by removing the selected value for a subentity field.

- **Parameters**

- * parentFieldId - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * subentityOT - BaseVO, subentity's deselected instance (registry) value.

- *editEntity*

```
public abstract OperationResult editEntity( )
```

- **Usage**

- * Edit entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), and allows the user to modify any content of the entity registry.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *fillEntityData*

```
public abstract void fillEntityData( )
```

- **Usage**

- * Fills the entity content panel with the data of the selected instance (registry) of the entity.

- *fireExceptionAndExit*

```
public void fireExceptionAndExit( java.lang.String msg )
```

- **Usage**

- * Fire a exception and stop the program

- **Parameters**

- * msg -

- *formatRegistries*

```
public ArrayList formatRegistries( )
```

- *formatRegistries*

```
public ArrayList formatRegistries( java.util.ArrayList lRegistries )
```

- *getDialog*

```
public SwtDialog getDialog( java.lang.String dialogName )
```

- *getEntity*

```
public EntityOT getEntity( )
```

- **Usage**

- * Gets the entity data.

- **Returns** - EntityOT, entity data.
- - *getEntityData*
 public abstract BaseVO **getEntityData**()
 - **Usage**
 - * Gets the content panel's data typed by the user formatted as an entity VO object.
- - *getEntityEditViewName*
 public String **getEntityEditViewName**(java.lang.String parentGuiName)
- - *getEntityFields*
 public abstract ArrayList **getEntityFields**()
 - **Usage**
 - * Gets the list of defined fields for an entity
 - **Returns** - ArrayList, list of entity fields.
- - *getEntityViewName*
 public String **getEntityViewName**(java.lang.String parentGuiName)
- - *getLRegistries*
 public ArrayList **getLRegistries**()
 - **Usage**
 - * Getter for lRegistries
- - *getRegistry*
 public BaseVO **getRegistry**(long id)
- - *getRevision*
 public static String **getRevision**()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
- - *getSelectedEntity*
 public BaseVO **getSelectedEntity**()
 - **Usage**
 - * Gets the selected instance (registry) of the entity.
 - **Returns** - BaseVO, selected instance (registry) of the entity.
- - *getSelectedId*
 public Long **getSelectedId**()
 - **Usage**
 - * Gets the internal id of the selected entity registry.
 - **Returns** - Long, internal id of the selected entity registry, null in case no entity registry has been selected yet.
- - *getSwtWidget*
 public SwtWidget **getSwtWidget**(java.lang.String widgetName)
- - *getWidget*
 public Widget **getWidget**(java.lang.String widgetName)
- - *initEntityLauncher*
 public void **initEntityLauncher**(Logger logger, Gui parentGui, SwtWidget parentPanel)
 - **Usage**
 - * Inits the entity launcher context and prepares the launcher to be started.
 - **Parameters**

- * **logger** - Logger, logger instance for printing logging messages.
- * **parentGuiName** - String, name of the JavaGlade instance that built the parent where the entity will be embedded.
- * **parentPanel** - Composite, parent panel where the entity contents will be embedded.

- *isDetail*

public boolean **isDetail**()

– **Usage**

- * Indicates whether is a main entity or is embedded in another

- *on_entitySelectionButton_clicked*

public void **on_entitySelectionButton_clicked**(Event e)

– **Usage**

- * Handler for the subentity selection buttons (magnifying glass buttons) clicked. Shows the selection dialog for selecting a valid instance of a subentity.

– **Parameters**

- * **e** - Event, magnifying glass button clicked event data.

- *printEntity*

public abstract OperationResult **printEntity**()

– **Usage**

- * Print entity button clicked handler. Builds the report with the data of the selected instance of the entity (selected registry), and allows the user to print the elaborated report.

– **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *printWidgets*

public void **printWidgets**()

- *queryEntity*

public abstract ArrayList **queryEntity**()

– **Usage**

- * Search entity button clicked handler. Searches the system for the entity instances (registries) available.

– **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.

- *queryEntity*

public abstract ArrayList **queryEntity**(java.lang.String fieldId, java.lang.String value)

– **Usage**

- * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria.

– **Parameters**

- * **fieldId** - String, name of the entity field that will be queried.
- * **value** - String, value of the entity field that will be searched for.

– **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.

- *queryEntityWithFormat*

public abstract ArrayList **queryEntityWithFormat**(java.lang.String fieldId, java.lang.String value)

– **Usage**

- * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria and show them (formatted as a list) to the user in the entity's screen.
 - **Parameters**
 - * **fieldId** - String, name of the entity field that will be queried.
 - * **value** - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList<RegistryOT>, list of entity instances (registries) taken as the result of the query operation.

- *querySubentities*

```
public abstract ArrayList querySubentities(
com.imagos.axilice.core.installer.SubentityOT  entData )
```

 - **Usage**
 - * Queries the selected entity for all the detail subentities associated with it.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to query for.
 - **Returns** - ArrayList, list of detail subentity registries assigned to the master detail registry.

- *saveEntity*

```
public abstract OperationResult saveEntity( )
```

 - **Usage**
 - * Save entity button clicked handler. Update the system with the data of the edited entity registry. It's called for saving new entity registries (add a new registry) or for updating the contents of existing entity registries.
 - **Returns** - OperationResult, Result of the operation and list of messages generated through the operation.

- *saveEntityRegistry*

```
public void saveEntityRegistry( )
```
- *selectEntity*

```
public abstract void selectEntity( java.lang.Long  idEntity )
```

 - **Usage**
 - * Marks an instance (registry) of the entity as the selected.
 - **Parameters**
 - * **idEntity** - Long, internal id of the selected instance (registry).

- *selectSubentity*

```
public void selectSubentity( java.lang.String  parentFieldId,
com.imagos.axilice.core.common.BaseVO  subentityOT )
```

 - **Usage**
 - * Updates an entity instance by setting up the selected value for a subentity field.
 - **Parameters**
 - * **parentFieldId** - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * **subentityOT** - BaseVO, subentity's selected instance (registry) value.

- *setEntity*

```
public void setEntity( com.imagos.axilice.core.installer.EntityOT  entity )
```

 - **Usage**
 - * Sets up the entity name.
 - **Parameters**
 - * **name** - String, entity name.

- *setRegistries*

```
public void setRegistries( java.util.ArrayList lRegistries )
```

- Usage

- * Sets the list of registries that are showed in the entity view controlled by the Entity Launcher instance. It's used for detail entities where queries are made in the master entity launcher.

- Parameters

- * lRegistries - ArrayList<BaseVO>, list of shown registries of the entity.

- *showSelectionDialog*

```
public void showSelectionDialog( com.imagos.axilice.core.installer.SubentityOT entData )
```

- Usage

- * Opens the selection dialog that enables the user to select an instance of a subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.

- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT entData, long selEntity, boolean readonly )
```

- Usage

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.
 - * selEntity - long, id of the selected instance of the subentity.
 - * readonly - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT entData, java.lang.String parentGuiName, long selEntity, boolean readonly )
```

- Usage

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.
 - * parentGuiName - String,
 - * selEntity - long, id of the selected instance of the subentity.
 - * readonly - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

- *showSubentitySelectionDialog*

```
public void showSubentitySelectionDialog( com.imagos.axilice.core.installer.SubentityOT entData )
```

- Usage

- * Opens the selection dialog that enables the user to select an instance of a multiple selection detail subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.

- *undoEntityEdition*

```
public void undoEntityEdition( )
```

- *updateEntityData*
`public abstract void updateEntityData()`
 – **Usage**
 * Updates the selected instance (registry) data with the content panel’s data typed by the user.
-
- *viewEntity*
`public abstract OperationResult viewEntity()`
 – **Usage**
 * View entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), in read-only mode.
 – **Returns** - **OperationResult**, result of the delete operation and list of messages generated during the operation.

7.1.6 CLASS ProductStockClient

DECLARATION

```
public class ProductStockClient
extends com.imagos.axilice.client.gui.common.CommonClient
```

FIELDS

- public static final String AXILICE_LOG_EDITSTOCK_GLADEPATH
 – Message for logging the entity glade configuration file path.

CONSTRUCTORS

- *ProductStockClient*
`public ProductStockClient()`

METHODS

- *build_extension_fields_view*
`public void build_extension_fields_view()`
- *buildEntity*
`public AxiliceDetailView buildEntity(SwtWidget parent,
 com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData,
 java.lang.String rootName)`
- *buildSelectionSets*
`public void buildSelectionSets()`
- *clearEntityData*
`public void clearEntityData()`

- *fillEntityData*
public void fillEntityData()
- *getEntityData*
public BaseVO getEntityData()
- *getEntityFields*
public ArrayList getEntityFields()
- *getRevision*
public static String getRevision()
 - Usage
 - * Returns tag Id assigned to CVS source file.
- *getWidgets*
public void getWidgets()
- *init_specific_components*
public void init_specific_components()
- *setEditionMode*
public void setEditionMode(boolean enabled)
- *setWidgetsValues*
public void setWidgetsValues(com.imagos.axilice.core.common.BaseVO entity)
- *updateEntityData*
public void updateEntityData()
- *validateEntity*
public OperationResult validateEntity()

METHODS INHERITED FROM CLASS com.imagos.axilice.client.gui.common.CommonClient

(in 17.1.2, page 402)

- *addAxiliceEventListener*
public void addAxiliceEventListener(com.imagos.axilice.client.gui.AxiliceEventListener l)
- *addEntity*
public ArrayList addEntity()
- *addExistingSubentity*
public void addExistingSubentity(com.imagos.axilice.core.installer.SubentityOT ent)
 - Usage
 - * Shows a dialog for selection of a new subentity
 - Parameters
 - * ent - SubentityOT. Encapsulate data type of subentity
- *addSubentity*
public void addSubentity(com.imagos.axilice.core.common.BaseVO subentity, com.imagos.axilice.core.common.MultiSelectionSubentityOT subE)
 - Usage

- * Adds a existent entity to a multiselection list
 - **Parameters**
 - * **subentity** - Subentity to be added
 - * **subE** - Object that encapsulates information about the subentity

 - *addTrazabilityTree*
public void **addTrazabilityTree**(TabFolder tabFolder)
 - *build_extension_fields_view*
public abstract void **build_extension_fields_view**()
 - **Usage**
 - * Build the view of the extension fields.
- It must implement only those clients who require it
-
- *buildEntity*
public AxiliceEntityView **buildEntity**()
- *buildEntityDetail*
public AxiliceEntityView **buildEntityDetail**(
com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData)
- *buildEntitySelectionView*
public AxiliceEntitySelectionView **buildEntitySelectionView**()
- **Usage**
 - * Build view of selection entities
- **Returns** - AxilicePartnerSelectionView. SelectionView created
-
- *buildSelectionSets*
public abstract void **buildSelectionSets**()
- **Usage**
 - * Build 2 sets of SubentityOT: lSelectionSubentities and lMultiSelectionSubentities
- These sets are used to connect the widgets with the handlers for the selection buttons and the multiselection panels
- It must implement only those clients who require it
-
- *buildSubentityMultiSelection*
public void **buildSubentityMultiSelection**()
- *clearEntityData*
public void **clearEntityData**()
- *clearParentField*
public OperationResult **clearParentField**(java.util.ArrayList selRegistries,
java.lang.String parentField)
- **Usage**
 - * Clears the value of the field indicated
- **Parameters**
 - * **selRegistries** - ArrayList. Index list of records to process
 - * **parentField** - Name of the field to clear
- **Returns** - OperationResult. Result of operation
-
- *delEntity*
public OperationResult **delEntity**()
- *delEntity*
public OperationResult **delEntity**(java.util.ArrayList selRegistries)

- *delEntity*
 public OperationResult **delEntity**(com.imagos.axilice.core.common.BaseVO **b**)

 - **Usage**
 * Delete a entity from database
 - **Parameters**
 * **b** - BaseVO. Entity to delete
 - **Returns** - OperationResult. Result of the operation

- *delSubentities*
 public OperationResult **delSubentities**(java.util.ArrayList **selRegistries**)

 - **Usage**
 * Delete a set of subentities from record list of controller, not from database

 Set the parent field of subentity to null
 - **Parameters**
 * **selRegistries** - List of index to delete
 - **Returns** - OperationResult. Result of the operation

- *editEntity*
 public OperationResult **editEntity**()

- *fillEntityData*
 public void **fillEntityData**()

- *fillEntityExtensionTable*
 public void **fillEntityExtensionTable**(java.lang.Long **id**)

 - **Usage**
 * Retrieve extension field values from the server for a given entity and displays them in the view
 - **Parameters**
 * **id** - Long. Key of the entity whose values are to be displayed

- *fillEntityTable*
 public void **fillEntityTable**()

 - **Usage**
 * Load records from database and fill the entity view table and controller record list

- *fillEntityTable*
 public void **fillEntityTable**(java.util.ArrayList **lR**)

 - **Usage**
 * Fill the entity view table with the records given
 - **Parameters**
 * **lR** - ArrayList. Records to be displayed

- *getClassEntity*
 public Class **getClassEntity**()

 - **Returns** - Class of the entity for which the controller is built

- *getDialogName*
 public String **getDialogName**()

- *getEditGladePath*
 public String **getEditGladePath**()

- *getEditionGui*
public Gui **getEditionGui**()
 - **Usage**
 - * Returns the edition Gui instance.
 - **Returns** - Gui, entity edition Gui.

- *getEditWindow*
public SwtTopLevelWindow **getEditWindow**()
- *getEntityData*
public abstract BaseV0 **getEntityData**()
- *getEntityFields*
public ArrayList **getEntityFields**()
- *getLogger*
public Logger **getLogger**()
- *getParentEntityLauncher*
public EntityLauncher **getParentEntityLauncher**()
 - **Returns** - EntityLauncher of parent entity if is a embedded entity or null otherwise

- *getRegistries*
public Collection **getRegistries**()
 - **Returns** - lRegistries

- *getRegistries*
public List **getRegistries**(java.util.ArrayList lIds)
 - **Usage**
 - * Get a list of entities stored by the controller
 - **Parameters**
 - * lIds - Index list
 - **Returns** - List of entities

- *getView*
public AxiliceEntityView **getView**()
 - **Returns** - AxiliceEntityView. Main view or Detail view if is a embedded entity.

- *getWidgets*
public abstract void **getWidgets**()
 - **Usage**
 - * Get the widgets from the glade and stored in local variables.
 - Must be overwritten

- *init_specific_components*
public abstract void **init_specific_components**()
 - **Usage**
 - * Initializes the specific components of the client

- *printEntity*
public OperationResult **printEntity**()
- *printEntity*
public OperationResult **printEntity**(java.lang.String reportPath, java.util.HashMap parameters)

- *queryEntity*
public ArrayList **queryEntity**()
- *queryEntity*
public ArrayList **queryEntity**(java.lang.String fieldId, java.lang.String value)
- *queryEntityWithFormat*
public ArrayList **queryEntityWithFormat**(java.lang.String fieldId, java.lang.String value)
- *queryFreeEntities*
public ArrayList **queryFreeEntities**(com.imagos.axilice.core.common.MultiSelectionSubentityOT mSubE)
 - **Usage**
 - * Search entities not associated with a parent entity (null parent field)
 - **Parameters**
 - * mSubE - SubentityOT for the parent entity that must be null
 - **Returns** - ArrayList. List of entities found
- *querySubentities*
public ArrayList **querySubentities**(com.imagos.axilice.core.installer.SubentityOT entData)
- *refreshEntity*
public OperationResult **refreshEntity**()
- *refreshEntity*
public BaseVO **refreshEntity**(com.imagos.axilice.core.common.BaseVO entity)
- *refreshParentSelectedEntity*
public OperationResult **refreshParentSelectedEntity**()
 - **Usage**
 - * Reload from database the value of the parent entity of the selected entity
 - **Returns** - OperationResult. The result of the operation.
- *removeRegistries*
public OperationResult **removeRegistries**(java.util.ArrayList selRegistries, java.lang.String parentField)
 - **Usage**
 - * Deletes a list of the records stored by the controller. Also clean your parent fields and updates the database
 - **Parameters**
 - * selRegistries - Index list to remove
 - * parentField - Field name that links to the parent entity
 - **Returns** - OperationResult. Result of the operation
- *saveEntity*
public OperationResult **saveEntity**()
- *saveEntity*
public OperationResult **saveEntity**(java.util.ArrayList lIds)
- *saveEntity*
public OperationResult **saveEntity**(com.imagos.axilice.core.common.BaseVO b)
 - **Usage**
 - * Save or update a entity to database
 - **Parameters**
 - * b - BaseVO. Entity to save or update
 - **Returns** - OperationResult. Result of the operation

- *selectEntity*
public void selectEntity(java.lang.Long idEntity)
- *selectSubentity*
public void selectSubentity(java.lang.String parentFieldId, com.imagos.axilice.core.common.BaseVO subentityOT)
- *selectTreeItem*
public void selectTreeItem(java.lang.Long id)
 - **Usage**
 - * Select a item in the entity tree. No me gusta esto aqui
 - **Parameters**
 - * id - Long. Index of item to select
- *setDialogName*
public void setDialogName(java.lang.String dialogName)
- *setEditGladePath*
public void setEditGladePath(java.lang.String editGladePath)
- *setEditionMode*
public abstract void setEditionMode(boolean enabled)
- *setRegistries*
public void setRegistries(java.util.Collection collection)
 - **Usage**
 - * Setter for lRegistries
- *setWidgetsValues*
public abstract void setWidgetsValues(com.imagos.axilice.core.common.BaseVO entity)
 - **Usage**
 - * Set the values of the widgets on the edit dialog.
 - Must be overwritten.
 - **Parameters**
 - * entity - BaseVO. Entity that contains the values to be displayed
- *showErrorMessages*
public void showErrorMessages(com.imagos.axilice.core.util.OperationResult result)
- *updateEntityData*
public void updateEntityData()
- *updateRegistry*
public void updateRegistry(com.imagos.axilice.core.common.BaseVO ent)
 - **Usage**
 - * Add or update an entity (to the controller records)
 - **Parameters**
 - * ent - Entity to add or update
- *updateViews*
public void updateViews(boolean updateFromParent)
 - **Usage**
 - * Force the update of the views associated
 - **Parameters**
 - * updateFromParent - Indicates whether the view must be updated by reading the values from the parent entity. takes effect only when the entity is detail (is embedded in another)

-
- *validateEntity*
 public abstract **OperationResult validateEntity()**
 – **Usage**
 * Validate the values of the widgets in the edit dialog.
 Must be overwritten.
 – **Returns** - **OperationResult**. Result of the validation.
-
- *viewEntity*
 public **OperationResult viewEntity()**

METHODS INHERITED FROM CLASS com.imagos.axilice.core.installer.EntityLauncher

(in 24.2.2, page 532)

- *addEntity*
 public abstract **ArrayList addEntity()**
 – **Usage**
 * Add entity button clicked handler. Show the entity content screen ready to start inserting data for a new instance of the entity (new registry).
 – **Returns** - **ArrayList<ResultMessage>**, list of errors if any has taken place, null in any other case.
-
- *buildEditEntityGUI*
 public void **buildEditEntityGUI()**
 – **Usage**
 * Builds the edit entity dialog window from the corresponding glade interface file. Stores the entity dialog instance for posterior accesses.
-
- *buildEntity*
 public **AxiliceEntityView buildEntity()**
 – **Usage**
 * Builds the entity GUI and show it on screen to the user.
-
- *buildEntity*
 public **AxiliceDetailEntityView buildEntity(SwtWidget parent, com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData, java.lang.String rootName)**
 – **Usage**
 * Builds the entity GUI and show it on screen to the user. The entity GUI will be embedded into the parent composite widget.
 – **Parameters**
 * **parent** - Composite, parent holder for embedding the entity view.
 * **subentityData** - SubentityOT, data of the subentity to build if the entity launcher will be used to handle a multi-selection detail entity view.
 * **rootName** - String, name of the root widget that will be embedded into the master entity edition panel.
 – **Returns** - **AxiliceDetailEntityView**, the entity detail view built.
-
- *cancelEntityEdition*
 public void **cancelEntityEdition()**
-
- *clearEntityData*
 public abstract void **clearEntityData()**

- **Usage**
 - * Clears the entity content panel by resetting the content widgets.

- *delEntity*

```
public abstract OperationResult delEntity( )
```

 - **Usage**
 - * Delete entity button clicked handler. Deletes the selected instance of the entity (selected registry) from the database.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *delEntity*

```
public abstract OperationResult delEntity( java.util.ArrayList lSelRegistries )
```

 - **Usage**
 - * Delete entity button clicked handler. Deletes the selected instances of the entity (selected registries) from the database. Its a multiple delete method.
 - **Parameters**
 - * lSelRegistries - ArrayList<Long>, list of selected instances of the entity (selected registries). The list will contain only the internal id of the selected registries.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *deselectSubentity*

```
public void deselectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

 - **Usage**
 - * Updates an entity instance by removing the selected value for a subentity field.
 - **Parameters**
 - * parentFieldId - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * subentityOT - BaseVO, subentity's deselected instance (registry) value.

- *editEntity*

```
public abstract OperationResult editEntity( )
```

 - **Usage**
 - * Edit entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), and allows the user to modify any content of the entity registry.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *fillEntityData*

```
public abstract void fillEntityData( )
```

 - **Usage**
 - * Fills the entity content panel with the data of the selected instance (registry) of the entity.

- *fireExceptionAndExit*

```
public void fireExceptionAndExit( java.lang.String msg )
```

 - **Usage**
 - * Fire a exception and stop the program
 - **Parameters**
 - * msg -

- *formatRegistries*
public ArrayList formatRegistries()
- *formatRegistries*
public ArrayList formatRegistries(java.util.ArrayList lRegistries)
- *getDialog*
public SwtDialog getDialog(java.lang.String dialogName)
- *getEntity*
public EntityOT getEntity()
 - **Usage**
 - * Gets the entity data.
 - **Returns** - EntityOT, entity data.

- *getEntityData*
public abstract BaseVO getEntityData()
 - **Usage**
 - * Gets the content panel's data typed by the user formatted as an entity VO object.

- *getEntityEditViewName*
public String getEntityEditViewName(java.lang.String parentGuiName)
- *getEntityFields*
public abstract ArrayList getEntityFields()
 - **Usage**
 - * Gets the list of defined fields for an entity
 - **Returns** - ArrayList, list of entity fields.

- *getEntityViewName*
public String getEntityViewName(java.lang.String parentGuiName)
- *getLRegistries*
public ArrayList getLRegistries()
 - **Usage**
 - * Getter for lRegistries

- *getRegistry*
public BaseVO getRegistry(long id)
- *getRevision*
public static String getRevision()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.

- *getSelectedEntity*
public BaseVO getSelectedEntity()
 - **Usage**
 - * Gets the selected instance (registry) of the entity.
 - **Returns** - BaseVO, selected instance (registry) of the entity.

- *getSelectedId*
public Long getSelectedId()
 - **Usage**
 - * Gets the internal id of the selected entity registry.
 - **Returns** - Long, internal id of the selected entity registry, null in case no entity registry has been selected yet.

- *getSwtWidget*
public SwtWidget **getSwtWidget**(java.lang.String widgetName)
- *getWidget*
public Widget **getWidget**(java.lang.String widgetName)
- *initEntityLauncher*
public void **initEntityLauncher**(Logger logger, Gui parentGui, SwtWidget parentPanel)
 - **Usage**
 - * Initiates the entity launcher context and prepares the launcher to be started.
 - **Parameters**
 - * **logger** - Logger, logger instance for printing logging messages.
 - * **parentGuiName** - String, name of the JavaGlade instance that built the parent where the entity will be embedded.
 - * **parentPanel** - Composite, parent panel where the entity contents will be embedded.
- *isDetail*
public boolean **isDetail**()
 - **Usage**
 - * Indicates whether is a main entity or is embedded in another
- *on_entitySelectionButton_clicked*
public void **on_entitySelectionButton_clicked**(Event e)
 - **Usage**
 - * Handler for the subentity selection buttons (magnifying glass buttons) clicked. Shows the selection dialog for selecting a valid instance of a subentity.
 - **Parameters**
 - * **e** - Event, magnifying glass button clicked event data.
- *printEntity*
public abstract OperationResult **printEntity**()
 - **Usage**
 - * Print entity button clicked handler. Builds the report with the data of the selected instance of the entity (selected registry), and allows the user to print the elaborated report.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.
- *printWidgets*
public void **printWidgets**()
- *queryEntity*
public abstract ArrayList **queryEntity**()
 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) available.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.
- *queryEntity*
public abstract ArrayList **queryEntity**(java.lang.String fieldId, java.lang.String value)
 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria.
 - **Parameters**

- * **fieldId** - String, name of the entity field that will be queried.
 - * **value** - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.

- *queryEntityWithFormat*

```
public abstract ArrayList queryEntityWithFormat( java.lang.String fieldId,
java.lang.String value )
```

 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria and show them (formatted as a list) to the user in the entity's screen.
 - **Parameters**
 - * **fieldId** - String, name of the entity field that will be queried.
 - * **value** - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList<RegistryOT>, list of entity instances (registries) taken as the result of the query operation.

- *querySubentities*

```
public abstract ArrayList querySubentities(
com.imagos.axilice.core.installer.SubentityOT entData )
```

 - **Usage**
 - * Queries the selected entity for all the detail subentities associated with it.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to query for.
 - **Returns** - ArrayList, list of detail subentity registries assigned to the master detail registry.

- *saveEntity*

```
public abstract OperationResult saveEntity( )
```

 - **Usage**
 - * Save entity button clicked handler. Update the system with the data of the edited entity registry. It's called for saving new entity registries (add a new registry) or for updating the contents of existing entity registries.
 - **Returns** - OperationResult, Result of the operation and list of messages generated through the operation.

- *saveEntityRegistry*

```
public void saveEntityRegistry( )
```
- *selectEntity*

```
public abstract void selectEntity( java.lang.Long idEntity )
```

 - **Usage**
 - * Marks an instance (registry) of the entity as the selected.
 - **Parameters**
 - * **idEntity** - Long, internal id of the selected instance (registry).

- *selectSubentity*

```
public void selectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

 - **Usage**
 - * Updates an entity instance by setting up the selected value for a subentity field.
 - **Parameters**
 - * **parentFieldId** - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.

* subentityOT - BaseVO, subentity's selected instance (registry) value.

- *setEntity*

public void **setEntity**(com.imagos.axilice.core.installer.EntityOT entity)

- Usage

- * Sets up the entity name.

- Parameters

- * name - String, entity name.

- *setRegistries*

public void **setRegistries**(java.util.ArrayList lRegistries)

- Usage

- * Sets the list of registries that are showed in the entity view controlled by the Entity Launcher instance. It's used for detail entities where queries are made in the master entity launcher.

- Parameters

- * lRegistries - ArrayList<BaseVO>, list of shown registries of the entity.

- *showSelectionDialog*

public void **showSelectionDialog**(com.imagos.axilice.core.installer.SubentityOT entData)

- Usage

- * Opens the selection dialog that enables the user to select an instance of a subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.

- *showSubentityDialog*

public void **showSubentityDialog**(com.imagos.axilice.core.installer.SubentityOT entData, long selEntity, boolean readonly)

- Usage

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.
 - * selEntity - long, id of the selected instance of the subentity.
 - * readonly - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

- *showSubentityDialog*

public void **showSubentityDialog**(com.imagos.axilice.core.installer.SubentityOT entData, java.lang.String parentGuiName, long selEntity, boolean readonly)

- Usage

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.
 - * parentGuiName - String,
 - * selEntity - long, id of the selected instance of the subentity.
 - * readonly - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

- *showSubentitySelectionDialog*

public void **showSubentitySelectionDialog**(
com.imagos.axilice.core.installer.SubentityOT entData)

- **Usage**
 - * Opens the selection dialog that enables the user to select an instance of a multiple selection detail subentity.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to select.
-
- *undoEntityEdition*
public void undoEntityEdition()
 - *updateEntityData*
public abstract void updateEntityData()
 - **Usage**
 - * Updates the selected instance (registry) data with the content panel’s data typed by the user.
-
- *viewEntity*
public abstract OperationResult viewEntity()
 - **Usage**
 - * View entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), in read-only mode.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

7.1.7 CLASS StockClient

DECLARATION

```
public class StockClient
extends com.imagos.axilice.client.gui.common.CommonClient
```

FIELDS

- public static final String AXILICE_LOG_EDITSTOCK_GLADEPATH
 - Message for logging the entity glade configuration file path.

CONSTRUCTORS

- *StockClient*
public StockClient()

METHODS

- *build_extension_fields_view*
public void **build_extension_fields_view**()

- *buildSelectionSets*
public void **buildSelectionSets**()

- *clearEntityData*
public void **clearEntityData**()

- *fillEntityData*
public void **fillEntityData**()

- *getEntityData*
public BaseVO **getEntityData**()

- *getEntityFields*
public ArrayList **getEntityFields**()

- *getLotDescription*
public String **getLotDescription**()

- *getRevision*
public static String **getRevision**()

- Usage

- * Returns tag Id assigned to CVS source file.

- *getWarehouseDescription*
public String **getWarehouseDescription**()

- *getWidgets*
public void **getWidgets**()

- *init_specific_components*
public void **init_specific_components**()

- *setEditionMode*
public void **setEditionMode**(boolean enabled)

- *setWidgetsValues*
public void **setWidgetsValues**(com.imagos.axilice.core.common.BaseVO entity)

- *updateEntityData*
public void **updateEntityData**()

- *validateEntity*
public OperationResult **validateEntity**()

METHODS INHERITED FROM CLASS com.imagos.axilice.client.gui.common.CommonClient

(in 17.1.2, page 402)

- *addAxiliceEventListener*
 public void **addAxiliceEventListener**(
com.imagos.axilice.client.gui.AxiliceEventListener l)
- *addEntity*
 public ArrayList **addEntity**()
- *addExistingSubentity*
 public void **addExistingSubentity**(com.imagos.axilice.core.installer.SubentityOT
 ent)
 - Usage
 - * Shows a dialog for selection of a new subentity
 - Parameters
 - * ent - SubentityOT. Encapsulate data type of subentity
- *addSubentity*
 public void **addSubentity**(com.imagos.axilice.core.common.BaseVO subentity,
 com.imagos.axilice.core.common.MultiSelectionSubentityOT subE)
 - Usage
 - * Adds a existent entity to a multiselection list
 - Parameters
 - * subentity - Subentity to be added
 - * subE - Object that encapsulates information about the subentity
- *addTrazabilityTree*
 public void **addTrazabilityTree**(TabFolder tabFolder)
- *build_extension_fields_view*
 public abstract void **build_extension_fields_view**()
 - Usage
 - * Build the view of the extension fields.

It must implement only those clients who require it
- *buildEntity*
 public AxiliceEntityView **buildEntity**()
- *buildEntityDetail*
 public AxiliceEntityView **buildEntityDetail**(
com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData)
- *buildEntitySelectionView*
 public AxiliceEntitySelectionView **buildEntitySelectionView**()
 - Usage
 - * Build view of selection entities
 - Returns - AxilicePartnerSelectionView. SelectionView created
- *buildSelectionSets*
 public abstract void **buildSelectionSets**()
 - Usage

* Build 2 sets of SubentityOT: lSelectionSubentities and lMultiSelectionSubentities

These sets are used to connect the widgets with the handlers for the selection buttons and the multiselection panels

It must implement only those clients who require it

- *buildSubentityMultiSelection*
public void buildSubentityMultiSelection()
- *clearEntityData*
public void clearEntityData()
- *clearParentField*
public OperationResult clearParentField(java.util.ArrayList selRegistries, java.lang.String parentField)
 - Usage
 - * Clears the value of the field indicated
 - Parameters
 - * selRegistries - ArrayList. Index list of records to process
 - * parentField - Name of the field to clear
 - Returns - OperationResult. Result of operation
- *delEntity*
public OperationResult delEntity()
- *delEntity*
public OperationResult delEntity(java.util.ArrayList selRegistries)
- *delEntity*
public OperationResult delEntity(com.imagos.axilice.core.common.BaseVO b)
 - Usage
 - * Delete a entity from database
 - Parameters
 - * b - BaseVO. Entity to delete
 - Returns - OperationResult. Result of the operation
- *delSubentities*
public OperationResult delSubentities(java.util.ArrayList selRegistries)
 - Usage
 - * Delete a set of subentities from record list of controller, not from database
 - Set the parent field of subentity to null
 - Parameters
 - * selRegistries - List of index to delete
 - Returns - OperationResult. Result of the operation
- *editEntity*
public OperationResult editEntity()
- *fillEntityData*
public void fillEntityData()
- *fillEntityExtensionTable*
public void fillEntityExtensionTable(java.lang.Long id)
 - Usage
 - * Retrieve extension field values from the server for a given entity and displays them in the view

- **Parameters**
 - * id - Long. Key of the entity whose values are to be displayed

- *fillEntityTable*
public void fillEntityTable()
 - **Usage**
 - * Load records from database and fill the entity view table and controller record list

- *fillEntityTable*
public void fillEntityTable(java.util.ArrayList lR)
 - **Usage**
 - * Fill the entity view table with the records given
 - **Parameters**
 - * lR - ArrayList. Records to be displayed

- *getClassEntity*
public Class getClassEntity()
 - **Returns** - Class of the entity for which the controller is built

- *getDialogName*
public String getDialogName()

- *getEditGladePath*
public String getEditGladePath()

- *getEditionGui*
public Gui getEditionGui()
 - **Usage**
 - * Returns the edition Gui instance.
 - **Returns** - Gui, entity edition Gui.

- *getEditWindow*
public SwtTopLevelWindow getEditWindow()

- *getEntityData*
public abstract BaseVO getEntityData()

- *getEntityFields*
public ArrayList getEntityFields()

- *getLogger*
public Logger getLogger()

- *getParentEntityLauncher*
public EntityLauncher getParentEntityLauncher()
 - **Returns** - EntityLauncher of parent entity if is a embedded entity or null otherwise

- *getRegistries*
public Collection getRegistries()
 - **Returns** - lRegistries

- *getRegistries*
public List getRegistries(java.util.ArrayList lIds)
 - **Usage**
 - * Get a list of entities stored by the controller
 - **Parameters**
 - * lIds - Index list
 - **Returns** - List of entities

- *getView*
 public AxiliceEntityView **getView**()
 – **Returns** - AxiliceEntityView. Main view or Detail view if is a embedded entity.

- *getWidgets*
 public abstract void **getWidgets**()
 – **Usage**
 * Get the widgets from the glade and stored in local variables.
 Must be overwritten

- *init_specific_components*
 public abstract void **init_specific_components**()
 – **Usage**
 * Initializes the specific components of the client

- *printEntity*
 public OperationResult **printEntity**()

- *printEntity*
 public OperationResult **printEntity**(java.lang.String reportPath, java.util.HashMap parameters)

- *queryEntity*
 public ArrayList **queryEntity**()

- *queryEntity*
 public ArrayList **queryEntity**(java.lang.String fieldId, java.lang.String value)

- *queryEntityWithFormat*
 public ArrayList **queryEntityWithFormat**(java.lang.String fieldId, java.lang.String value)

- *queryFreeEntities*
 public ArrayList **queryFreeEntities**(com.imagos.axilice.core.common.MultiSelectionSubentityOT mSubE)
 – **Usage**
 * Search entities not associated with a parent entity (null parent field)
 – **Parameters**
 * mSubE - SubentityOT for the parent entity that must be null
 – **Returns** - ArrayList. List of entities found

- *querySubentities*
 public ArrayList **querySubentities**(com.imagos.axilice.core.installer.SubentityOT entData)

- *refreshEntity*
 public OperationResult **refreshEntity**()

- *refreshEntity*
 public BaseVO **refreshEntity**(com.imagos.axilice.core.common.BaseVO entity)

- *refreshParentSelectedEntity*
 public OperationResult **refreshParentSelectedEntity**()
 – **Usage**
 * Reload from database the value of the parent entity of the selected entity
 – **Returns** - OperationResult. The result of the operation.

- *removeRegistries*
 public OperationResult **removeRegistries**(java.util.ArrayList selRegistries, java.lang.String parentField)

- **Usage**
 - * Deletes a list of the records stored by the controller. Also clean your parent fields and updates the database
 - **Parameters**
 - * **selRegistries** - Index list to remove
 - * **parentField** - Field name that links to the parent entity
 - **Returns** - **OperationResult**. Result of the operation
-
- *saveEntity*
public **OperationResult** **saveEntity**()
 - *saveEntity*
public **OperationResult** **saveEntity**(java.util.ArrayList lIds)
 - *saveEntity*
public **OperationResult** **saveEntity**(com.imagos.axilice.core.common.BaseVO b)
- **Usage**
 - * Save or update a entity to database
 - **Parameters**
 - * **b** - BaseVO. Entity to save or update
 - **Returns** - **OperationResult**. Result of the operation
-
- *selectEntity*
public void **selectEntity**(java.lang.Long idEntity)
 - *selectSubentity*
public void **selectSubentity**(java.lang.String parentFieldId, com.imagos.axilice.core.common.BaseVO subentityOT)
 - *selectTreeItem*
public void **selectTreeItem**(java.lang.Long id)
- **Usage**
 - * Select a item in the entity tree. No me gusta esto aqui
 - **Parameters**
 - * **id** - Long. Index of item to select
-
- *setDialogName*
public void **setDialogName**(java.lang.String dialogName)
 - *setEditGladePath*
public void **setEditGladePath**(java.lang.String editGladePath)
 - *setEditionMode*
public abstract void **setEditionMode**(boolean enabled)
 - *setRegistries*
public void **setRegistries**(java.util.Collection collection)
- **Usage**
 - * Setter for lRegistries
-
- *setWidgetsValues*
public abstract void **setWidgetsValues**(com.imagos.axilice.core.common.BaseVO entity)
- **Usage**
 - * Set the values of the widgets on the edit dialog.
 - **Parameters**
 - * **entity** - BaseVO. Entity that contains the values to be displayed

-
- *showErrorMessage*
 public void **showErrorMessage**(com.imagos.axilice.core.util.OperationResult result)

 - *updateEntityData*
 public void **updateEntityData**()

 - *updateRegistry*
 public void **updateRegistry**(com.imagos.axilice.core.common.BaseVO ent)
 - **Usage**
 - * Add or update an entity (to the controller records)
 - **Parameters**
 - * ent - Entity to add or update

 - *updateViews*
 public void **updateViews**(boolean updateFromParent)
 - **Usage**
 - * Force the update of the views associated
 - **Parameters**
 - * updateFromParent - Indicates whether the view must be updated by reading the values from the parent entity. takes effect only when the entity is detail (is embedded in another)

 - *validateEntity*
 public abstract OperationResult **validateEntity**()
 - **Usage**
 - * Validate the values of the widgets in the edit dialog.
 - **Returns** - OperationResult. Result of the validation.

 - *viewEntity*
 public OperationResult **viewEntity**()

METHODS INHERITED FROM CLASS com.imagos.axilice.core.installer.EntityLauncher

(in 24.2.2, page 532)

- *addEntity*
 public abstract ArrayList **addEntity**()
 - **Usage**
 - * Add entity button clicked handler. Show the entity content screen ready to start inserting data for a new instance of the entity (new registry).
 - **Returns** - ArrayList<ResultMessage>, list of errors if any has taken place, null in any other case.

- *buildEditEntityGUI*
 public void **buildEditEntityGUI**()
 - **Usage**
 - * Builds the edit entity dialog window from the corresponding glade interface file. Stores the entity dialog instance for posterior accesses.

- *buildEntity*
 public AxiliceEntityView **buildEntity**()
 - **Usage**

* Builds the entity GUI and show it on screen to the user.

- *buildEntity*

```
public AxiliceDetailView buildEntity( SwtWidget parent,
com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData,
java.lang.String rootName )
```

- **Usage**

- * Builds the entity GUI and show it on screen to the user. The entity GUI will be embedded into the parent composite widget.

- **Parameters**

- * **parent** - Composite, parent holder for embedding the entity view.
 - * **subentityData** - SubentityOT, data of the subentity to build if the entity launcher will be used to handle a multi-selection detail entity view.
 - * **rootName** - String, name of the root widget that will be embedded into the master entity edition panel.

- **Returns** - AxiliceDetailView, the entity detail view built.

- *cancelEntityEdition*

```
public void cancelEntityEdition( )
```

- *clearEntityData*

```
public abstract void clearEntityData( )
```

- **Usage**

- * Clears the entity content panel by resetting the content widgets.

- *delEntity*

```
public abstract OperationResult delEntity( )
```

- **Usage**

- * Delete entity button clicked handler. Deletes the selected instance of the entity (selected registry) from the database.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *delEntity*

```
public abstract OperationResult delEntity( java.util.ArrayList lSelRegistries )
```

- **Usage**

- * Delete entity button clicked handler. Deletes the selected instances of the entity (selected registries) from the database. Its a multiple delete method.

- **Parameters**

- * **lSelRegistries** - ArrayList<Long>, list of selected instances of the entity (selected registries). The list will contain only the internal id of the selected registries.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *deselectSubentity*

```
public void deselectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

- **Usage**

- * Updates an entity instance by removing the selected value for a subentity field.

- **Parameters**

- * **parentFieldId** - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * **subentityOT** - BaseVO, subentity's deselected instance (registry) value.

- *editEntity*
 public abstract **OperationResult editEntity()**
 - **Usage**
 - * Edit entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), and allows the user to modify any content of the entity registry.
 - **Returns** - **OperationResult**, result of the delete operation and list of messages generated during the operation.

- *fillEntityData*
 public abstract void **fillEntityData()**
 - **Usage**
 - * Fills the entity content panel with the data of the selected instance (registry) of the entity.

- *fireExceptionAndExit*
 public void **fireExceptionAndExit(java.lang.String msg)**
 - **Usage**
 - * Fire a exception and stop the program
 - **Parameters**
 - * **msg** -

- *formatRegistries*
 public **ArrayList formatRegistries()**

- *formatRegistries*
 public **ArrayList formatRegistries(java.util.ArrayList lRegistries)**

- *getDialog*
 public **SwtDialog getDialog(java.lang.String dialogName)**

- *getEntity*
 public **EntityOT getEntity()**
 - **Usage**
 - * Gets the entity data.
 - **Returns** - **EntityOT**, entity data.

- *getEntityData*
 public abstract **BaseVO getEntityData()**
 - **Usage**
 - * Gets the content panel's data typed by the user formatted as an entity VO object.

- *getEntityEditViewName*
 public **String getEntityEditViewName(java.lang.String parentGuiName)**

- *getEntityFields*
 public abstract **ArrayList getEntityFields()**
 - **Usage**
 - * Gets the list of defined fields for an entity
 - **Returns** - **ArrayList**, list of entity fields.

- *getEntityViewName*
 public **String getEntityViewName(java.lang.String parentGuiName)**

- *getLRegistries*
 public **ArrayList getLRegistries()**
 - **Usage**
 - * Getter for lRegistries

-
- *getRegistry*
`public BaseVO getRegistry(long id)`
 - *getRevision*
`public static String getRevision()`
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
-
- *getSelectedEntity*
`public BaseVO getSelectedEntity()`
 - **Usage**
 - * Gets the selected instance (registry) of the entity.
 - **Returns** - BaseVO, selected instance (registry) of the entity.
-
- *getSelectedId*
`public Long getSelectedId()`
 - **Usage**
 - * Gets the internal id of the selected entity registry.
 - **Returns** - Long, internal id of the selected entity registry, null in case no entity registry has been selected yet.
-
- *getSwtWidget*
`public SwtWidget getSwtWidget(java.lang.String widgetName)`
 - *getWidget*
`public Widget getWidget(java.lang.String widgetName)`
 - *initEntityLauncher*
`public void initEntityLauncher(Logger logger, Gui parentGui, SwtWidget parentPanel)`
 - **Usage**
 - * Inits the entity launcher context and prepares the launcher to be started.
 - **Parameters**
 - * **logger** - Logger, logger instance for printing logging messages.
 - * **parentGuiName** - String, name of the JavaGlade instance that built the parent where the entity will be embedded.
 - * **parentPanel** - Composite, parent panel where the entity contents will be embedded.
-
- *isDetail*
`public boolean isDetail()`
 - **Usage**
 - * Indicates whether is a main entity or is embedded in another
-
- *on_entitySelectionButton_clicked*
`public void on_entitySelectionButton_clicked(Event e)`
 - **Usage**
 - * Handler for the subentity selection buttons (magnifying glass buttons) clicked. Shows the selection dialog for selecting a valid instance of a subentity.
 - **Parameters**
 - * **e** - Event, magnifying glass button clicked event data.
-
- *printEntity*
`public abstract OperationResult printEntity()`
 - **Usage**

- * Print entity button clicked handler. Builds the report with the data of the selected instance of the entity (selected registry), and allows the user to print the elaborated report.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *printWidgets*
public void printWidgets()

- *queryEntity*
public abstract ArrayList queryEntity()
 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) available.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.

- *queryEntity*
public abstract ArrayList queryEntity(java.lang.String fieldId, java.lang.String value)
 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria.
 - **Parameters**
 - * **fieldId** - String, name of the entity field that will be queried.
 - * **value** - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.

- *queryEntityWithFormat*
public abstract ArrayList queryEntityWithFormat(java.lang.String fieldId, java.lang.String value)
 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria and show them (formatted as a list) to the user in the entity's screen.
 - **Parameters**
 - * **fieldId** - String, name of the entity field that will be queried.
 - * **value** - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList<RegistryOT>, list of entity instances (registries) taken as the result of the query operation.

- *querySubentities*
public abstract ArrayList querySubentities(com.imagos.axilice.core.installer.SubentityOT entData)
 - **Usage**
 - * Queries the selected entity for all the detail subentities associated with it.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to query for.
 - **Returns** - ArrayList, list of detail subentity registries assigned to the master detail registry.

- *saveEntity*
public abstract OperationResult saveEntity()
 - **Usage**

- * Save entity button clicked handler. Update the system with the data of the edited entity registry. It's called for saving new entity registries (add a new registry) or for updating the contents of existing entity registries.
 - **Returns** - OperationResult, Result of the operation and list of messages generated through the operation.

- *saveEntityRegistry*
public void saveEntityRegistry()
- *selectEntity*
public abstract void selectEntity(java.lang.Long idEntity)
 - **Usage**
 - * Marks an instance (registry) of the entity as the selected.
 - **Parameters**
 - * idEntity - Long, internal id of the selected instance (registry).

- *selectSubentity*
public void selectSubentity(java.lang.String parentFieldId, com.imagos.axilice.core.common.BaseVO subentityOT)
 - **Usage**
 - * Updates an entity instance by setting up the selected value for a subentity field.
 - **Parameters**
 - * parentFieldId - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * subentityOT - BaseVO, subentity's selected instance (registry) value.

- *setEntity*
public void setEntity(com.imagos.axilice.core.installer.EntityOT entity)
 - **Usage**
 - * Sets up the entity name.
 - **Parameters**
 - * name - String, entity name.

- *setRegistries*
public void setRegistries(java.util.ArrayList lRegistries)
 - **Usage**
 - * Sets the list of registries that are showed in the entity view controlled by the Entity Launcher instance. It's used for detail entities where queries are made in the master entity launcher.
 - **Parameters**
 - * lRegistries - ArrayList<BaseVO>, list of shown registries of the entity.

- *showSelectionDialog*
public void showSelectionDialog(com.imagos.axilice.core.installer.SubentityOT entData)
 - **Usage**
 - * Opens the selection dialog that enables the user to select an instance of a subentity.
 - **Parameters**
 - * entData - SubentityOT, data of the detail subentity to select.

- *showSubentityDialog*
public void showSubentityDialog(com.imagos.axilice.core.installer.SubentityOT entData, long selEntity, boolean readonly)
 - **Usage**

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

– **Parameters**

- * **entData** - SubentityOT, data of the detail subentity to select.
- * **selEntity** - long, id of the selected instance of the subentity.
- * **readonly** - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

• *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT
entData, java.lang.String parentGuiName, long selEntity, boolean readonly )
```

– **Usage**

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

– **Parameters**

- * **entData** - SubentityOT, data of the detail subentity to select.
- * **parentGuiName** - String,
- * **selEntity** - long, id of the selected instance of the subentity.
- * **readonly** - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

• *showSubentitySelectionDialog*

```
public void showSubentitySelectionDialog(
com.imagos.axilice.core.installer.SubentityOT entData )
```

– **Usage**

- * Opens the selection dialog that enables the user to select an instance of a multiple selection detail subentity.

– **Parameters**

- * **entData** - SubentityOT, data of the detail subentity to select.

• *undoEntityEdition*

```
public void undoEntityEdition( )
```

• *updateEntityData*

```
public abstract void updateEntityData( )
```

– **Usage**

- * Updates the selected instance (registry) data with the content panel's data typed by the user.

• *viewEntity*

```
public abstract OperationResult viewEntity( )
```

– **Usage**

- * View entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), in read-only mode.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

7.1.8 CLASS TestClient

DECLARATION

```
public class TestClient
extends java.lang.Object
```

CONSTRUCTORS

- *TestClient*
public **TestClient**()

METHODS

- *main*
public static void **main**(java.lang.String [] args)
– **Parameters**
* args -
- *startWarehouse*
public void **startWarehouse**()

7.1.9 CLASS WarehouseClient

DECLARATION

```
public class WarehouseClient
extends com.imagos.axilice.client.gui.common.CommonClient
```

FIELDS

- public static final String AXILICE_LOG_EDITWAREHOUSE_GLADEPATH
– Message for logging the entity glade configuration file path.

CONSTRUCTORS

- *WarehouseClient*
public **WarehouseClient**()

METHODS

- *build_extension_fields_view*
public void **build_extension_fields_view**()
- *buildSelectionSets*
public void **buildSelectionSets**()
- *clearEntityData*
public void **clearEntityData**()

- *fillEntityData*
public void fillEntityData()
- *getEntityData*
public BaseVO getEntityData()
- *getEntityFields*
public ArrayList getEntityFields()
- *getRevision*
public static String getRevision()
 - Usage
 - * Returns tag Id assigned to CVS source file.
- *getWidgets*
public void getWidgets()
- *init_specific_components*
public void init_specific_components()
- *setEditionMode*
public void setEditionMode(boolean enabled)
- *setWidgetsValues*
public void setWidgetsValues(com.imagos.axilice.core.common.BaseVO entity)
- *updateEntityData*
public void updateEntityData()
- *validateEntity*
public OperationResult validateEntity()

METHODS INHERITED FROM CLASS com.imagos.axilice.client.gui.common.CommonClient

(in 17.1.2, page 402)

- *addAxiliceEventListener*
public void addAxiliceEventListener(com.imagos.axilice.client.gui.AxiliceEventListener l)
- *addEntity*
public ArrayList addEntity()
- *addExistingSubentity*
public void addExistingSubentity(com.imagos.axilice.core.installer.SubentityOT ent)
 - Usage
 - * Shows a dialog for selection of a new subentity
 - Parameters
 - * ent - SubentityOT. Encapsulate data type of subentity
- *addSubentity*
public void addSubentity(com.imagos.axilice.core.common.BaseVO subentity, com.imagos.axilice.core.common.MultiSelectionSubentityOT subE)
 - Usage

- * Adds a existent entity to a multiselection list
- **Parameters**
 - * **subentity** - Subentity to be added
 - * **subE** - Object that encapsulates information about the subentity

- *addTrazabilityTree*
public void **addTrazabilityTree**(TabFolder tabFolder)
- *build_extension_fields_view*
public abstract void **build_extension_fields_view**()
- **Usage**
 - * Build the view of the extension fields.

It must implement only those clients who require it

- *buildEntity*
public AxiliceEntityView **buildEntity**()
- *buildEntityDetail*
public AxiliceEntityView **buildEntityDetail**(
com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData)
- *buildEntitySelectionView*
public AxiliceEntitySelectionView **buildEntitySelectionView**()
- **Usage**
 - * Build view of selection entities
- **Returns** - AxilicePartnerSelectionView. SelectionView created

- *buildSelectionSets*
public abstract void **buildSelectionSets**()
- **Usage**
 - * Build 2 sets of SubentityOT: lSelectionSubentities and lMultiSelectionSubentities

These sets are used to connect the widgets with the handlers for the selection buttons and the multiseleccion panels

It must implement only those clients who require it

- *buildSubentityMultiSelection*
public void **buildSubentityMultiSelection**()
- *clearEntityData*
public void **clearEntityData**()
- *clearParentField*
public OperationResult **clearParentField**(java.util.ArrayList selRegistries,
java.lang.String parentField)
- **Usage**
 - * Clears the value of the field indicated
- **Parameters**
 - * **selRegistries** - ArrayList. Index list of records to process
 - * **parentField** - Name of the field to clear
- **Returns** - OperationResult. Result of operation

- *delEntity*
public OperationResult **delEntity**()
- *delEntity*
public OperationResult **delEntity**(java.util.ArrayList selRegistries)

- *delEntity*
 public OperationResult **delEntity**(com.imagos.axilice.core.common.BaseVO **b**)

 - **Usage**
 * Delete a entity from database
 - **Parameters**
 * **b** - BaseVO. Entity to delete
 - **Returns** - OperationResult. Result of the operation

- *delSubentities*
 public OperationResult **delSubentities**(java.util.ArrayList **selRegistries**)

 - **Usage**
 * Delete a set of subentities from record list of controller, not from database

 Set the parent field of subentity to null
 - **Parameters**
 * **selRegistries** - List of index to delete
 - **Returns** - OperationResult. Result of the operation

- *editEntity*
 public OperationResult **editEntity**()

- *fillEntityData*
 public void **fillEntityData**()

- *fillEntityExtensionTable*
 public void **fillEntityExtensionTable**(java.lang.Long **id**)

 - **Usage**
 * Retrieve extension field values from the server for a given entity and displays them in the view
 - **Parameters**
 * **id** - Long. Key of the entity whose values are to be displayed

- *fillEntityTable*
 public void **fillEntityTable**()

 - **Usage**
 * Load records from database and fill the entity view table and controller record list

- *fillEntityTable*
 public void **fillEntityTable**(java.util.ArrayList **lR**)

 - **Usage**
 * Fill the entity view table with the records given
 - **Parameters**
 * **lR** - ArrayList. Records to be displayed

- *getClassEntity*
 public Class **getClassEntity**()

 - **Returns** - Class of the entity for which the controller is built

- *getDialogName*
 public String **getDialogName**()

- *getEditGladePath*
 public String **getEditGladePath**()

- *getEditionGui*
public Gui **getEditionGui**()
 - **Usage**
 - * Returns the edition Gui instance.
 - **Returns** - Gui, entity edition Gui.

- *getEditWindow*
public SwtTopLevelWindow **getEditWindow**()
- *getEntityData*
public abstract BaseVO **getEntityData**()
- *getEntityFields*
public ArrayList **getEntityFields**()
- *getLogger*
public Logger **getLogger**()
- *getParentEntityLauncher*
public EntityLauncher **getParentEntityLauncher**()
 - **Returns** - EntityLauncher of parent entity if is a embedded entity or null otherwise

- *getRegistries*
public Collection **getRegistries**()
 - **Returns** - lRegistries

- *getRegistries*
public List **getRegistries**(java.util.ArrayList lIds)
 - **Usage**
 - * Get a list of entities stored by the controller
 - **Parameters**
 - * lIds - Index list
 - **Returns** - List of entities

- *getView*
public AxiliceEntityView **getView**()
 - **Returns** - AxiliceEntityView. Main view or Detail view if is a embedded entity.

- *getWidgets*
public abstract void **getWidgets**()
 - **Usage**
 - * Get the widgets from the glade and stored in local variables.
 - Must be overwritten

- *init_specific_components*
public abstract void **init_specific_components**()
 - **Usage**
 - * Initializes the specific components of the client

- *printEntity*
public OperationResult **printEntity**()
- *printEntity*
public OperationResult **printEntity**(java.lang.String reportPath, java.util.HashMap parameters)

- *queryEntity*
public ArrayList **queryEntity**()
- *queryEntity*
public ArrayList **queryEntity**(java.lang.String fieldId, java.lang.String value)
- *queryEntityWithFormat*
public ArrayList **queryEntityWithFormat**(java.lang.String fieldId, java.lang.String value)
- *queryFreeEntities*
public ArrayList **queryFreeEntities**(com.imagos.axilice.core.common.MultiSelectionSubentityOT mSubE)
 - **Usage**
 - * Search entities not associated with a parent entity (null parent field)
 - **Parameters**
 - * mSubE - SubentityOT for the parent entity that must be null
 - **Returns** - ArrayList. List of entities found
- *querySubentities*
public ArrayList **querySubentities**(com.imagos.axilice.core.installer.SubentityOT entData)
- *refreshEntity*
public OperationResult **refreshEntity**()
- *refreshEntity*
public BaseVO **refreshEntity**(com.imagos.axilice.core.common.BaseVO entity)
- *refreshParentSelectedEntity*
public OperationResult **refreshParentSelectedEntity**()
 - **Usage**
 - * Reload from database the value of the parent entity of the selected entity
 - **Returns** - OperationResult. The result of the operation.
- *removeRegistries*
public OperationResult **removeRegistries**(java.util.ArrayList selRegistries, java.lang.String parentField)
 - **Usage**
 - * Deletes a list of the records stored by the controller. Also clean your parent fields and updates the database
 - **Parameters**
 - * selRegistries - Index list to remove
 - * parentField - Field name that links to the parent entity
 - **Returns** - OperationResult. Result of the operation
- *saveEntity*
public OperationResult **saveEntity**()
- *saveEntity*
public OperationResult **saveEntity**(java.util.ArrayList lIds)
- *saveEntity*
public OperationResult **saveEntity**(com.imagos.axilice.core.common.BaseVO b)
 - **Usage**
 - * Save or update a entity to database
 - **Parameters**
 - * b - BaseVO. Entity to save or update
 - **Returns** - OperationResult. Result of the operation

- *selectEntity*
public void selectEntity(java.lang.Long idEntity)
- *selectSubentity*
public void selectSubentity(java.lang.String parentFieldId, com.imagos.axilice.core.common.BaseVO subentityOT)
- *selectTreeItem*
public void selectTreeItem(java.lang.Long id)
 - Usage
 - * Select a item in the entity tree. No me gusta esto aqui
 - Parameters
 - * id - Long. Index of item to select

- *setDialogName*
public void setDialogName(java.lang.String dialogName)
- *setEditGladePath*
public void setEditGladePath(java.lang.String editGladePath)
- *setEditionMode*
public abstract void setEditionMode(boolean enabled)
- *setRegistries*
public void setRegistries(java.util.Collection collection)
 - Usage
 - * Setter for lRegistries

- *setWidgetsValues*
public abstract void setWidgetsValues(com.imagos.axilice.core.common.BaseVO entity)
 - Usage
 - * Set the values of the widgets on the edit dialog.
 - Must be overwritten.
 - Parameters
 - * entity - BaseVO. Entity that contains the values to be displayed

- *showErrorMessages*
public void showErrorMessages(com.imagos.axilice.core.util.OperationResult result)

- *updateEntityData*
public void updateEntityData()
- *updateRegistry*
public void updateRegistry(com.imagos.axilice.core.common.BaseVO ent)
 - Usage
 - * Add or update an entity (to the controller records)
 - Parameters
 - * ent - Entity to add or update

- *updateViews*
public void updateViews(boolean updateFromParent)
 - Usage
 - * Force the update of the views associated
 - Parameters
 - * updateFromParent - Indicates whether the view must be updated by reading the values from the parent entity. takes effect only when the entity is detail (is embedded in another)

-
- *validateEntity*
 public abstract `OperationResult` **validateEntity**()
 – **Usage**
 * Validate the values of the widgets in the edit dialog.
 Must be overwritten.
 – **Returns** - `OperationResult`. Result of the validation.
-
- *viewEntity*
 public `OperationResult` **viewEntity**()

METHODS INHERITED FROM CLASS `com.imagos.axilice.core.installer.EntityLauncher`

(in 24.2.2, page 532)

- *addEntity*
 public abstract `ArrayList` **addEntity**()
 – **Usage**
 * Add entity button clicked handler. Show the entity content screen ready to start inserting data for a new instance of the entity (new registry).
 – **Returns** - `ArrayList<ResultMessage>`, list of errors if any has taken place, null in any other case.
-
- *buildEditEntityGUI*
 public void **buildEditEntityGUI**()
 – **Usage**
 * Builds the edit entity dialog window from the corresponding glade interface file. Stores the entity dialog instance for posterior accesses.
-
- *buildEntity*
 public `AxiliceEntityView` **buildEntity**()
 – **Usage**
 * Builds the entity GUI and show it on screen to the user.
-
- *buildEntity*
 public `AxiliceDetailEntityView` **buildEntity**(`SwtWidget` parent,
`com.imagos.axilice.core.common.MultiSelectionSubentityOT` subentityData,
`java.lang.String` rootName)
 – **Usage**
 * Builds the entity GUI and show it on screen to the user. The entity GUI will be embedded into the parent composite widget.
 – **Parameters**
 * **parent** - Composite, parent holder for embedding the entity view.
 * **subentityData** - SubentityOT, data of the subentity to build if the entity launcher will be used to handle a multi-selection detail entity view.
 * **rootName** - String, name of the root widget that will be embedded into the master entity edition panel.
 – **Returns** - `AxiliceDetailEntityView`, the entity detail view built.
-
- *cancelEntityEdition*
 public void **cancelEntityEdition**()
-
- *clearEntityData*
 public abstract void **clearEntityData**()

- **Usage**
 - * Clears the entity content panel by resetting the content widgets.

- *delEntity*

```
public abstract OperationResult delEntity( )
```

 - **Usage**
 - * Delete entity button clicked handler. Deletes the selected instance of the entity (selected registry) from the database.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *delEntity*

```
public abstract OperationResult delEntity( java.util.ArrayList lSelRegistries )
```

 - **Usage**
 - * Delete entity button clicked handler. Deletes the selected instances of the entity (selected registries) from the database. Its a multiple delete method.
 - **Parameters**
 - * lSelRegistries - ArrayList<Long>, list of selected instances of the entity (selected registries). The list will contain only the internal id of the selected registries.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *deselectSubentity*

```
public void deselectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

 - **Usage**
 - * Updates an entity instance by removing the selected value for a subentity field.
 - **Parameters**
 - * parentFieldId - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * subentityOT - BaseVO, subentity's deselected instance (registry) value.

- *editEntity*

```
public abstract OperationResult editEntity( )
```

 - **Usage**
 - * Edit entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), and allows the user to modify any content of the entity registry.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *fillEntityData*

```
public abstract void fillEntityData( )
```

 - **Usage**
 - * Fills the entity content panel with the data of the selected instance (registry) of the entity.

- *fireExceptionAndExit*

```
public void fireExceptionAndExit( java.lang.String msg )
```

 - **Usage**
 - * Fire a exception and stop the program
 - **Parameters**
 - * msg -

- *formatRegistries*
public ArrayList formatRegistries()
- *formatRegistries*
public ArrayList formatRegistries(java.util.ArrayList lRegistries)
- *getDialog*
public SwtDialog getDialog(java.lang.String dialogName)
- *getEntity*
public EntityOT getEntity()
 - **Usage**
 - * Gets the entity data.
 - **Returns** - EntityOT, entity data.

- *getEntityData*
public abstract BaseVO getEntityData()
 - **Usage**
 - * Gets the content panel's data typed by the user formatted as an entity VO object.

- *getEntityEditViewName*
public String getEntityEditViewName(java.lang.String parentGuiName)
- *getEntityFields*
public abstract ArrayList getEntityFields()
 - **Usage**
 - * Gets the list of defined fields for an entity
 - **Returns** - ArrayList, list of entity fields.

- *getEntityViewName*
public String getEntityViewName(java.lang.String parentGuiName)
- *getLRegistries*
public ArrayList getLRegistries()
 - **Usage**
 - * Getter for lRegistries

- *getRegistry*
public BaseVO getRegistry(long id)
- *getRevision*
public static String getRevision()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.

- *getSelectedEntity*
public BaseVO getSelectedEntity()
 - **Usage**
 - * Gets the selected instance (registry) of the entity.
 - **Returns** - BaseVO, selected instance (registry) of the entity.

- *getSelectedId*
public Long getSelectedId()
 - **Usage**
 - * Gets the internal id of the selected entity registry.
 - **Returns** - Long, internal id of the selected entity registry, null in case no entity registry has been selected yet.

- *getSwtWidget*
public SwtWidget getSwtWidget(java.lang.String widgetName)
- *getWidget*
public Widget getWidget(java.lang.String widgetName)
- *initEntityLauncher*
public void initEntityLauncher(Logger logger, Gui parentGui, SwtWidget parentPanel)
 - **Usage**
 - * Initiates the entity launcher context and prepares the launcher to be started.
 - **Parameters**
 - * **logger** - Logger, logger instance for printing logging messages.
 - * **parentGuiName** - String, name of the JavaGlade instance that built the parent where the entity will be embedded.
 - * **parentPanel** - Composite, parent panel where the entity contents will be embedded.
- *isDetail*
public boolean isDetail()
 - **Usage**
 - * Indicates whether is a main entity or is embedded in another
- *on_entitySelectionButton_clicked*
public void on_entitySelectionButton_clicked(Event e)
 - **Usage**
 - * Handler for the subentity selection buttons (magnifying glass buttons) clicked. Shows the selection dialog for selecting a valid instance of a subentity.
 - **Parameters**
 - * **e** - Event, magnifying glass button clicked event data.
- *printEntity*
public abstract OperationResult printEntity()
 - **Usage**
 - * Print entity button clicked handler. Builds the report with the data of the selected instance of the entity (selected registry), and allows the user to print the elaborated report.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.
- *printWidgets*
public void printWidgets()
- *queryEntity*
public abstract ArrayList queryEntity()
 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) available.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.
- *queryEntity*
public abstract ArrayList queryEntity(java.lang.String fieldId, java.lang.String value)
 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria.
 - **Parameters**

- * **fieldId** - String, name of the entity field that will be queried.
 - * **value** - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.

- *queryEntityWithFormat*

```
public abstract ArrayList queryEntityWithFormat( java.lang.String fieldId,
java.lang.String value )
```

 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria and show them (formatted as a list) to the user in the entity's screen.
 - **Parameters**
 - * **fieldId** - String, name of the entity field that will be queried.
 - * **value** - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList<RegistryOT>, list of entity instances (registries) taken as the result of the query operation.

- *querySubentities*

```
public abstract ArrayList querySubentities(
com.imagos.axilice.core.installer.SubentityOT entData )
```

 - **Usage**
 - * Queries the selected entity for all the detail subentities associated with it.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to query for.
 - **Returns** - ArrayList, list of detail subentity registries assigned to the master detail registry.

- *saveEntity*

```
public abstract OperationResult saveEntity( )
```

 - **Usage**
 - * Save entity button clicked handler. Update the system with the data of the edited entity registry. It's called for saving new entity registries (add a new registry) or for updating the contents of existing entity registries.
 - **Returns** - OperationResult, Result of the operation and list of messages generated through the operation.

- *saveEntityRegistry*

```
public void saveEntityRegistry( )
```
- *selectEntity*

```
public abstract void selectEntity( java.lang.Long idEntity )
```

 - **Usage**
 - * Marks an instance (registry) of the entity as the selected.
 - **Parameters**
 - * **idEntity** - Long, internal id of the selected instance (registry).

- *selectSubentity*

```
public void selectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

 - **Usage**
 - * Updates an entity instance by setting up the selected value for a subentity field.
 - **Parameters**
 - * **parentFieldId** - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.

-
- * subentityOT - BaseVO, subentity's selected instance (registry) value.
-
- *setEntity*
 public void **setEntity**(com.imagos.axilice.core.installer.EntityOT entity)
 – **Usage**
 * Sets up the entity name.
 – **Parameters**
 * name - String, entity name.
-
- *setRegistries*
 public void **setRegistries**(java.util.ArrayList lRegistries)
 – **Usage**
 * Sets the list of registries that are showed in the entity view controlled by the Entity Launcher instance. It's used for detail entities where queries are made in the master entity launcher.
 – **Parameters**
 * lRegistries - ArrayList<BaseVO>, list of shown registries of the entity.
-
- *showSelectionDialog*
 public void **showSelectionDialog**(com.imagos.axilice.core.installer.SubentityOT entData)
 – **Usage**
 * Opens the selection dialog that enables the user to select an instance of a subentity.
 – **Parameters**
 * entData - SubentityOT, data of the detail subentity to select.
-
- *showSubentityDialog*
 public void **showSubentityDialog**(com.imagos.axilice.core.installer.SubentityOT entData, long selEntity, boolean readonly)
 – **Usage**
 * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.
 – **Parameters**
 * entData - SubentityOT, data of the detail subentity to select.
 * selEntity - long, id of the selected instance of the subentity.
 * readonly - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.
-
- *showSubentityDialog*
 public void **showSubentityDialog**(com.imagos.axilice.core.installer.SubentityOT entData, java.lang.String parentGuiName, long selEntity, boolean readonly)
 – **Usage**
 * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.
 – **Parameters**
 * entData - SubentityOT, data of the detail subentity to select.
 * parentGuiName - String,
 * selEntity - long, id of the selected instance of the subentity.
 * readonly - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.
-
- *showSubentitySelectionDialog*
 public void **showSubentitySelectionDialog**(
 com.imagos.axilice.core.installer.SubentityOT entData)

- **Usage**
 - * Opens the selection dialog that enables the user to select an instance of a multiple selection detail subentity.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to select.
-
- *undoEntityEdition*
public void undoEntityEdition()
 - *updateEntityData*
public abstract void updateEntityData()
- **Usage**
 - * Updates the selected instance (registry) data with the content panel's data typed by the user.
-
- *viewEntity*
public abstract OperationResult viewEntity()
- **Usage**
 - * View entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), in read-only mode.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

Chapter 8

Package

com.imagos.axilice.core.traceability.tree

Package Contents

Page

Classes

AppccInfo	217
<i>ValueObject with the data of an Appcc item.</i>	
CommentInfo	217
<i>...no description...</i>	
ProcessInfo	217
<i>ValueObject with the data of a Process item.</i>	
ProductInfo	222
<i>ValueObject with the data of a Product item.</i>	
PropertyInfo	229
<i>ValueObject with the data of a Property item.</i>	
PurchaseInfo	234
<i>ValueObject with the data of a Purchase item.</i>	
TraceabilityTree	235
<i>ValueObject with the data of a Traceability Tree.</i>	
TraceabilityTreeBuilder	237
<i>Factory class that gives access to the Traceability Tree Construction mechanism.</i>	
WarehouseInfo	238
<i>ValueObject with the data of a Warehouse item.</i>	
WorkcenterInfo	240
<i>ValueObject with the data of a Workcenter item.</i>	
WorkerInfo	242
<i>ValueObject with the data of a Worker item.</i>	

8.1 Classes

8.1.1 CLASS AppccInfo

ValueObject with the data of an Appcc item.
Responsibilities:

- Store the appcc data.

DECLARATION

```
public class AppccInfo
extends java.lang.Object
```

CONSTRUCTORS

- *AppccInfo*
public **AppccInfo**()

METHODS

- *getRevision*
public static String **getRevision**()
– **Usage**
* Returns tag Id assigned to CVS source file.

8.1.2 CLASS CommentInfo

DECLARATION

```
public class CommentInfo
extends java.lang.Object
```

CONSTRUCTORS

- *CommentInfo*
public **CommentInfo**()

8.1.3 CLASS ProcessInfo

ValueObject with the data of a Process item.
Responsibilities:

- Store the process data.

DECLARATION

```
public class ProcessInfo
extends java.lang.Object
```

CONSTRUCTORS

- *ProcessInfo*
 public **ProcessInfo**()
 – **Usage**
 * Default ProcessInfo constructor.

- *ProcessInfo*
 public **ProcessInfo**(java.lang.String **dateProcess**, java.lang.String **codOperation**)
 – **Usage**
 * ProcessInfo constructor that initializes the process values.
 – **Parameters**
 * **dateProcess** - String, process date.
 * **codOperation** - String, process code.

METHODS

- *addComment*
 public void **addComment**(java.lang.String **comment**)
 – **Usage**
 * Adds a new comment to the process comments' list.
 – **Parameters**
 * **comment** - String, comment's data to insert.

- *addComments*
 public void **addComments**(java.util.ArrayList **comments**)
 – **Usage**
 * Adds a list of comments to the process comments' list.
 – **Parameters**
 * **comments** - ArrayList<String>, comments' list to insert.

- *addIn*
 public void **addIn**(com.imagos.axilice.core.traceability.tree.ProductInfo **p**)
 – **Usage**
 * Adds a new product to the input process list.
 – **Parameters**

* **p** - ProductInfo, input product's data to insert.

- *addOperation*

public void addOperation(java.lang.String ope)

- **Usage**

* Adds a new operation to the process operations' list.

- **Parameters**

* **ope** - String, operation's data to insert.

- *addWorkcenter*

public void addWorkcenter(com.imagos.axilice.core.traceability.tree.WorkcenterInfo workcenter)

- **Usage**

* Adds a new workcenter to the process workcenters' list.

- **Parameters**

* **workcenter** - WorkcenterInfo, workcenter's data to insert.

- *addWorker*

public void addWorker(com.imagos.axilice.core.traceability.tree.WorkerInfo worker)

- **Usage**

* Adds a new worker to the process workers' list.

- **Parameters**

* **worker** - WorkerInfo, worker's data to insert.

- *delComment*

public void delComment(java.lang.String c)

- **Usage**

* Removes a comment from the process comments' list.

- **Parameters**

* **c** - String, comment's data to delete.

- *delIn*

public void delIn(com.imagos.axilice.core.traceability.tree.ProductInfo p)

- **Usage**

* Removes a product from the input process list.

- **Parameters**

* **p** - ProductInfo, input product's data to delete.

- *delWorkcenter*

public void delWorkcenter(com.imagos.axilice.core.traceability.tree.WorkcenterInfo w)

- **Usage**

* Removes a workcenter from the process workcenters' list.

– **Parameters**

* **w** - WorkcenterInfo, workcenter's data to delete.

• *delWorker*

```
public void delWorker( com.imagos.axilice.core.traceability.tree.WorkerInfo
w )
```

– **Usage**

* Removes a worker from the process workers' list.

– **Parameters**

* **w** - WorkerInfo, worker's data to delete.

• *getCodOperation*

```
public String getCodOperation( )
```

– **Usage**

* Gets the process code.

– **Returns** - String, process code.

• *getDateProcess*

```
public String getDateProcess( )
```

– **Usage**

* Gets the process date.

– **Returns** - String, process date.

• *getLComments*

```
public ArrayList getLComments( )
```

– **Usage**

* Gets the list of comments linked to the process.

– **Returns** - ArrayList, list of comments.

• *getLIn*

```
public ArrayList getLIn( )
```

– **Usage**

* Gets the list of input products for the process.

– **Returns** - ArrayList, list of input products.

• *getLOperations*

```
public ArrayList getLOperations( )
```

– **Usage**

* Gets the list of operations of the process.

– **Returns** - ArrayList, list of operations.

• *getLWorkcenters*

```
public ArrayList getLWorkcenters( )
```

– **Usage**

* Gets the list of workcenters of the process.

- **Returns** - ArrayList, list of workcenters.

- *getLWorkers*

```
public ArrayList getLWorkers( )
```

- **Usage**
 - * Gets the list of workers involved in the process.
- **Returns** - ArrayList, list of workers.

- *getRevision*

```
public static String getRevision( )
```

- **Usage**
 - * Returns tag Id assigned to CVS source file.

- *setCodOperation*

```
public void setCodOperation( java.lang.String codOperation )
```

- **Usage**
 - * Sets up the process code.
- **Parameters**
 - * *codOperation* - String, an operation code.

- *setDateProcess*

```
public void setDateProcess( java.lang.String dateProcess )
```

- **Usage**
 - * Sets up the process date.
- **Parameters**
 - * *dateProcess* - String, a date.

- *setLComments*

```
public void setLComments( java.util.ArrayList com )
```

- **Usage**
 - * Sets up the input list of comments linked to the process.
- **Parameters**
 - * *com* - ArrayList, list of comments.

- *setLIn*

```
public void setLIn( java.util.ArrayList in )
```

- **Usage**
 - * Sets up the input list of products for the process.
- **Parameters**
 - * *in* - ArrayList, list of input products.

- *setLOperations*

```
public void setLOperations( java.util.ArrayList ope )
```

- **Usage**

- * Sets up the list of operations of the process.
 - **Parameters**
 - * ope - ArrayList, list of operations.

- *setLWorkcenters*

```
public void setLWorkcenters( java.util.ArrayList lWorkcenters )
```

 - **Usage**
 - * Sets up the list of workcenters of the process.
 - **Parameters**
 - * lWorkcenters - ArrayList, list of workcenters.

- *setLWorkers*

```
public void setLWorkers( java.util.ArrayList per )
```

 - **Usage**
 - * Sets up the list of workers involved in the process.
 - **Parameters**
 - * per - ArrayList, list of workers.

- *toString*

```
public String toString( )
```

 - **Usage**
 - * Converts the process info to a formatted String.
 - **Returns** - String, formatted String with the process's data.

- *update*

```
public void update( com.imagos.axilice.core.traceability.tree.ProcessInfo info )
```

 - **Usage**
 - * Updates the process data.
 - **Parameters**
 - * info - ProcessInfo, new process data.

8.1.4 CLASS ProductInfo

ValueObject with the data of a Product item.
Responsibilities:

- Store the product data.

DECLARATION

```
public class ProductInfo
extends java.lang.Object
```

FIELDS

- public static final int SOURCE_NONE
 - Constant definition with the product source type NONE. It's the default value to point out that no source type has been assigned to the product yet.
- public static final int SOURCE_PURCHASE
 - Constant definition with the product source type PURCHASE. It points out that the product has been acquired from an external supplier.
- public static final int SOURCE_PROCESS
 - Constant definition with the product source type PROCESS. It points out that the product has been produced from other products as the result of an internal process.

CONSTRUCTORS

- *ProductInfo*
 public **ProductInfo**()
 - **Usage**
 - * Default ProductInfo constructor.
- *ProductInfo*
 public **ProductInfo**(java.lang.String reference, java.lang.String setNumber, java.lang.String barcode, java.lang.String description, java.lang.String dateIn, java.lang.String dateExpiry, java.lang.String traceabilityFile)
 - **Usage**
 - * ProductInfo constructor that initializes the product values.
 - **Parameters**
 - * **reference** - String, product reference.
 - * **setNumber** - String, product set number.
 - * **barcode** - String, product barcode.
 - * **description** - String, product description.
 - * **dateIn** - String, product in date.
 - * **dateExpiry** - String, product expiry date.
 - * **traceabilityFile** - String, product traceability file path.

METHODS

- *addComment*
 public void **addComment**(java.lang.String c)
 - **Usage**
 - * Adds a new comment to the product comments' list.
 - **Parameters**
 - * **c** - String, comment's data to insert.

- *addComments*

```
public void addComments( java.util.ArrayList comments )
```

- **Usage**

- * Adds a list of comments to the product comments' list.

- **Parameters**

- * **comments** - ArrayList<String>, comments' list to insert.

- *addProperty*

```
public void addProperty(
com.imagos.axilice.core.traceability.tree.PropertyInfo p )
```

- **Usage**

- * Adds a new property to the product properties' list.

- **Parameters**

- * **p** - PropertyInfo, property's data to insert.

- *addWarehouse*

```
public void addWarehouse(
com.imagos.axilice.core.traceability.tree.WarehouseInfo w )
```

- **Usage**

- * Adds a new warehouse to the product warehouses' list.

- **Parameters**

- * **w** - WarehouseInfo, warehouse's data to insert.

- *delComment*

```
public void delComment( java.lang.String c )
```

- **Usage**

- * Removes a comment from the product comments' list.

- **Parameters**

- * **c** - String, comment's data to delete.

- *delProperty*

```
public void delProperty(
com.imagos.axilice.core.traceability.tree.PropertyInfo p )
```

- **Usage**

- * Removes a property from the product properties' list.

- **Parameters**

- * **p** - PropertyInfo, property's data to delete.

- *delWarehouse*

```
public void delWarehouse(
com.imagos.axilice.core.traceability.tree.WarehouseInfo w )
```

- **Usage**

- * Removes a warehouse from the product warehouses' list.

- **Parameters**

* w - WarehouseInfo, warehouse's data to delete.

- *getBarcode*

public String **getBarcode**()

– **Usage**

* Gets the product barcode number.

– **Returns** - String, product barcode number.

- *getDateExpiry*

public String **getDateExpiry**()

– **Usage**

* Gets the product expiry date.

– **Returns** - String, product expiry date.

- *getDateIn*

public String **getDateIn**()

– **Usage**

* Gets the product in date.

– **Returns** - String, product in date.

- *getDescription*

public String **getDescription**()

– **Usage**

* Gets the product description.

– **Returns** - String, product description.

- *getLComments*

public ArrayList **getLComments**()

– **Usage**

* Gets the product comments' list.

– **Returns** - ArrayList, list of product comments.

- *getLProperties*

public ArrayList **getLProperties**()

– **Usage**

* Gets the product properties' list.

– **Returns** - ArrayList, list of product properties.

- *getLWarehouses*

public ArrayList **getLWarehouses**()

– **Usage**

* Gets the product warehouses' list.

– **Returns** - ArrayList, list of product warehouses.

- *getReference*

public String **getReference**()

- **Usage**

- * Gets the product reference.

- **Returns** - String, product reference.

- *getRevision*

public static String **getRevision**()

- **Usage**

- * Returns tag Id assigned to CVS source file.

- *getSetNumber*

public String **getSetNumber**()

- **Usage**

- * Gets the product set number.

- **Returns** - String, product set number.

- *getSourceProcess*

public ProcessInfo **getSourceProcess**()

- **Usage**

- * Gets the product source process.

- **Returns** - ProcessInfo, product source process.

- *getSourcePurchase*

public PurchaseInfo **getSourcePurchase**()

- **Usage**

- * Gets the product source purchase.

- **Returns** - PurchaseInfo, product source purchase.

- *getSourceType*

public int **getSourceType**()

- **Usage**

- * Gets the product source type.

- **Returns** - int, product source type.

- **See Also**

- * com.imagos.axilice.core.traceability.tree.ProductInfo.SOURCE_NONE

- * com.imagos.axilice.core.traceability.tree.ProductInfo.SOURCE_PROCESS

- * com.imagos.axilice.core.traceability.tree.ProductInfo.SOURCE_PURCHASE

- *getTraceabilityFile*

public String **getTraceabilityFile**()

- **Usage**

- * Gets the product traceability file path.

- **Returns** - String, product traceability file path.

-
- *setBarcode*
public void **setBarcode**(java.lang.String barcode)
 - **Usage**
 - * Sets up the product barcode number.
 - **Parameters**
 - * barcode - String, a barcode number.
-
- *setDateExpiry*
public void **setDateExpiry**(java.lang.String dateExpiry)
 - **Usage**
 - * Sets up the product expiry date.
 - **Parameters**
 - * dateExpiry - String, a date.
-
- *setDateIn*
public void **setDateIn**(java.lang.String dateIn)
 - **Usage**
 - * Sets up the product in date.
 - **Parameters**
 - * dateIn - String, a date.
-
- *setDescription*
public void **setDescription**(java.lang.String description)
 - **Usage**
 - * Sets up the product description.
 - **Parameters**
 - * description - String, a description.
-
- *setLComments*
public void **setLComments**(java.util.ArrayList comments)
 - **Usage**
 - * Sets up the list of comments for the product.
 - **Parameters**
 - * comments - ArrayList, list of comments.
-
- *setLProperties*
public void **setLProperties**(java.util.ArrayList properties)
 - **Usage**
 - * Sets up the list of properties for the product.
 - **Parameters**
 - * properties - ArrayList, list of properties.
-
- *setLWarehouses*
public void **setLWarehouses**(java.util.ArrayList warehouses)

- **Usage**
 - * Sets up the list of warehouses for the product.
- **Parameters**
 - * **warehouses** - ArrayList, list of warehouses.

- *setReference*

```
public void setReference( java.lang.String reference )
```

 - **Usage**
 - * Sets up the product reference.
 - **Parameters**
 - * **reference** - String, a product reference.

- *setSetNumber*

```
public void setSetNumber( java.lang.String setNumber )
```

 - **Usage**
 - * Sets up the product set number.
 - **Parameters**
 - * **setNumber** - String, a product set number.

- *setSource*

```
public void setSource( com.imagos.axilice.core.traceability.tree.ProcessInfo source )
```

 - **Usage**
 - * Sets up the product source of type Process.
 - **Parameters**
 - * **source** - ProcessInfo, a source process.

- *setSource*

```
public void setSource(
com.imagos.axilice.core.traceability.tree.PurchaseInfo source )
```

 - **Usage**
 - * Sets up the product source of type Purchase.
 - **Parameters**
 - * **source** - PurchaseInfo, a source purchase.

- *setSourceProcess*

```
public void setSourceProcess(
com.imagos.axilice.core.traceability.tree.ProcessInfo sourceProcess )
```

 - **Usage**
 - * Sets up the product source of type Process.
 - **Parameters**
 - * **sourceProcess** - ProcessInfo, a source process.

- *setSourcePurchase*

```
public void setSourcePurchase(
com.imagos.axilice.core.traceability.tree.PurchaseInfo sourcePurchase )
```

- **Usage**
 - * Sets up the product source of type Purchase.
 - **Parameters**
 - * `sourcePurchase` - PurchaseInfo, a source purchase.
-
- *setSourceType*
 public void **setSourceType**(int **sourceType**)
 - **Usage**
 - * Sets up the product source type.
 - **Parameters**
 - * `sourceType` - int, a product source type.
 - **See Also**
 - * com.imagos.axilice.core.traceability.tree.ProductInfo.SOURCE_NONE
 - * com.imagos.axilice.core.traceability.tree.ProductInfo.SOURCE_PROCESS
 - * com.imagos.axilice.core.traceability.tree.ProductInfo.SOURCE_PURCHASE
-
- *setTraceabilityFile*
 public void **setTraceabilityFile**(java.lang.String **traceabilityFile**)
 - **Usage**
 - * Sets up the product traceability file path.
 - **Parameters**
 - * `traceabilityFile` - String, a file path.
-
- *toString*
 public String **toString**()
 - **Usage**
 - * Converts the product info to a formatted String.
 - **Returns** - String, formatted String with the product's data.
-
- *update*
 public void **update**(com.imagos.axilice.core.traceability.tree.ProductInfo
info)
 - **Usage**
 - * Updates the product data.
 - **Parameters**
 - * `info` - ProductInfo, new product data.

8.1.5 CLASS PropertyInfo

ValueObject with the data of a Property item.
Responsibilities:

- Store the property data.

DECLARATION

```
public class PropertyInfo  
extends java.lang.Object
```

FIELDS

- public static final int PROPERTY_TYPE_NONE
 - Constant definition with the property type NONE. It's the default value to point out that no property type has been assigned yet.
- public static final int PROPERTY_TYPE_NUMBER
 - Constant definition with the property type NUMBER. It's the value for properties of any type of numeric value, such as Integers, Floats, Decimals, Doubles, etc.
- public static final int PROPERTY_TYPE_STRING
 - Constant definition with the property type STRING. It's the value for properties of any type of alphanumeric string value.
- public static final int PROPERTY_TYPE_DATE
 - Constant definition with the property type DATE. It's the value for properties of date/time values in any format.
- public static final int PROPERTY_TYPE_BOOLEAN
 - Constant definition with the property type BOOLEAN. It's the value for properties with a True/False, On/Off value.
- public static final int PROPERTY_TYPE_OBJECT
 - Constant definition with the property type OBJECT. It's the value for properties of any other type of value different from the ones described above, as for example, BLOBS, IMAGES, etc.
- public static final String CONST_PROPERTY_TYPE_NONE
 - Constant definition with the text showed to the user for the property type NONE.
- public static final String CONST_PROPERTY_TYPE_NUMBER
 - Constant definition with the text showed to the user for the property type NUMBER.
- public static final String CONST_PROPERTY_TYPE_STRING
 - Constant definition with the text showed to the user for the property type STRING.
- public static final String CONST_PROPERTY_TYPE_DATE
 - Constant definition with the text showed to the user for the property type DATE.
- public static final String CONST_PROPERTY_TYPE_BOOLEAN
 - Constant definition with the text showed to the user for the property type BOOLEAN.

- public static final String CONST_PROPERTY_TYPE_OBJECT
 - Constant definition with the text showed to the user for the property type OBJECT.
- public static final Hashtable propertyTypes
 - Hashtable that links any property type with its corresponding label showed to the user.

CONSTRUCTORS

- *PropertyInfo*
public **PropertyInfo**()
 - **Usage**
 - * Default PropertyInfo constructor.
- *PropertyInfo*
public **PropertyInfo**(java.lang.String cod, java.lang.String val, java.lang.String type, java.lang.String description, java.lang.String pattern)
 - **Usage**
 - * PropertyInfo constructor that initializes the property values.
 - **Parameters**
 - * cod - String, property code.
 - * val - String, property value in String format.
 - * type - String, property type.
 - * description - String, property description.
 - * pattern - String, property format pattern.

METHODS

- *getCod*
public String **getCod**()
 - **Usage**
 - * Gets the property code.
 - **Returns** - String, property code.
- *getDescription*
public String **getDescription**()
 - **Usage**
 - * Gets the property description.
 - **Returns** - String, property description.
- *getPattern*
public String **getPattern**()
 - **Usage**
 - * Gets the property format pattern.

– **Returns** - String, property format pattern.

- *getRevision*

public static String **getRevision**()

– **Usage**

* Returns tag Id assigned to CVS source file.

- *getType*

public int **getType**()

– **Usage**

* Gets the property type.

– **Returns** - int, property type.

– **See Also**

*
 com.imagos.axilice.core.traceability.tree.PropertyInfo.PROPERTY_TYPE_NONE
 *
 com.imagos.axilice.core.traceability.tree.PropertyInfo.PROPERTY_TYPE_NUMBER
 *
 com.imagos.axilice.core.traceability.tree.PropertyInfo.PROPERTY_TYPE_STRING
 *
 com.imagos.axilice.core.traceability.tree.PropertyInfo.PROPERTY_TYPE_DATE
 *
 com.imagos.axilice.core.traceability.tree.PropertyInfo.PROPERTY_TYPE_BOOLEAN
 *
 com.imagos.axilice.core.traceability.tree.PropertyInfo.PROPERTY_TYPE_OBJECT

- *getVal*

public Object **getVal**()

– **Usage**

* Gets the property value.

– **Returns** - Object, property value.

- *setCod*

public void **setCod**(java.lang.String cod)

– **Usage**

* Sets up the property code.

– **Parameters**

* cod - String, a property code.

- *setDescription*

public void **setDescription**(java.lang.String description)

– **Usage**

* Sets up the property description.

– **Parameters**

* description - String, a description.

- *setPattern*

```
public void setPattern( java.lang.String pattern )
```

- **Usage**

- * Sets up the property format pattern.

- **Parameters**

- * *pattern* - String, a format pattern.

- *setType*

```
public void setType( int type )
```

- **Usage**

- * Sets up the property type.

- **Parameters**

- * *type* - int, a property type.

- **See Also**

- *
com.imagos.axilice.core.traceability.tree.PropertyInfo.PROPERTY_TYPE_NONE
 - *
com.imagos.axilice.core.traceability.tree.PropertyInfo.PROPERTY_TYPE_NUMBER
 - *
com.imagos.axilice.core.traceability.tree.PropertyInfo.PROPERTY_TYPE_STRING
 - *
com.imagos.axilice.core.traceability.tree.PropertyInfo.PROPERTY_TYPE_DATE
 - *
com.imagos.axilice.core.traceability.tree.PropertyInfo.PROPERTY_TYPE_BOOLEAN
 - *
com.imagos.axilice.core.traceability.tree.PropertyInfo.PROPERTY_TYPE_OBJECT

- *setVal*

```
public void setVal( java.lang.Object val )
```

- **Usage**

- * Sets up the property code.

- **Parameters**

- * *val* - Object, a value.

- *toString*

```
public String toString( )
```

- **Usage**

- * Converts the property info to a formatted String.

- **Returns** - String, formatted String with the property's data.

- *update*

```
public void update( com.imagos.axilice.core.traceability.tree.PropertyInfo  
info )
```

- **Usage**

- * Updates the property data.
- **Parameters**
 - * **info** - PropertyInfo, new property data.

8.1.6 CLASS *PurchaseInfo*

ValueObject with the data of a Purchase item.
Responsibilities:

- Store the purchase data.

DECLARATION

```
public class PurchaseInfo
extends java.lang.Object
```

CONSTRUCTORS

- *PurchaseInfo*

```
public PurchaseInfo( )
```

 - **Usage**
 - * Default PurchaseInfo constructor
- *PurchaseInfo*

```
public PurchaseInfo( java.lang.String  codPurchaseConfirmation,
java.lang.String  codSupplier )
```

 - **Usage**
 - * PurchaseInfo constructor that initializes the purchase values.
 - **Parameters**
 - * **codPurchaseConfirmation** - String, purchase confirmation code.
 - * **codSupplier** - String, purchase supplier code.

METHODS

- *getCodPurchaseConfirmation*

```
public String getCodPurchaseConfirmation( )
```

 - **Usage**
 - * Gets the purchase confirmation code.
 - **Returns** - String, purchase confirmation code.
- *getCodSupplier*

```
public String getCodSupplier( )
```

 - **Usage**

- * Gets the purchase supplier code.
 - **Returns** - String, purchase supplier code.

- *getRevision*
 - public static String **getRevision**()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.

- *setCodPurchaseConfirmation*
 - public void **setCodPurchaseConfirmation**(java.lang.String **codPurchaseConfirmation**)
 - **Usage**
 - * Sets up the purchase confirmation code.
 - **Parameters**
 - * **codPurchaseConfirmation** - String, a purchase confirmation code.

- *setCodSupplier*
 - public void **setCodSupplier**(java.lang.String **codSupplier**)
 - **Usage**
 - * Sets up the purchase supplier code.
 - **Parameters**
 - * **codSupplier** - String, a supplier code.

- *toString*
 - public String **toString**()
 - **Usage**
 - * Converts the purchase info to a formatted String.
 - **Returns** - String, formatted String with the purchase's data.

- *update*
 - public void **update**(com.imagos.axilice.core.traceability.tree.PurchaseInfo **info**)
 - **Usage**
 - * Updates the purchase data.
 - **Parameters**
 - * **info** - PurchaseInfo, new purchase data.

8.1.7 CLASS TraceabilityTree

ValueObject with the data of a Traceability Tree.
Responsibilities:

- Store the traceability tree data.

DECLARATION

```
public class TraceabilityTree
extends java.lang.Object
```

CONSTRUCTORS

- *TraceabilityTree*
`public TraceabilityTree(java.lang.String xmlns)`
 - **Usage**
 - * Default *TraceabilityTree* constructor.
 - **Parameters**
 - * **xmlns** - String, path to the validation DTD file.

METHODS

- *getProduct*
`public ProductInfo getProduct()`
 - **Usage**
 - * Gets the root product of the traceability tree.
 - **Returns** - *ProductInfo*, root product's data.
-
- *getXmlns*
`public String getXmlns()`
 - **Usage**
 - * Gets the validation DTD file path.
 - **Returns** - String, validation DTD file path.
-
- *setProduct*
`public void setProduct(
com.imagos.axilice.core.traceability.tree.ProductInfo product)`
 - **Usage**
 - * Sets up the root product of the traceability tree.
 - **Parameters**
 - * **product** - *ProductInfo*, product data.
-
- *setXmlns*
`public void setXmlns(java.lang.String xmlns)`
 - **Usage**
 - * Sets up the validation DTD path.
 - **Parameters**
 - * **xmlns** - String, a file path.
-

- *toString*
`public String toString()`
 - **Usage**
 - * Converts the traceability tree info to a formatted String.
 - **Returns** - String, formatted String with the traceability tree's root product data.

8.1.8 CLASS TraceabilityTreeBuilder

Factory class that gives access to the Traceability Tree Construction mechanism.
Responsibilities:

- Instantiation of the Traceability Files Parser.
- Faade to the Traceability Tree Construction.

DECLARATION

```
public class TraceabilityTreeBuilder
extends java.lang.Object
```

METHODS

- *buildTraceabilityTree*
`public TraceabilityTree buildTraceabilityTree(byte [] xmlContent)`
 - **Usage**
 - * Parses a traceability tree file and builds a traceability tree based on this parsed file.
 - **Parameters**
 - * `xmlContent` - byte[], byte content stored in the traceability file.
 - **Returns** - TraceabilityTree, traceability tree corresponding to the parsed contents of the traceability file.
 - **Exceptions**
 - * **TracerException**, - in case any error takes place during the parsing of the traceability file or during the building of the traceability tree object that will store the traceability data.
-
- *buildTraceabilityTree*
`public TraceabilityTree buildTraceabilityTree(java.lang.String xmlFilename)`
 - **Usage**
 - * Parses a traceability tree file and builds a traceability tree based on this parsed file.
 - **Parameters**
 - * `xmlFilename` - String, file path to the traceability file.
 - **Returns** - TraceabilityTree, traceability tree corresponding to the parsed contents of the traceability file.
 - **Exceptions**

- * **TracerException**, - in case any error takes place during the parsing of the traceability file or during the building of the traceability tree object that will store the traceability data.

- *getInstance*

```
public static final TraceabilityTreeBuilder getInstance( )
```

- **Usage**

- * Returns the unique instance of the TraceabilityTreeBuilder class. Implements the singleton design pattern.

- **Returns** - TraceabilityTreeBuilder, unique instance of TraceabilityTreeBuilder.

- *getRevision*

```
public static String getRevision( )
```

- **Usage**

- * Returns tag Id assigned to CVS source file.

8.1.9 CLASS WarehouseInfo

ValueObject with the data of a Warehouse item.
Responsibilities:

- Store the warehouse data.

DECLARATION

```
public class WarehouseInfo
extends java.lang.Object
```

CONSTRUCTORS

- *WarehouseInfo*

```
public WarehouseInfo( )
```

- **Usage**

- * Default WarehouseInfo constructor.
-

- *WarehouseInfo*

```
public WarehouseInfo( java.lang.String codWarehouse, java.lang.String
description )
```

- **Usage**

- * WarehouseInfo constructor that initializes the warehouse values.

- **Parameters**

- * **codWarehouse** - String, warehouse code.
- * **description** - String, warehouse description.

METHODS

- *addAppcc*
public void **addAppcc**(java.lang.Object appcc)
 - **Usage**
 - * Adds a new appcc to the warehouse appccs' list.
 - **Parameters**
 - * appcc - Object, appcc's data to insert.

- *getCodWarehouse*
public String **getCodWarehouse**()
 - **Usage**
 - * Gets the warehouse code.
 - **Returns** - String, warehouse code.

- *getDescription*
public String **getDescription**()
 - **Usage**
 - * Gets the warehouse description.
 - **Returns** - String, warehouse description.

- *getLAppccs*
public ArrayList **getLAppccs**()
 - **Usage**
 - * Gets the warehouse appccs' list.
 - **Returns** - ArrayList, list of product comments.

- *getRevision*
public static String **getRevision**()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.

- *setCodWarehouse*
public void **setCodWarehouse**(java.lang.String codWarehouse)
 - **Usage**
 - * Sets up the warehouse code.
 - **Parameters**
 - * codWarehouse - String, a warehouse code.

- *setDescription*
public void **setDescription**(java.lang.String description)
 - **Usage**
 - * Sets up the warehouse description.
 - **Parameters**

* **description** - String, a description.

- *setLAppccs*

public void setLAppccs(java.util.ArrayList appccs)

- **Usage**

- * Sets up the list of appccs for the warehouse.

- **Parameters**

- * **appccs** - ArrayList, list of appccs.

- *toString*

public String toString()

- **Usage**

- * Converts the warehouse info to a formatted String.

- **Returns** - String, formatted String with the warehouses's data.

- *update*

public void update(com.imagos.axilice.core.traceability.tree.WarehouseInfo info)

- **Usage**

- * Updates the warehouse data.

- **Parameters**

- * **info** - WarehouseInfo, new warehouse data.

8.1.10 CLASS WorkcenterInfo

ValueObject with the data of a Workcenter item.
Responsibilities:

- Store the workcenter data.

DECLARATION

<pre>public class WorkcenterInfo extends java.lang.Object</pre>
--

CONSTRUCTORS

- *WorkcenterInfo*

public WorkcenterInfo()

- **Usage**

- * Default WorkcenterInfo constructor.

- *WorkcenterInfo*

public WorkcenterInfo(java.lang.String codWorkcenter, java.lang.String description)

- **Usage**
 - * WorkcenterInfo constructor that initializes the workcenter values.
- **Parameters**
 - * `codWorkcenter` -
 - * `description` -

METHODS

- *getCodWorkcenter*
 public String **getCodWorkcenter**()
 - **Usage**
 - * Gets the workcenter code.
 - **Returns** - String, workcenter code.

- *getDescription*
 public String **getDescription**()
 - **Usage**
 - * Gets the workcenter description.
 - **Returns** - String, workcenter description.

- *getRevision*
 public static String **getRevision**()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.

- *setCodWorkcenter*
 public void **setCodWorkcenter**(java.lang.String `codWorkcenter`)
 - **Usage**
 - * Sets up the workcenter code.
 - **Parameters**
 - * `codWorkcenter` - String, a workcenter code.

- *setDescription*
 public void **setDescription**(java.lang.String `description`)
 - **Usage**
 - * Sets up the workcenter description.
 - **Parameters**
 - * `description` - String, a description.

- *toString*
 public String **toString**()
 - **Usage**
 - * Converts the workcenter info to a formatted String.
 - **Returns** - String, formatted String with the workcenter's data.

- *update*

```
public void update( com.imagos.axilice.core.traceability.tree.WorkcenterInfo
info )
```

- **Usage**

- * Updates the workcenter data.

- **Parameters**

- * **info** - WorkcenterInfo, new workcenter data.

8.1.11 CLASS WorkerInfo

ValueObject with the data of a Worker item.
Responsibilities:

- Store the worker data.

DECLARATION

```
public class WorkerInfo
extends java.lang.Object
```

CONSTRUCTORS

- *WorkerInfo*

```
public WorkerInfo( )
```

- **Usage**

- * Default WorkerInfo constructor

- *WorkerInfo*

```
public WorkerInfo( java.lang.String codWorker, java.lang.String
description )
```

- **Usage**

- * WorkerInfo constructor that initializes the worker values.

- **Parameters**

- * **codWorker** - String, worker code.

- * **description** - String, worker name.

METHODS

- *getCodWorker*

```
public String getCodWorker( )
```

- **Usage**

- * Gets the worker code.

- **Returns** - String, worker code.

- *getDescription*
 public String getDescription()
 - **Usage**
 - * Gets the worker name.
 - **Returns** - String, worker name.

- *getRevision*
 public static String getRevision()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.

- *setCodWorker*
 public void setCodWorker(java.lang.String codWorker)
 - **Usage**
 - * Sets up the worker code.
 - **Parameters**
 - * codWorker - String, a worker code.

- *setDescription*
 public void setDescription(java.lang.String description)
 - **Usage**
 - * Sets up the worker name.
 - **Parameters**
 - * description - String, a name.

- *toString*
 public String toString()
 - **Usage**
 - * Converts the worker info to a formatted String.
 - **Returns** - String, formatted String with the worker's data.

- *update*
 public void update(com.imagos.axilice.core.traceability.tree.WorkerInfo info)
 - **Usage**
 - * Updates the worker data.
 - **Parameters**
 - * info - WorkerInfo, new worker data.

Chapter 9

Package com.imagos.axilice.core.warehouse

<i>Package Contents</i>	<i>Page</i>
<hr/>	
Interfaces	
WarehouseManager 245	
<i>...no description...</i>	
Classes	
LotOT 246	
<i>Store class to hold the data of a warehouse.</i>	
MeasureUnitOT 249	
<i>Store class to hold the data of a measure unit.</i>	
ProductGroupOT 252	
<i>Store class to hold the data of a product group.</i>	
ProductOT 255	
<i>Store class to hold the data of a product.</i>	
ProductStockOT 265	
<i>Store class to hold the data of a warehouse.</i>	
StockOT 268	
<i>Store class to hold the data of a warehouse.</i>	
WarehouseOT 270	
<i>Store class to hold the data of a warehouse.</i>	
<hr/>	

9.1 Interfaces

9.1.1 INTERFACE WarehouseManager

DECLARATION

```
public interface WarehouseManager
implements java.rmi.Remote
```

METHODS

- *delMeasureUnit*
public OperationResult delMeasureUnit(java.lang.Long idMeasureUnit)
- *delMeasureUnits*
public OperationResult delMeasureUnits(java.util.ArrayList lSelMeasureUnits)
- *delProduct*
public OperationResult delProduct(java.lang.Long idProduct)
- *delProductGroup*
public OperationResult delProductGroup(java.lang.Long idGroup)
- *delProductGroups*
public OperationResult delProductGroups(java.util.ArrayList lSelGroups)
- *delProducts*
public OperationResult delProducts(java.util.ArrayList lSelProducts)
- *getMeasureUnit*
public MeasureUnitOT getMeasureUnit(java.lang.Long idMeasureUnit)
- *getNextProductGroupId*
public long getNextProductGroupId()
- *getProduct*
public ProductOT getProduct(java.lang.Long idProduct)
- *getProductGroup*
public ProductGroupOT getProductGroup(java.lang.Long idGroup)
- *queryMeasureUnits*
public List queryMeasureUnits(java.lang.String fieldId, java.lang.String value)
- *queryProductGroups*
public List queryProductGroups(java.lang.String fieldId, java.lang.String value)

- *queryProducts*

```
public List queryProducts( java.lang.String fieldId, java.lang.String value )
```

- *resetNextProductId*

```
public void resetNextProductId( long idGroup )
```

- *saveMeasureUnit*

```
public OperationResult saveMeasureUnit(
com.imagos.axilice.core.warehouse.MeasureUnitOT measureUnit )
```

- *saveProduct*

```
public OperationResult saveProduct(
com.imagos.axilice.core.warehouse.ProductOT product )
```

- *saveProductGroup*

```
public OperationResult saveProductGroup(
com.imagos.axilice.core.warehouse.ProductGroupOT group )
```

9.2 Classes

9.2.1 CLASS LotOT

Store class to hold the data of a warehouse.
Responsibilities:

- Store of warehouse data for exchanging data among presentation, business and data access layers.

DECLARATION

```
public class LotOT
extends com.imagos.axilice.core.common.BaseVO
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idLot
 - Internal id of the lot.
- private String description
 - Description of the lot.
- private String number
 - Lot number of the lot.
- private Date creationDate
 - Creation date of the lot.

- private Date expirationDate
 - Expiration date of the lot.
- private BigDecimal quantity
 - Initial quantity of products belonging to the lot.
- private ProductOT product
 - Product of the lot.
- private long idProduct
 - Internal id of the product (Foreign key).
- private Integer version
 - Version of the product used by Hibernate for optimistic concurrency control.

CONSTRUCTORS

- *LotOT*
`public LotOT()`

METHODS

- *equals*
`public boolean equals(java.lang.Long id)`
- *equals*
`public boolean equals(java.lang.Object other)`
 - Usage
 - * Implementation of the equals method provided by the serializable interface.
- *getCreationDate*
`public Date getCreationDate()`
- *getDescription*
`public String getDescription()`
- *getEntityId*
`public String getEntityId()`
- *getExpirationDate*
`public Date getExpirationDate()`
- *getIdLot*
`public long getIdLot()`
- *getInternalId*
`public long getInternalId()`
- *getInternalValue*
`public String getInternalValue()`

- *getNumber*
public String **getNumber**()
- *getProduct*
public ProductOT **getProduct**()
- *getProductName*
public String **getProductName**()
- *getQuantity*
public BigDecimal **getQuantity**()
- *getRevision*
public static String **getRevision**()
- Usage
 - * Returns tag Id assigned to CVS source file.
- *getVersion*
public Integer **getVersion**()
- *hashCode*
public int **hashCode**()
- Usage
 - * Implementation of the hashCode method provided by the serializable interface.
- *setCreationDate*
public void **setCreationDate**(java.util.Date **creationDate**)
- *setDescription*
public void **setDescription**(java.lang.String **description**)
- *setExpirationDate*
public void **setExpirationDate**(java.util.Date **expirationDate**)
- *setIdLot*
public void **setIdLot**(long **idLot**)
- *setNumber*
public void **setNumber**(java.lang.String **number**)
- *setProduct*
public void **setProduct**(com.imagos.axilice.core.warehouse.ProductOT **product**)
- *setQuantity*
public void **setQuantity**(java.math.BigDecimal **quantity**)
- *setVersion*
public void **setVersion**(java.lang.Integer **version**)
- *toString*
public String **toString**()
- Usage
 - * Implementation of the toString() method to show the data of the ProductOT formatted.

METHODS INHERITED FROM CLASS com.imagos.axilice.core.common.BaseVO

(in 31.2.2, page 740)

- *equals*
public abstract boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *format*
public String format(java.lang.String fieldId, java.util.ArrayList lFields)
- *formatValue*
public String formatValue(java.lang.String fieldId)
- *getEntityId*
public abstract String getEntityId()
- *getInternalId*
public abstract long getInternalId()
- *getInternalValue*
public abstract String getInternalValue()
- *getVisibleFieldsCount*
public int getVisibleFieldsCount(java.util.ArrayList lFields)
 - **Usage**
 - * Calculates the number of visible fields of the entity.
 - **Returns** - int, visible fields number; 1 in case no visible fields have been defined.

9.2.2 CLASS MeasureUnitOT

Store class to hold the data of a measure unit.
Responsibilities:

Store of measure unit data for exchanging data among presentation, business and data access layers.

.

DECLARATION

```
public class MeasureUnitOT
extends com.imagos.axilice.core.common.BaseVO
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idMeasureUnit
 - Internal id of the measure unit.
- private String description
 - Description of the measure unit.
- private Integer version
 - Version of the measure unit used by Hibernate for optimistic concurrency control.

CONSTRUCTORS

- *MeasureUnitOT*
`public MeasureUnitOT()`
 - **Usage**
 - * Default MeasureUnitOT constructor.

METHODS

- *equals*
`public boolean equals(java.lang.Long id)`
- *equals*
`public boolean equals(java.lang.Object obj)`
- *getDescription*
`public String getDescription()`
 - **Usage**
 - * Gets the description of the measure unit.
 - **Returns** - String, description of the group.
- *getEntityId*
`public String getEntityId()`
- *getIdMeasureUnit*
`public long getIdMeasureUnit()`
 - **Usage**
 - * Gets the identifier of the measure unit.
 - **Returns** - String, identifier of the measure unit.
- *getInternalId*
`public long getInternalId()`
- *getInternalValue*
`public String getInternalValue()`
- *getRevision*
`public static String getRevision()`
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
- *getVersion*
`public Integer getVersion()`
 - **Usage**
 - * Gets the version of the measure unit used by Hibernate for optimistic concurrency control.
 - **Returns** - Integer, version of the group.

-
- *hashCode*
public int hashCode()
 - *setDescription*
public void setDescription(java.lang.String description)
 - **Usage**
 - * Sets up the description of the measure unit.
 - **Parameters**
 - * **description** - String, description of the group.
 - *setIdMeasureUnit*
public void setIdMeasureUnit(long idMeasureUnit)
 - **Usage**
 - * Sets up the identifier of the measure unit.
 - **Parameters**
 - * **idMeasureUnit** - String, identifier of the measure unit.
 - *setVersion*
public void setVersion(java.lang.Integer version)
 - **Usage**
 - * Sets up the version of the measure unit used by Hibernate for optimistic concurrency control.
 - **Parameters**
 - * **version** - Integer, version of the group.
 - *toString*
public String toString()
 - **Usage**
 - * Implementation of the toString() method to show the data of the ExtFieldOT formatted.
-

METHODS INHERITED FROM CLASS com.imagos.axilice.core.common.BaseVO

(in 31.2.2, page 740)

- *equals*
public abstract boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *format*
public String format(java.lang.String fieldId, java.util.ArrayList lFields)
- *formatValue*
public String formatValue(java.lang.String fieldId)
- *getEntityId*
public abstract String getEntityId()

- *getInternalId*
`public abstract long getInternalId()`
- *getInternalValue*
`public abstract String getInternalValue()`
- *getVisibleFieldsCount*
`public int getVisibleFieldsCount(java.util.ArrayList IFields)`
 - **Usage**
 - * Calculates the number of visible fields of the entity.
 - **Returns** - int, visible fields number; 1 in case no visible fields have been defined.

9.2.3 CLASS ProductGroupOT

Store class to hold the data of a product group.
Responsibilities:

Store of product group data for exchanging data among presentation, business and data access layers.

.

DECLARATION

```
public class ProductGroupOT
extends com.imagos.axilice.core.common.BaseVO
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idGroup
 - Internal id of the product group.
- private String description
 - Description of the product group.
- private ExtTableOT extTable
 - Extension table defined for the group of products.
- private Integer version
 - Version of the product group used by Hibernate for optimistic concurrency control.

CONSTRUCTORS

- *ProductGroupOT*
`public ProductGroupOT()`
 - **Usage**
 - * Default ProductGroupOT constructor.

METHODS

- *equals*
public boolean equals(java.lang.Long id)
 - *equals*
public boolean equals(java.lang.Object other)
 - **Usage**
 - * Implementation of the equals method provided by the serializable interface.
-
- *getDescription*
public String getDescription()
 - **Usage**
 - * Gets the description of the product group.
 - **Returns** - String, description of the group.
-
- *getEntityId*
public String getEntityId()
 - *getExtTable*
public ExtTableOT getExtTable()
 - **Usage**
 - * Gets the extension table for the product group.
 - **Returns** - ExtTableOT, extension table of the group.
-
- *getIdGroup*
public long getIdGroup()
 - **Usage**
 - * Gets the identifier of the product group.
 - **Returns** - String, identifier of the group.
-
- *getInternalId*
public long getInternalId()
 - *getInternalValue*
public String getInternalValue()
 - *getRevision*
public static String getRevision()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
-
- *getVersion*
public Integer getVersion()
 - **Usage**
 - * Gets the version of the product group used by Hibernate for optimistic concurrency control.

– **Returns** - Integer, version of the group.

- *hashCode*

public int hashCode()

– **Usage**

* Implementation of the hashCode method provided by the serializable interface.

- *setDescription*

public void setDescription(java.lang.String description)

– **Usage**

* Sets up the description of the product group.

– **Parameters**

* **description** - String, description of the group.

- *setExtTable*

public void setExtTable(com.imagos.axilice.core.extTables.ExtTableOT
extTable)

– **Usage**

* Sets up the extension table for the product group.

– **Parameters**

* **extTable** - ExtTableOT, extension table of the group.

- *setIdGroup*

public void setIdGroup(long idGroup)

– **Usage**

* Sets up the identifier of the product group.

– **Parameters**

* **idGroup** - String, identifier of the group.

- *setVersion*

public void setVersion(java.lang.Integer version)

– **Usage**

* Sets up the version of the product group used by Hibernate for optimistic concurrency control.

– **Parameters**

* **version** - Integer, version of the group.

- *toString*

public String toString()

– **Usage**

* Implementation of the toString() method to show the data of the ExtFieldOT formatted.

METHODS INHERITED FROM CLASS com.imagos.axilice.core.common.BaseVO

(in 31.2.2, page 740)

- *equals*
public abstract boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *format*
public String format(java.lang.String fieldId, java.util.ArrayList IFields)
- *formatValue*
public String formatValue(java.lang.String fieldId)
- *getEntityId*
public abstract String getEntityId()
- *getInternalId*
public abstract long getInternalId()
- *getInternalValue*
public abstract String getInternalValue()
- *getVisibleFieldsCount*
public int getVisibleFieldsCount(java.util.ArrayList IFields)
 - Usage
 - * Calculates the number of visible fields of the entity.
 - Returns - int, visible fields number; 1 in case no visible fields have been defined.

9.2.4 CLASS ProductOT

Store class to hold the data of a product.
Responsibilities:

Store of product data for exchanging data among presentation, business and data access layers.

.

DECLARATION

```
public class ProductOT
extends com.imagos.axilice.core.common.BaseVO
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idProduct
 - Internal id of the product.
- private String description
 - Description of the product.
- private String reference

- Reference of the product.
- private String barcode
 - Barcode number of the product.
- private BarcodeTypeOT barcodeType
 - Barcode type of the product.
- private BigDecimal price
 - Price of the product.
- private BigDecimal averageCost
 - Average cost made out of all the suppliers costs.
- private boolean allowNoStock
 - Flag to show if sales with no stock are allowed for the product.
- private long idGroup
 - Internal id of the product group (Foreign key).
- private long idVatType
 - Internal id of the product VAT type (Foreign key).
- private long idBarcodeType
 - Internal id of the product barcode type (Foreign key).
- private ProductGroupOT group
 - Group of the product.
- private VatTypeOT vatType
 - Vat type applied to the product.
- private Set images
 - List of product images.
- private Set measureUnits
 - List of product supported measure units.
- private Set prices
 - List of prices available for the product.
- private Set stocks
 - List of stocks available for the product.
- private byte trData
 - Traceability information of the product. It's the content bytes of the traceability tree built for the product.
- private Integer version
 - Version of the product used by Hibernate for optimistic concurrency control.

CONSTRUCTORS

- *ProductOT*

public ProductOT()

- **Usage**

- * Default ProductOT constructor.

METHODS

- *addImage*

public void addImage(com.imagos.axilice.core.common.ImageOT image)

- **Usage**

- * Adds a new image to the set of product images.

- **Parameters**

- * image - ImageOT, data of the new product image.

- *addMeasureUnit*

**public void addMeasureUnit(
com.imagos.axilice.core.warehouse.MeasureUnitOT measureUnit)**

- **Usage**

- * Adds a new measure unit to the set of supported product measure units.

- **Parameters**

- * measureUnit - MeasureUnitOT, data of the new product measure unit.

- *addPrice*

public void addPrice(com.imagos.axilice.core.common.PriceListLineOT pll)

- **Usage**

- * Adds a new price to the set of supported product prices.

- **Parameters**

- * pll - PriceListLineOT, data of the new product price line.

- *delImage*

public void delImage(com.imagos.axilice.core.common.ImageOT image)

- **Usage**

- * Removes an image from the set of product images.

- **Parameters**

- * image - ImageOT, data of the removed product image.

- *delMeasureUnit*

**public void delMeasureUnit(com.imagos.axilice.core.warehouse.MeasureUnitOT
measureUnit)**

- **Usage**

- * Removes a measure unit from the set of supported product measure units.

– **Parameters**

* `measureUnit` - MeasureUnitOT, data of the product measure unit.

• *delPrice*

`public void delPrice(com.imagos.axilice.core.common.PriceListLineOT pll)`

– **Usage**

* Deletes a price from the set of supported product prices.

– **Parameters**

* `pll` - PriceListLineOT, data of the product price line.

• *equals*

`public boolean equals(java.lang.Long id)`

• *equals*

`public boolean equals(java.lang.Object obj)`

• *getAverageCost*

`public BigDecimal getAverageCost()`

– **Usage**

* Gets the average cost of the product among all the suppliers.

– **Returns** - BigDecimal, average cost of the product.

• *getBarcode*

`public String getBarcode()`

– **Usage**

* Gets the barcode of the product.

– **Returns** - String, barcode of the product.

• *getBarcodeType*

`public BarcodeTypeOT getBarcodeType()`

– **Usage**

* Gets the barcode's type of the product.

– **Returns** - BarcodeTypeOT, data of the barcode's type of the product.

• *getDescription*

`public String getDescription()`

– **Usage**

* Gets the description of the product.

– **Returns** - String, description of the product.

• *getEntityId*

`public String getEntityId()`

• *getGroup*

`public ProductGroupOT getGroup()`

– **Usage**

* Gets the group of the product.

- **Returns** - ProductGroupOT, data of the group of the product.

- *getIdBarcodeType*

public long getIdBarcodeType()

- **Usage**
 - * Gets the identifier of the product barcode type.
- **Returns** - long, identifier of the product barcode type (foreign key).

- *getIdGroup*

public long getIdGroup()

- **Usage**
 - * Gets the identifier of the product group.
- **Returns** - long, identifier of the product group (foreign key).

- *getIdProduct*

public long getIdProduct()

- **Usage**
 - * Gets the identifier of the product.
- **Returns** - String, identifier of the product.

- *getIdVatType*

public long getIdVatType()

- **Usage**
 - * Gets the identifier of the product VAT type.
- **Returns** - long, identifier of the product VAT type (foreign key).

- *getImages*

public Set getImages()

- **Usage**
 - * Gets the set of product images.
- **Returns** - Set, product images.

- *getInternalId*

public long getInternalId()

- *getInternalValue*

public String getInternalValue()

- *getMeasureUnits*

public Set getMeasureUnits()

- **Usage**
 - * Gets the set of supported product measure units.
- **Returns** - Set, measure units.

- *getPrice*

public BigDecimal getPrice()

- **Usage**
 - * Gets the price of the product.
 - **Returns** - String, price of the product.
-
- *getPrices*
public Set getPrices()
 - **Usage**
 - * Gets the set of supported product prices.
 - **Returns** - Set, prices list.
-
- *getReference*
public String getReference()
 - **Usage**
 - * Gets the reference of the product.
 - **Returns** - String, reference of the product.
-
- *getRevision*
public static String getRevision()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
-
- *getStocks*
public Set getStocks()
 - **Usage**
 - * Gets the set of available product stocks.
 - **Returns** - Set, stocks list.
-
- *getTrData*
public byte getTrData()
 - **Usage**
 - * Gets the product's traceability data.
 - **Returns** - byte[], product's traceability data.
-
- *getVatType*
public VatTypeOT getVatType()
 - **Usage**
 - * Gets the VAT tax that is applied to the product.
 - **Returns** - VatTypeOT, data of the VAT tax type of the product.
-
- *getVersion*
public Integer getVersion()
 - **Usage**
 - * Gets the version of the product used by Hibernate for optimistic concurrency control.
 - **Returns** - Integer, version of the group.

- *hashCode*

```
public int hashCode( )
```

- *isAllowNoStock*

```
public boolean isAllowNoStock( )
```

- **Usage**

- * Gets the flag to show that the product can be sold even if there is no stock available in the warehouse.

- **Returns** - boolean, True if sales with no stock are admitted; False in any other case.

- *setAllowNoStock*

```
public void setAllowNoStock( boolean allowNoStock )
```

- **Usage**

- * Sets up the flag to show that the product can be sold even if there is no stock available in the warehouse.

- **Parameters**

- * **allowNoStock** - boolean, True if sales with no stock are admitted; False in any other case.

- *setAverageCost*

```
public void setAverageCost( java.math.BigDecimal averageCost )
```

- **Usage**

- * Sets up the average cost of the product among all the suppliers.

- **Parameters**

- * **averageCost** - BigDecimal, average cost of the product.

- *setBarcode*

```
public void setBarcode( java.lang.String barcode )
```

- **Usage**

- * Sets up the barcode of the product.

- **Parameters**

- * **barcode** - String, barcode of the product.

- *setBarcodeType*

```
public void setBarcodeType( com.imagos.axilice.core.common.BarcodeTypeOT  
barcodeType )
```

- **Usage**

- * Sets up the barcode's type of the product.

- **Parameters**

- * **barcodeType** - BarcodeTypeOT, data of the barcode's type of the product.

- *setDescription*

```
public void setDescription( java.lang.String description )
```

- **Usage**

- * Sets up the description of the product.
 - **Parameters**
 - * **description** - String, description of the product.

- *setGroup*
`public void setGroup(com.imagos.axilice.core.warehouse.ProductGroupOT group)`
 - **Usage**
 - * Sets up the group of the product.
 - **Parameters**
 - * **group** - ProductGroupOT, data of the group of the product.

 - *setIdBarcodeType*
`public void setIdBarcodeType(long idBarcodeType)`
 - **Usage**
 - * Sets up the identifier of the product barcode type.
 - **Parameters**
 - * **idBarcodeType** - long, identifier of the product barcode type (foreign key).

 - *setIdGroup*
`public void setIdGroup(long idGroup)`
 - **Usage**
 - * Sets up the identifier of the product group.
 - **Parameters**
 - * **idGroup** - long, identifier of the product group (foreign key).

 - *setIdProduct*
`public void setIdProduct(long idProduct)`
 - **Usage**
 - * Sets up the identifier of the product.
 - **Parameters**
 - * **idGroup** - String, identifier of the product.

 - *setIdVatType*
`public void setIdVatType(long idVatType)`
 - **Usage**
 - * Sets up the identifier of the product VAT type.
 - **Parameters**
 - * **idVatType** - long, identifier of the product VAT type(foreign key).

 - *setImages*
`public void setImages(java.util.Set images)`
 - **Usage**
 - * Sets up the set of product images.

- **Parameters**
 - * `images` - Set, product images.

- *setMeasureUnits*

```
public void setMeasureUnits( java.util.Set  measureUnits )
```

 - **Usage**
 - * Sets up the set of supported product measure units.
 - **Parameters**
 - * `measureUnits` - Set, measure units.

 - *setPrice*

```
public void setPrice( java.math.BigDecimal  price )
```

 - **Usage**
 - * Sets up the price of the product.
 - **Parameters**
 - * `price` - String, price of the product.

 - *setPrices*

```
public void setPrices( java.util.Set  prices )
```

 - **Usage**
 - * Sets up the set of supported product prices.
 - **Parameters**
 - * `prices` - Set, prices list.

 - *setReference*

```
public void setReference( java.lang.String  reference )
```

 - **Usage**
 - * Sets up the reference of the product.
 - **Parameters**
 - * `reference` - String, reference of the product.

 - *setStocks*

```
public void setStocks( java.util.Set  stocks )
```

 - **Usage**
 - * Sets up the set of available product stocks.
 - **Parameters**
 - * `stocks` - Set, stocks list.

 - *setTrData*

```
public void setTrData( byte [] trData )
```

 - **Usage**
 - * Sets up the product's traceability data.
 - **Parameters**
 - * `trData` - byte[], product's traceability data.

- *setVatType*
`public void setVatType(com.imagos.axilice.core.common.VatTypeOT vatType)`
 - **Usage**
 - * Sets up the VAT tax that is applied to the product.
 - **Parameters**
 - * `vatType` - VatTypeOT, data of the VAT tax type of the product.

- *setVersion*
`public void setVersion(java.lang.Integer version)`
 - **Usage**
 - * Sets up the version of the product used by Hibernate for optimistic concurrency control.
 - **Parameters**
 - * `version` - Integer, version of the group.

- *toString*
`public String toString()`
 - **Usage**
 - * Implementation of the toString() method to show the data of the ProductOT formatted.

METHODS INHERITED FROM CLASS com.imagos.axilice.core.common.BaseVO

(in 31.2.2, page 740)

- *equals*
`public abstract boolean equals(java.lang.Long id)`

- *equals*
`public boolean equals(java.lang.Object obj)`

- *format*
`public String format(java.lang.String fieldId, java.util.ArrayList IFields)`

- *formatValue*
`public String formatValue(java.lang.String fieldId)`

- *getEntityId*
`public abstract String getEntityId()`

- *getInternalId*
`public abstract long getInternalId()`

- *getInternalValue*
`public abstract String getInternalValue()`

- *getVisibleFieldsCount*
`public int getVisibleFieldsCount(java.util.ArrayList IFields)`
 - **Usage**
 - * Calculates the number of visible fields of the entity.
 - **Returns** - int, visible fields number; 1 in case no visible fields have been defined.

9.2.5 CLASS ProductStockOT

Store class to hold the data of a warehouse.
Responsibilities:

Store of warehouse data for exchanging data among presentation, business and data access layers.

.

DECLARATION

```
public class ProductStockOT
extends com.imagos.axilice.core.common.BaseVO
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idStock
 - Internal id of the stock.
- private Date lastUpdate
 - Last update stock's date.
- private BigDecimal stock
 - Quantity of products in stock.
- private LotOT lot
 - Lot of the stock.
- private WarehouseOT warehouse
 - Warehouse of the stock.
- private ProductOT product
 - Product of the stock.
- private Integer version
 - Version of the stock used by Hibernate for optimistic concurrency control.

CONSTRUCTORS

- *ProductStockOT*
public **ProductStockOT**()

METHODS

- *equals*
public boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object other)
- Usage
 - * Implementation of the equals method provided by the serializable interface.
- *getEntityId*
public String getEntityId()
- *getIdStock*
public long getIdStock()
- *getInternalId*
public long getInternalId()
- *getInternalValue*
public String getInternalValue()
- *getLastUpdate*
public Date getLastUpdate()
- *getLot*
public LotOT getLot()
- *getLotDescription*
public String getLotDescription()
- *getLotNumber*
public String getLotNumber()
- *getProduct*
public ProductOT getProduct()
- *getProductDescription*
public String getProductDescription()
- *getRevision*
public static String getRevision()
- Usage
 - * Returns tag Id assigned to CVS source file.
- *getStock*
public BigDecimal getStock()
- *getVersion*
public Integer getVersion()
- *getWarehouse*
public WarehouseOT getWarehouse()

- *getWarehouseDescription*
public String getWarehouseDescription()
- *hashCode*
public int hashCode()
 - Usage
 - * Implementation of the hashCode method provided by the serializable interface.
- *setIdStock*
public void setIdStock(long idStock)
- *setLastUpdate*
public void setLastUpdate(java.util.Date lastUpdate)
- *setLot*
public void setLot(com.imagos.axilice.core.warehouse.LotOT lot)
- *setProduct*
public void setProduct(com.imagos.axilice.core.warehouse.ProductOT product)
- *setStock*
public void setStock(java.math.BigDecimal stock)
- *setVersion*
public void setVersion(java.lang.Integer version)
- *setWarehouse*
public void setWarehouse(com.imagos.axilice.core.warehouse.WarehouseOT warehouse)
- *toString*
public String toString()
 - Usage
 - * Implementation of the toString() method to show the data of the ProductOT formatted.

METHODS INHERITED FROM CLASS com.imagos.axilice.core.common.BaseVO

(in 31.2.2, page 740)

- *equals*
public abstract boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *format*
public String format(java.lang.String fieldId, java.util.ArrayList lFields)
- *formatValue*
public String formatValue(java.lang.String fieldId)
- *getEntityId*
public abstract String getEntityId()
- *getInternalId*
public abstract long getInternalId()

- *getInternalValue*
`public abstract String getInternalValue()`
- *getVisibleFieldsCount*
`public int getVisibleFieldsCount(java.util.ArrayList IFields)`
 - **Usage**
 - * Calculates the number of visible fields of the entity.
 - **Returns** - int, visible fields number; 1 in case no visible fields have been defined.

9.2.6 CLASS StockOT

Store class to hold the data of a warehouse.
Responsibilities:

Store of warehouse data for exchanging data among presentation, business and data access layers.

.

DECLARATION

```
public class StockOT
extends com.imagos.axilice.core.common.BaseVO
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idStock
 - Internal id of the stock.
- private Date lastUpdate
 - Last update stock's date.
- private BigDecimal stock
 - Quantity of products in stock.
- private LotOT lot
 - Lot of the stock.
- private WarehouseOT warehouse
 - Warehouse of the stock.
- private Integer version
 - Version of the stock used by Hibernate for optimistic concurrency control.

CONSTRUCTORS

- *StockOT*
`public StockOT()`

METHODS

• *equals*

public boolean equals(java.lang.Long id)

• *equals*

public boolean equals(java.lang.Object other)

– Usage

* Implementation of the equals method provided by the serializable interface.

• *getEntityId*

public String getEntityId()

• *getIdStock*

public long getIdStock()

• *getInternalId*

public long getInternalId()

• *getInternalValue*

public String getInternalValue()

• *getLastUpdate*

public Date getLastUpdate()

• *getLot*

public LotOT getLot()

• *getLotName*

public String getLotName()

• *getRevision*

public static String getRevision()

– Usage

* Returns tag Id assigned to CVS source file.

• *getStock*

public BigDecimal getStock()

• *getVersion*

public Integer getVersion()

• *getWarehouse*

public WarehouseOT getWarehouse()

• *hashCode*

public int hashCode()

– Usage

* Implementation of the hashCode method provided by the serializable interface.

• *setIdStock*

public void setIdStock(long idStock)

- *setLastUpdate*
public void setLastUpdate(java.util.Date lastUpdate)
- *setLot*
public void setLot(com.imagos.axilice.core.warehouse.LotOT lot)
- *setStock*
public void setStock(java.math.BigDecimal stock)
- *setVersion*
public void setVersion(java.lang.Integer version)
- *setWarehouse*
public void setWarehouse(com.imagos.axilice.core.warehouse.WarehouseOT warehouse)
- *toString*
public String toString()

– Usage

- * Implementation of the toString() method to show the data of the ProductOT formatted.

METHODS INHERITED FROM CLASS com.imagos.axilice.core.common.BaseVO

(in 31.2.2, page 740)

- *equals*
public abstract boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *format*
public String format(java.lang.String fieldId, java.util.ArrayList lFields)
- *formatValue*
public String formatValue(java.lang.String fieldId)
- *getEntityId*
public abstract String getEntityId()
- *getInternalId*
public abstract long getInternalId()
- *getInternalValue*
public abstract String getInternalValue()
- *getVisibleFieldsCount*
public int getVisibleFieldsCount(java.util.ArrayList lFields)

– Usage

- * Calculates the number of visible fields of the entity.

– **Returns** - int, visible fields number; 1 in case no visible fields have been defined.

9.2.7 CLASS WarehouseOT

Store class to hold the data of a warehouse.
Responsibilities:

Store of warehouse data for exchanging data among presentation, business and data access layers.

.

DECLARATION

```
public class WarehouseOT
extends com.imagos.axilice.core.common.BaseVO
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idWarehouse
 - Internal id of the warehouse.
- private String description
 - Description of the warehouse.
- private String phone
 - Phone number of the warehouse.
- private String fax
 - Fax number of the warehouse.
- private String contact
 - Contact person of the warehouse.
- private String notes
 - Other comments or notes about the warehouse.
- private AddressOT address
 - Address of the warehouse.
- private long idAddress
 - Internal id of the address (Foreign key).
- private Set lots
 - Set of product lots that are in the stock of the warehouse.
- private Integer version
 - Version of the product used by Hibernate for optimistic concurrency control.

CONSTRUCTORS

- *WarehouseOT*
public WarehouseOT()

METHODS

- *equals*
public boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object other)
 - Usage
 - * Implementation of the equals method provided by the serializable interface.
- *getAddress*
public AddressOT getAddress()
- *getContact*
public String getContact()
- *getDescription*
public String getDescription()
- *getEntityId*
public String getEntityId()
- *getFax*
public String getFax()
- *getIdAddress*
public long getIdAddress()
- *getIdWarehouse*
public long getIdWarehouse()
- *getInternalId*
public long getInternalId()
- *getInternalValue*
public String getInternalValue()
- *getLots*
public Set getLots()
- *getNotes*
public String getNotes()
- *getPhone*
public String getPhone()
- *getRevision*
public static String getRevision()
 - Usage
 - * Returns tag Id assigned to CVS source file.
- *getVersion*
public Integer getVersion()
 - Usage

* Removes a lot from the stock of the warehouse.

– **Parameters**

* *lot* - LotOT, data of the lot.

• *hashCode*

public int hashCode()

– **Usage**

* Implementation of the hashCode method provided by the serializable interface.

• *setAddress*

public void setAddress(com.imagos.axilice.core.partners.AddressOT *address*)

• *setContact*

public void setContact(java.lang.String *contact*)

• *setDescription*

public void setDescription(java.lang.String *description*)

• *setFax*

public void setFax(java.lang.String *fax*)

• *setIdAddress*

public void setIdAddress(long *idAddress*)

• *setIdWarehouse*

public void setIdWarehouse(long *idWarehouse*)

• *setLots*

public void setLots(java.util.Set *lots*)

• *setNotes*

public void setNotes(java.lang.String *notes*)

• *setPhone*

public void setPhone(java.lang.String *phone*)

• *setVersion*

public void setVersion(java.lang.Integer *version*)

• *toString*

public String toString()

– **Usage**

* Implementation of the toString() method to show the data of the ProductOT formatted.

METHODS INHERITED FROM CLASS `com.imagos.axilice.core.common.BaseVO`

(in 31.2.2, page 740)

- *equals*
public abstract boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *format*
public String format(java.lang.String fieldId, java.util.ArrayList IFields)
- *formatValue*
public String formatValue(java.lang.String fieldId)
- *getEntityId*
public abstract String getEntityId()
- *getInternalId*
public abstract long getInternalId()
- *getInternalValue*
public abstract String getInternalValue()
- *getVisibleFieldsCount*
public int getVisibleFieldsCount(java.util.ArrayList IFields)

– Usage

* Calculates the number of visible fields of the entity.

– **Returns** - int, visible fields number; 1 in case no visible fields have been defined.

Chapter 10

Package com.imagos.axilice.server.util.db

<i>Package Contents</i>	<i>Page</i>
<hr/>	
Classes	
CriteriaBean	276
<i>This bean class holds the data belonging to a query criteria for the WHERE statement in a generic SQL query performed in the database.</i>	
CriteriaQueue	278
<i>...no description...</i>	
HibernateUtil	279
<i>Implementation of an Hibernate utility class to encapsulate the Hibernate framework.</i>	
SQLGenerator	280
<i>Business Logic class that implements an engine for generating SQL statements.</i>	
<hr/>	

10.1 Classes

10.1.1 CLASS CriteriaBean

This bean class holds the data belonging to a query criteria for the WHERE statement in a generic SQL query performed in the database.

DECLARATION

```
public class CriteriaBean
extends java.lang.Object
```

CONSTRUCTORS

- *CriteriaBean*

```
public CriteriaBean( )
```
- *CriteriaBean*

```
public CriteriaBean( java.lang.String fieldId, java.util.ArrayList
    lFieldValues, java.lang.String fieldOperator, java.lang.String
    fieldLogicOperator, java.lang.String fieldBracketsPrefix, java.lang.String
    fieldBracketsSuffix )
```
- *CriteriaBean*

```
public CriteriaBean( java.lang.String fieldId, java.lang.Object fieldValue,
    java.lang.String fieldOperator, java.lang.String fieldLogicOperator,
    java.lang.String fieldBracketsPrefix, java.lang.String fieldBracketsSuffix )
```

METHODS

- *getCriteria*

```
public Criterion getCriteria( )
```

 - **Usage**
 - * Builds an Hibernate Criterion instance with the SQL WHERE statement corresponding to the search criteria.
 - **Returns** - Criterion, Hibernate Criterion built for the searching criteria.
- *getCriteriaAsSql*

```
public String getCriteriaAsSql( boolean isFirstCriteria )
```

 - **Usage**
 - * Returns the SQL WHERE statement string value corresponding to the Criteria.
 - **Parameters**
 - * **isFirstCriteria** - boolean, True in case the criteria is the first of the WHERE statement.
 - **Returns** - String, sql string value.

- *getFieldBracketsPrefix*

public String getFieldBracketsPrefix()
- *getFieldBracketsSuffix*

public String getFieldBracketsSuffix()
- *getFieldId*

public String getFieldId()
- *getFieldLogicOperator*

public String getFieldLogicOperator()
- *getFieldOperator*

public String getFieldOperator()
- *getFieldValue*

public Object getFieldValue()
 - **Usage**
 - * Gets the unique value of the criteria for unitary operators.
 - **Returns** - Object, value for the criteria.

- *getLFieldValues*

public ArrayList getLFieldValues()
- *getOrderCriteriaAsSql*

public String getOrderCriteriaAsSql(boolean isFirstCriteria)
 - **Usage**
 - * Returns the SQL ORDER BY statement string value corresponding to the Criteria.
 - **Parameters**
 - * **isFirstCriteria** - boolean, True in case the criteria is the first of the ORDER BY statement.
 - **Returns** - String, sql string value.

- *getRevision*

public static String getRevision()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.

- *isFieldNormalized*

public boolean isFieldNormalized()
- *setFieldBracketsPrefix*

public void setFieldBracketsPrefix(java.lang.String fieldBracketsPrefix)
- *setFieldBracketsSuffix*

public void setFieldBracketsSuffix(java.lang.String fieldBracketsSuffix)
- *setFieldId*

public void setFieldId(java.lang.String fieldId)
- *setFieldLogicOperator*

public void setFieldLogicOperator(java.lang.String fieldLogicOperator)

- *setFieldNormalized*
public void setFieldNormalized(boolean fieldNormalized)
- *setFieldOperator*
public void setFieldOperator(java.lang.String fieldOperator)
- *setFieldValue*
public void setFieldValue(java.lang.Object value)
 - **Usage**
 - * Sets the LFieldValues with an unique value for unitary operators.
 - **Parameters**
 - * **value** - Object, value for the criteria.
- *setLFieldValues*
public void setLFieldValues(java.util.ArrayList lFieldValues)

10.1.2 CLASS CriteriaQueue

DECLARATION

```
public class CriteriaQueue
extends java.lang.Object
```

CONSTRUCTORS

- *CriteriaQueue*
public CriteriaQueue()
- *CriteriaQueue*
public CriteriaQueue(java.lang.String logicOp)

METHODS

- *getLogicOp*
public String getLogicOp()
- *getQueue*
public LinkedList getQueue()
- *setLogicOp*
public void setLogicOp(java.lang.String logicOp)
- *setQueue*
public void setQueue(java.util.LinkedList queue)

10.1.3 CLASS **HibernateUtil**

Implementation of an Hibernate utility class to encapsulate the Hibernate framework. It will provide access to Hibernate framework utilities and will be in charge of configuring, starting up and shutting down the Hibernate engine.
Responsibilities:

Configure and startup Hibernate framework.

Provide access to Hibernate Sessions.

Shutdown Hibernate framework.

DECLARATION

```
public class HibernateUtil
extends java.lang.Object
```

CONSTRUCTORS

- *HibernateUtil*
public **HibernateUtil**()

METHODS

- *configHibernate*
public static void **configHibernate**(java.io.InputStream **cfgFile**)
 - **Usage**
* Configuration of the session factory.
 - **Parameters**
* **cfgFile** - InputStream, stream opened to the Hibernate configuration file.

- *getRevision*
public static String **getRevision**()
 - **Usage**
* Returns tag Id assigned to CVS source file.

- *getSessionFactory*
public static SessionFactory **getSessionFactory**()
 - **Usage**
* Return the unique SessionFactory to open a new session to manage the connection to the Database through Hibernate framework.
 - **Returns** - SessionFactory, Factory for creation of Hibernate sessions.

- *shutdown*

```
public static void shutdown( )
```

– **Usage**

- * Close caches and connection pools of Hibernate framework.

10.1.4 CLASS SQLGenerator

Business Logic class that implements an engine for generating SQL statements.
Responsibilities:

- SQL statements generation.

DECLARATION

```
public class SQLGenerator
extends java.lang.Object
```

CONSTRUCTORS

- *SQLGenerator*

```
public SQLGenerator( )
```

METHODS

- *getRevision*

```
public static String getRevision( )
```

– **Usage**

- * Returns tag Id assigned to CVS source file.
-

- *getSQLCriteria*

```
public Criteria getSQLCriteria( Session session, java.lang.Class entity,
java.util.ArrayList IWhereCriteria )
```

– **Usage**

- * Build and return the where clause of the sql query made from the sql searching criteria.

– **Parameters**

- * **session** - Session, Hibernate session used to create the criteria instance.
- * **entity** - Class, Entity (Table or View) that will be searched.
- * **IWhereCriteria** - ArrayList, List with all the query criteria that will made the WHERE statement of the SQL query.

- **Returns** - Criteria, where clause of the SQL query built from the searching criteria.
-

- *getSQLCriteria*

```
public Criteria getSQLCriteria( Session session, java.lang.String
tableName, java.util.ArrayList IWhereCriteria )
```


- **Usage**

- * Build and return the where clause of the sql query made from the sql searching criteria.

- **Parameters**

- * **session** - Session, Hibernate session used to create the criteria instance.
- * **tableName** - String, Name of the queried table (or view).
- * **lWhereCriteria** - ArrayList, List with all the query criteria that will made the WHERE statement of the SQL query.

- **Returns** - Criteria, where clause of the SQL query built from the searching criteria.
-

- *getSQLQuery*

```
public String getSQLQuery( java.lang.String  tableName,
java.util.ArrayList  lFields, java.util.ArrayList  lWhereCriteria,
java.util.ArrayList  lOrderCriteria )
```

- **Usage**

- * Build and return the SQL query made from the sql query arguments.

- **Parameters**

- * **tableName** - String, Name of the queried table (or view).
- * **lFields** - ArrayList, List with all the fields the result is made of.
- * **lWhereCriteria** - ArrayList, List with all the query criteria that will made the WHERE statement of the SQL query.
- * **lOrderCriteria** - ArrayList, List with all the order by criteria that will made the ORDER BY statement of the SQL query.

- **Returns** - String, SQL query made from the arguments.

Chapter 11

Package com.imagos.axilice.client.gui

<i>Package Contents</i>	<i>Page</i>
<hr/>	
Interfaces	
AxiliceEventListener	284
<i>...no description...</i>	
GUIObject	284
<i>Public interface that defines the base event handlers for any GUI object.</i>	
Classes	
AxiliceCoreView	285
<i>Helper class that implements the Axilice core GUI.</i>	
AxiliceDateSelect	288
<i>...no description...</i>	
AxiliceDetailEntityView	288
<i>Helper class that implements the Axilice entity GUI.</i>	
AxiliceEditEntityView	293
<i>...no description...</i>	
AxiliceEntitySelectionView	294
<i>Helper class that implements the handler for sub-entities selection (magnifying glass buttons).</i>	
AxiliceEntityView	296
<i>Helper class that implements the Axilice entity GUI.</i>	
AxiliceEvent	300
<i>...no description...</i>	
AxiliceExtensionFieldsView	301
<i>...no description...</i>	
AxiliceGUIBuilder	303
<i>...no description...</i>	
AxiliceInitView	305
<i>Helper class that implements the Axilice initializing GUI.</i>	
AxiliceModuleView	306
<i>Helper class that implements the Axilice module GUI.</i>	
AxiliceValueInput	307
<i>...no description...</i>	
EntityExtensionCatalogListener	308
<i>...no description...</i>	
EntityExtensionCatalogResizeListener	308

...no description...	
EntityExtensionCatalogTextListener	309
...no description...	
GUIOperation	309
...no description...	
ImageUtils	310
...no description...	

11.1 Interfaces

11.1.1 INTERFACE AxiliceEventListener

DECLARATION

```
public interface AxiliceEventListener
implements java.util.EventListener
```

METHODS

- *axiliceViewModified*

```
public void axiliceViewModified( com.imagos.axilice.client.gui.AxiliceEvent
e )
```

11.1.2 INTERFACE GUIObject

Public interface that defines the base event handlers for any GUI object. The base event handlers are:

addAction: Event handler for insert operation (prepare the system to add a new object).

modAction: Event handler for update operation (prepare the system to update an existent object).

delAction: Event handler for delete operation (deletes the object).

showAction: Event handler for show operation (shows all the data of the object to the user).

queryAction: Event handler for search operation (looks for the object in the system).

acceptAction: Event handler for accept operation (accepts the changes made to the object).

cancelAction: Event handler for cancel operation (discards the changes made to the object).

DECLARATION

```
public interface GUIObject
```

METHODS

- *acceptAction*
public ResultMessage acceptAction()
- *addAction*
public ResultMessage addAction()
- *cancelAction*
public ResultMessage cancelAction()
- *delAction*
public ResultMessage delAction()
- *modAction*
public ResultMessage modAction()
- *queryAction*
public ResultMessage queryAction()
- *showAction*
public ResultMessage showAction()

11.2 Classes

11.2.1 CLASS AxiliceCoreView

Helper class that implements the Axilice core GUI. It completes the glade file specification to add extra features. It also implements the event handlers for the Axilice main screens and dialogs. Manages the login of the user and introduces the main window with the modules and operations available for its use.

Responsibilities:

- Definition and inicialization of the Axilice core GUI.
- Event management for the Axilice main windows.

DECLARATION

```
public class AxiliceCoreView
extends GenericHandler
```

METHODS

- *buildCoreView*
public void buildCoreView()
 - **Usage**
 - * Initializes the Axilice Core Viewer engine and builds the main window of the application. Searches automatically for the glade interface file with the application GUI configuration.

-
- *clearAxiliceAreas*
public void clearAxiliceAreas()
 - **Usage**
 - * Clears the combo of Areas. Removes all the axilice Areas from the combo.
-
- *clearAxiliceModules*
public void clearAxiliceModules()
 - **Usage**
 - * Removes all the selected area's module buttons.
-
- *getAreas*
public Hashtable getAreas()
 - **Usage**
 - * Gets the Axilice main areas' catalog (indexed by name of area). Each area contains a set of modules that implements the tools needed for the managing of that area.
 - **Returns** - Hashtable<String, AreaOT>, a catalog of areas.
-
- *getInstalledModules*
public Hashtable getInstalledModules()
 - **Usage**
 - * Gets the list of installed modules available for the user.
 - **Returns** - Hashtable<String, ModuleLauncher>, list of installed modules.
-
- *getInstance*
public static final AxiliceCoreView getInstance()
 - **Usage**
 - * Returns the unique instance of the TraceabilityTreeView class. Implements the singleton design pattern.
 - **Returns** - TraceabilityTreeView, unique instance of TraceabilityTreeView.
-
- *getRevision*
public static String getRevision()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
-
- *loadAxiliceAreas*
public void loadAxiliceAreas()
 - **Usage**
 - * (Re)Loads the combo with the available Axilice Areas.
-
- *on_areaCombo_changed*
public void on_areaCombo_changed(Event e)
 - **Usage**

- * Handler for the areas combo selection changed event. Builds the module button bar for the selected area.
- **Parameters**
 - * **e** - Event, areas combo changed event data.

- *on_buttonAxiliceAbout_clicked*
 - public void **on_buttonAxiliceAbout_clicked**(Event **e**)
 - **Usage**
 - * Handler for the About button click event. Shows the about information dialog.
 - **Parameters**
 - * **e** - Event, about button clicked event data.

- *on_buttonAxiliceExit_clicked*
 - public void **on_buttonAxiliceExit_clicked**(Event **e**)
 - **Usage**
 - * Handler for the Exit button click event. Closes the application.
 - **Parameters**
 - * **e** - Event, exit button clicked event data.

- *on_moduleButton_clicked*
 - public void **on_moduleButton_clicked**(Event **e**)
 - **Usage**
 - * Handler for the module button click event. Inits the module and shows the module main window on the screen.
 - **Parameters**
 - * **e** - Event, module button clicked event data.

- *setAreas*
 - public void **setAreas**(java.util.Hashtable **areas**)
 - **Usage**
 - * Sets up the Axilice main areas' catalog (indexed by name of area). Each area contains a set of modules that implements the tools needed for the managing of that area.
 - **Parameters**
 - * **items** - Hashtable<String, AreaOT>, a catalog of areas.

- *setInstalledModules*
 - public void **setInstalledModules**(java.util.Hashtable **modules**)
 - **Usage**
 - * Sets up the list of installed modules available for the user.
 - **Parameters**
 - * **modules** - Hashtable<String, ModuleLauncher>, list of installed modules.

11.2.2 CLASS AxiliceDateSelect

DECLARATION

```
public class AxiliceDateSelect
extends GenericHandler
```

CONSTRUCTORS

- *AxiliceDateSelect*
 public **AxiliceDateSelect**(Logger log,
 com.imagos.axilice.client.gui.common.CommonClient el, java.lang.String
 selMethod)

METHODS

- *build*
 public void **build**()
- *getDate*
 public Date **getDate**()
- *on_cancel_button_clicked*
 public void **on_cancel_button_clicked**(Event e)
- *on_ok_button_clicked*
 public void **on_ok_button_clicked**(Event e)
- *setDate*
 public void **setDate**(java.util.Date date)
- *show*
 public void **show**(java.util.Date d)

METHODS INHERITED FROM CLASS GenericHandler

11.2.3 CLASS AxiliceDetailView

Helper class that implements the Axilice entity GUI. It completes the glade file specification to add extra features. It also implements the event handlers for the Axilice entity buttons.
Responsibilities:

- Definition and inicialization of the Axilice entity GUI.
- Event management for the Axilice entity widgets.

DECLARATION

```
public class AxiliceDetailView
extends com.imagos.axilice.client.gui.AxiliceEntityView
```

CONSTRUCTORS

- *AxiliceDetailView*

```
public AxiliceDetailView( Gui parentGui, SwtWidget parentWidget,
Logger logger, com.imagos.axilice.core.common.MultiSelectionSubentityOT
subentityData, com.imagos.axilice.core.installer.EntityLauncher
entityLauncher, java.lang.String rootName )
```

- Usage

- * Subentity multiselection (Detail entity) constructor of the AxiliceDetailView class.

- Parameters

- * **parentGui** - Gui, instance of the parent GUI where the new GUI will be appended.
 - * **parentWidget** - SwtWidget, master entity widget where the detail view will be embedded.
 - * **Logger** - logger, instance of the logger used for reporting.
 - * **subentityData** - SubentityOT, data of the subentity that will be presented.
 - * **entityLauncher** - EntityLauncher, instance of the entity launcher that will handle the view.
 - * **rootName** - String, name of the root widget that will be embedded into the master entity edition panel.

METHODS

- *buildEntityView*

```
public void buildEntityView( )
```

- Usage

- * Initializes the Axilice Entity Viewer engine and builds the entity panel. Searches automatically for the glade interface file with the module GUI configuration.

- *fillEntityTable*

```
public void fillEntityTable( )
```

- Usage

- * Fills the entity tree viewer with the entity data.

- *fireExceptionAndExit*

```
public void fireExceptionAndExit( java.lang.String msg )
```

- *getEntityTree*

```
public SwtWidget getEntityTree( )
```

- *getRegistries*
public Collection getRegistries()
- *getRevision*
 public static String getRevision()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
- *getSubentityData*
public SubentityOT getSubentityData()
 - **Usage**
 - * Gets the subentity data.
 - **Returns** - SubentityOT, subentity data.
- *on_entityAddButton_clicked*
 public void on_entityAddButton_clicked(Event e)
 - **Usage**
 - * Handler for the entity add button click event. Calls the correspondent entity launcher method for adding a new entity.
 - **Parameters**
 - * e - Event, entity add button clicked event data.
- *on_entitySelectButton_clicked*
 public void on_entitySelectButton_clicked(Event e)
 - **Usage**
 - * Handler for the entity select button click event. Calls the correspondent entity launcher method for selecting a new entity.
 - **Parameters**
 - * e - Event, entity select button clicked event data.
- *setRegistries*
 public void setRegistries(java.util.Collection registries)

METHODS INHERITED FROM CLASS com.imagos.axilice.client.gui.AxiliceEntityView

(in 11.2.6, page 296)

- *addAxiliceEventListener*
public void addAxiliceEventListener(
com.imagos.axilice.client.gui.AxiliceEventListener l)
- *addAxiliceEventListenerAllViews*
public void addAxiliceEventListenerAllViews(
com.imagos.axilice.client.gui.AxiliceEventListener l)
- *buildEntityView*
 public void buildEntityView()
 - **Usage**
 - * Initializes the Axilice Entity Viewer engine and builds the entity panel. Searches automatically for the glade interface file with the module GUI configuration.

-
- *clearView*
public void clearView()
 - *distributeColumns*
public void distributeColumns(int totalWidth)
 - Usage
 - * Calculates the width of the columns distributing them equally to fit the provided container total width.
 - Parameters
 - * totalWidth - int, container total width in pixels.
-
- *fillEntityTable*
public void fillEntityTable()
 - Usage
 - * Fills the entity tree viewer with the entity data.
-
- *fillEntityTable*
public void fillEntityTable(java.util.ArrayList lRegistries)
 - Usage
 - * Fills the entity tree viewer with the entity data.
-
- *getEntityPanel*
public Composite getEntityPanel()
 - *getEntityTree*
public SwtWidget getEntityTree()
 - *getEntityTreeWidget*
public Tree getEntityTreeWidget()
 - *getRevision*
public static String getRevision()
 - Usage
 - * Returns tag Id assigned to CVS source file.
-
- *notifyEvent*
public void notifyEvent(int type)
 - *on_entity_doubleclicked*
public void on_entity_doubleclicked(Event e)
 - *on_entity_hidden*
public void on_entity_hidden(Event e)
 - *on_entity_selected*
public void on_entity_selected(Event e)
 - *on_entityAddButton_clicked*
public void on_entityAddButton_clicked(Event e)
 - Usage
 - * Handler for the entity add button click event. Calls the correspondent entity launcher method for adding a new entity.
 - Parameters
 - * e - Event, entity add button clicked event data.
-
- *on_entityDelButton_clicked*
public void on_entityDelButton_clicked(Event e)
 - Usage

- * Handler for the entity delete button click event. Calls the correspondent entity launcher method for deleting an entity.
 - **Parameters**
 - * **e** - Event, entity delete button clicked event data.

- *on_entityEditButton_clicked*
public void on_entityEditButton_clicked(Event e)
 - **Usage**
 - * Handler for the entity edit button click event. Calls the correspondent entity launcher method for editing an entity.
 - **Parameters**
 - * **e** - Event, entity edit button clicked event data.

- *on_entityGotoButton_clicked*
public void on_entityGotoButton_clicked(Event e)
 - **Usage**
 - * Handler for the entity search button click event. Calls the correspondent entity launcher method for searching for (going to) an entity.
 - **Parameters**
 - * **e** - Event, entity search button clicked event data.

- *on_entityPrintButton_clicked*
public void on_entityPrintButton_clicked(Event e)
 - **Usage**
 - * Handler for the entity print button click event. Calls the correspondent entity launcher method for printing an entity.
 - **Parameters**
 - * **e** - Event, entity print button clicked event data.

- *removeAxiliceEventListener*
public void removeAxiliceEventListener(com.imagos.axilice.client.gui.AxiliceEventListener l)

- *setColumnWidth*
public void setColumnWidth(int width)
 - **Usage**
 - * Sets the default column with of the entity table.
 - **Parameters**
 - * **width** - int, width in pixels.

- *showEntityView*
public void showEntityView()
 - **Usage**
 - * Fills the entity view with the selected data.

- *showEntityView*
public void showEntityView(Gui parentGui, com.imagos.axilice.core.installer.EntityLauncher entLauncher, Logger logger, SwtWidget parentWidget)
 - **Usage**
 - * Shows an entity in the entity view. Fills the entity view with the data of the selected entity.
 - **Parameters**
 - * **parentGui** - Gui, parent GUI where the view will be embedded.

- * **entLauncher** - EntityLauncher, instance of the entity launcher that will be shown in the entity view.
- * **logger** - Logger, instance of the logger for the entity view.
- * **parentWidget** - SwtWidget, parent widget where the contents of the view will be embedded.

- *updateView*
public void updateView()
- *verifyEntityLauncher*
public void verifyEntityLauncher()

METHODS INHERITED FROM CLASS GenericHandler

11.2.4 CLASS AxiliceEditEntityView

DECLARATION

```
public class AxiliceEditEntityView
extends GenericHandler
```

CONSTRUCTORS

- *AxiliceEditEntityView*
public AxiliceEditEntityView(
com.imagos.axilice.core.installer.EntityLauncher **entLauncher**, Logger
logger, java.lang.String **viewName**, Gui **parentGui**, java.lang.String
dialogName **)**

METHODS

- *addAxiliceEventListener*
public void addAxiliceEventListener(
com.imagos.axilice.client.gui.AxiliceEventListener **l** **)**
- *addMultiSelection*
public void addMultiSelection(
com.imagos.axilice.core.common.MultiSelectionSubentityOT **subE**,
com.imagos.axilice.client.gui.AxiliceDetailEntityView **subentView** **)**
- *buildAxiliceEditEntityView*
public void buildAxiliceEditEntityView(java.lang.String **gladepath** **)**
- *clearEditEntityView*
public void clearEditEntityView()
- *closeEditEntityView*
public void closeEditEntityView()

- *dispose*
public void **dispose**()
- *getDialog*
public SwtDialog **getDialog**()
- *getMultiSelectEntityView*
public AxiliceDetailView **getMultiSelectEntityView**(java.lang.String
subentityKey)
- *getRevision*
public static String **getRevision**()
 - Usage
 - * Returns tag Id assigned to CVS source file.
- *notifyEvent*
public void **notifyEvent**(int type)
- *on_entityEditionCancelButton_clicked*
public void **on_entityEditionCancelButton_clicked**(Event e)
- *on_entityEditionSaveButton_clicked*
public void **on_entityEditionSaveButton_clicked**(Event e)
- *on_entityEditionUndoButton_clicked*
public void **on_entityEditionUndoButton_clicked**(Event e)
- *removeAxiliceEventListener*
public void **removeAxiliceEventListener**(
com.imagos.axilice.client.gui.AxiliceEventListener l)
- *showEditEntityView*
public void **showEditEntityView**()
- *showSubentityMultiSelection*
public void **showSubentityMultiSelection**()
- *updateView*
public void **updateView**()
 - Usage
 - * Fills the widgets of the entity edition view with the contents of the selected registry of the shown entity.

METHODS INHERITED FROM CLASS `GenericHandler`

11.2.5 CLASS `AxiliceEntitySelectionView`

Helper class that implements the handler for sub-entities selection (magnifying glass buttons). Opens a new window with the list of registries for the chosen sub-entity. This window allows the user to select an existing entity registry or begin the insertion of a new one.
Responsibilities:

- Provide a selection of sub-entity event handler.
- Show the sub-entity selection dialog to the user.

DECLARATION

```
public class AxiliceEntitySelectionView
extends GenericHandler
```

FIELDS

- public static final String ERROR_ENTITY_NOHELPERCLASS
 - Constant value with the logging message for an error occurred during the loading of the entity's helper class.
- public static final String ERROR_ENTITY_NOHELPERINSTANCE
 - Constant value with the logging message for an error occurred during the instantiation of the entity's helper class.

CONSTRUCTORS

- *AxiliceEntitySelectionView*
 public **AxiliceEntitySelectionView**(java.lang.String name, Gui parentGui)

METHODS

- *buildSelectionDialog*
 public void **buildSelectionDialog**()
- *clearSelectionView*
 public void **clearSelectionView**()
- *closeSelectionView*
 public void **closeSelectionView**()
- *dispose*
 public void **dispose**()
- *getRevision*
 public static String **getRevision**()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
- *on_entitySelection_close*
 public void **on_entitySelection_close**(Event e)
 - **Usage**
 - * Handler for the entity selection dialog close event. Clears and releases the entity view component.
 - **Parameters**

* **e** - Event, close event data.

- *on_entitySelectionButton_clicked*

public void on_entitySelectionButton_clicked(Event e)

- **Usage**

- * Handler for the subentity selection buttons (magnifying glass buttons) clicked. Shows the selection dialog for selecting a valid instance of a subentity.

- **Parameters**

- * **e** - Event, magnifying glass button clicked event data.

- *on_entitySelectionCancelButton_clicked*

public void on_entitySelectionCancelButton_clicked(Event e)

- **Usage**

- * Handler for the entity selection dialog's Cancel button clicked event. Returns with no selected registry.

- **Parameters**

- * **e** - Event, cancel button clicked event data.

- *on_entitySelectionSelectButton_clicked*

public void on_entitySelectionSelectButton_clicked(Event e)

- **Usage**

- * Handler for the entity selection dialog's Select button clicked event. Returns the selected entity registry.

- **Parameters**

- * **e** - Event, select button clicked event data.

- *showSubentitySelection*

public void showSubentitySelection(com.imagos.axilice.core.installer.SubentityOT subentData)

METHODS INHERITED FROM CLASS **GenericHandler**

11.2.6 CLASS AxiliceEntityView

Helper class that implements the Axilice entity GUI. It completes the glade file specification to add extra features. It also implements the event handlers for the Axilice entity buttons.
Responsibilities:

- Definition and initialization of the Axilice entity GUI.
- Event management for the Axilice entity widgets.

DECLARATION

<pre>public class AxiliceEntityView extends GenericHandler</pre>

CONSTRUCTORS

• *AxiliceEntityView*

```
public AxiliceEntityView( java.lang.String  name )
```

– **Usage**

* Default constructor of the EntityView class.

– **Parameters**

* **name** - String, name of the interface object.

METHODS

• *addAxiliceEventListener*

```
public void addAxiliceEventListener(  
com.imagos.axilice.client.gui.AxiliceEventListener  l )
```

• *addAxiliceEventListenerAllViews*

```
public void addAxiliceEventListenerAllViews(  
com.imagos.axilice.client.gui.AxiliceEventListener  l )
```

• *buildEntityView*

```
public void buildEntityView( )
```

– **Usage**

* Initializes the Axilice Entity Viewer engine and builds the entity panel. Searches automatically for the glade interface file with the module GUI configuration.

• *clearView*

```
public void clearView( )
```

• *distributeColumns*

```
public void distributeColumns( int  totalWidth )
```

– **Usage**

* Calculates the width of the columns distributing them equally to fit the provided container total width.

– **Parameters**

* **totalWidth** - int, container total width in pixels.

• *fillEntityTable*

```
public void fillEntityTable( )
```

– **Usage**

* Fills the entity tree viewer with the entity data.

• *fillEntityTable*

```
public void fillEntityTable( java.util.ArrayList  lRegistries )
```

– **Usage**

* Fills the entity tree viewer with the entity data.

- *getEntityPanel*
public Composite **getEntityPanel**()
- *getEntityTree*
public SwtWidget **getEntityTree**()
- *getEntityTreeWidget*
public Tree **getEntityTreeWidget**()
- *getRevision*
public static String **getRevision**()
 - Usage
 - * Returns tag Id assigned to CVS source file.
- *notifyEvent*
public void **notifyEvent**(int type)
- *on_entity_doubleclicked*
public void **on_entity_doubleclicked**(Event e)
- *on_entity_hidden*
public void **on_entity_hidden**(Event e)
- *on_entity_selected*
public void **on_entity_selected**(Event e)
- *on_entityAddButton_clicked*
public void **on_entityAddButton_clicked**(Event e)
 - Usage
 - * Handler for the entity add button click event. Calls the correspondent entity launcher method for adding a new entity.
 - Parameters
 - * e - Event, entity add button clicked event data.
- *on_entityDelButton_clicked*
public void **on_entityDelButton_clicked**(Event e)
 - Usage
 - * Handler for the entity delete button click event. Calls the correspondent entity launcher method for deleting an entity.
 - Parameters
 - * e - Event, entity delete button clicked event data.
- *on_entityEditButton_clicked*
public void **on_entityEditButton_clicked**(Event e)
 - Usage
 - * Handler for the entity edit button click event. Calls the correspondent entity launcher method for editing an entity.
 - Parameters
 - * e - Event, entity edit button clicked event data.

- *on_entityGotoButton_clicked*

```
public void on_entityGotoButton_clicked( Event e )
```

- **Usage**

- * Handler for the entity search button click event. Calls the correspondent entity launcher method for searching for (going to) an entity.

- **Parameters**

- * e - Event, entity search button clicked event data.

- *on_entityPrintButton_clicked*

```
public void on_entityPrintButton_clicked( Event e )
```

- **Usage**

- * Handler for the entity print button click event. Calls the correspondent entity launcher method for printing an entity.

- **Parameters**

- * e - Event, entity print button clicked event data.

- *removeAxiliceEventListener*

```
public void removeAxiliceEventListener(  
com.imagos.axilice.client.gui.AxiliceEventListener l )
```

- *setColumnWidth*

```
public void setColumnWidth( int width )
```

- **Usage**

- * Sets the default column with of the entity table.

- **Parameters**

- * width - int, width in pixels.

- *showEntityView*

```
public void showEntityView( )
```

- **Usage**

- * Fills the entity view with the selected data.

- *showEntityView*

```
public void showEntityView( Gui parentGui,  
com.imagos.axilice.core.installer.EntityLauncher entLauncher, Logger  
logger, SwtWidget parentWidget )
```

- **Usage**

- * Shows an entity in the entity view. Fills the entity view with the data of the selected entity.

- **Parameters**

- * parentGui - Gui, parent GUI where the view will be embedded.

- * entLauncher - EntityLauncher, instance of the entity launcher that will be shown in the entity view.

- * logger - Logger, instance of the logger for the entity view.

- * parentWidget - SwtWidget, parent widget where the contents of the view will be embedded.

- *updateView*
public void updateView()
- *verifyEntityLauncher*
public void verifyEntityLauncher()

METHODS INHERITED FROM CLASS **GenericHandler**

11.2.7 CLASS **AxiliceEvent**

DECLARATION

```
public class AxiliceEvent
extends java.util.EventObject
```

SERIALIZABLE FIELDS

- private int eventType
—

FIELDS

- public static final int AXILICE_EVENT_ENTITY_MODIFIED
—
- public static final int AXILICE_EVENT_ENTITY_ADDED
—
- public static final int AXILICE_EVENT_ENTITY_DELETED
—
- public static final int AXILICE_EVENT_SUBENTITY_ADDED
—
- public static final int AXILICE_EVENT_ENTITY_PRINT_CLICKED
—
- public static final int AXILICE_EVENT_BUTTON_NEW_CLICKED
—
- public static final int AXILICE_EVENT_EDIT_UNDO
—

- public static final int AXILICE_EVENT_EDIT_CANCEL
—
- public static final int AXILICE_EVENT_ENTITY_SAVED
—

CONSTRUCTORS

- *AxiliceEvent*
public **AxiliceEvent**(java.lang.Object source, int eventType)

METHODS

- *getEventType*
public int **getEventType**()
- *setEventType*
public void **setEventType**(int eventType)

METHODS INHERITED FROM CLASS java.util.EventObject

- *getSource*
public Object **getSource**()
- *toString*
public String **toString**()

11.2.8 CLASS AxiliceExtensionFieldsView

DECLARATION

```
public class AxiliceExtensionFieldsView
extends GenericHandler
```

CONSTRUCTORS

- *AxiliceExtensionFieldsView*
public **AxiliceExtensionFieldsView**(Gui parentGui)

METHODS

- *clear*
`public void clear()`
 - **Usage**
 * Remove the contents of the table and create the columns if necessary

- *fillExtendedFieldsTree*
`public void fillExtendedFieldsTree(java.util.Map m)`
 - **Usage**
 * Fills the tree and displays it
 - **Parameters**
 * m - Map<String, ExtFieldOT>Set of values to be displayed

- *getEntityLauncher*
`public CommonClient getEntityLauncher()`

- *getExtCatalog*
`public Tree getExtCatalog()`

- *getExtensionFields*
`public ArrayList getExtensionFields()`
 - **Returns** - ArrayList<ExtFieldOT>List of fields stored on the three.

- *getExtTable*
`public ExtTableOT getExtTable()`

- *getLogger*
`public Logger getLogger()`

- *setEntityLauncher*
`public void setEntityLauncher(com.imagos.axilice.client.gui.common.CommonClient entityLauncher)`

- *setExtCatalog*
`public void setExtCatalog(Tree extCatalog)`

- *setExtTable*
`public void setExtTable(com.imagos.axilice.core.extTables.ExtTableOT extT)`

- *setLogger*
`public void setLogger(Logger logger)`

- *updateExtensionFields*
`public void updateExtensionFields()`
 - **Usage**
 * Updates the fields stored on the tree with the values displayed

METHODS INHERITED FROM CLASS GenericHandler

11.2.9 CLASS AxiliceGUIBuilder

DECLARATION

```
public class AxiliceGUIBuilder
extends java.lang.Object
```

METHODS

- *asyncExec*

```
public void asyncExec( java.lang.reflect.Method method, java.lang.Object
instance, java.lang.Object [] params )
```
 - *buildGUI*

```
public Gui buildGUI( java.lang.String guiName, java.lang.String
gladeFilename )
```
 - *buildGUI*

```
public Gui buildGUI( java.lang.String guiName, java.lang.String
gladeFilename, Gui parentGui, java.lang.String rootnode, Widget parent,
java.lang.String parentName, java.lang.String domain )
```
 - *buildGUI*

```
public Gui buildGUI( java.lang.String guiName, java.lang.String
gladeFilename, Gui parentGui, java.lang.String rootnode, Widget parent,
java.lang.String parentName, java.lang.String domain, int
backgroundColor )
```
 - *buildGUI*

```
public Gui buildGUI( java.lang.String guiName, java.lang.String
gladeFilename, Gui parentGui, SwtTopLevelWindow parentWindow )
```
 - *buildGUI*

```
public Gui buildGUI( java.lang.String guiName, java.lang.String
gladeFilename, int backgroundColor )
```
 - *connectSignal*

```
public void connectSignal( java.lang.String signalName, Widget widget,
java.lang.String handler, java.lang.Object objHandler )
```
- **Usage**
 - * Connect a signal (event) to its correspondent event handler.
 - **Parameters**
 - * **signalName** - String, SWT name of the signal (event type).
 - * **widget** - Widget, SWT widget in which the events take place.
 - * **handler** - String, handler method's name.

* objHandler - Object, instance of the class that implements the handler.

- *getColor*

public Color getColor(int color)

- *getCompletePath*

public URL getCompletePath(java.lang.String path)

- *getDisplay*

public Display getDisplay()

- *getGui*

public Gui getGui(java.lang.String guiName)

- *getGUI*

public Gui getGUI(java.lang.String guiName)

- *getImage*

public Image getImage(java.lang.String image)

- *getInputStream*

public InputStream getInputStream(java.lang.String path)

- *getInstance*

public static final AxiliceGUIBuilder getInstance()

- *getRevision*

public static String getRevision()

– **Usage**

* Returns tag Id assigned to CVS source file.

- *getToolkit*

public SwtToolkit getToolkit()

- *printGUIs*

public void printGUIs(Logger logger)

- *printGUIs*

public void printGUIs(java.io.PrintWriter out)

- *showConfirmationDialog*

public int showConfirmationDialog(Shell parentWindow, java.lang.String message, java.lang.String title, int dialogType, int buttons)

– **Usage**

* Shows a dialog with a confirmation message. It suspends the execution of the operation until the user selects an option in the confirmation window. Finally it returns the value selected by the user.

– **Parameters**

* **parentWindow** - Shell, parent window of the confirmation dialog.
 * **message** - String, message to the user.
 * **title** - String, title of the confirmation dialog window.
 * **dialogType** - int, type of the dialog icon. It may be one of the next:
 ICON_ERROR, ICON_INFORMATION, ICON_QUESTION, ICON_WARNING,
 ICON_WORKING

- * **buttons** - int, buttons showed in the dialog. Each button represents one of the options the user may choose. It may be one or more of the following: YES — NO — CANCEL, OK — CANCEL, ABORT — RETRY — IGNORE — CANCEL
- **Returns** - int, option chosen by the user. It's the pressed button ID.

- *syncExec*

```
public void syncExec( java.lang.reflect.Method  method, java.lang.Object
instance, java.lang.Object [] params )
```

11.2.10 CLASS AxiliceInitView

Helper class that implements the Axilice initializing GUI. It completes the glade file specification to add extra features. It also implements the event handlers for the Axilice initialing screens and dialogs.

Responsibilities:

- Definition and inicalization of the Axilice init GUI.
- Event management for the Axilice init windows.

DECLARATION

```
public class AxiliceInitView
extends GenericHandler
```

METHODS

- *buildInitView*

```
public void buildInitView( )
```

- **Usage**

- * Initializes the Axilice Init Viewer engine and builds the initializing window of the application. Searches automatically for the glade interface file with the application GUI configuration.

- *getInstance*

```
public static final AxiliceInitView getInstance( )
```

- **Usage**

- * Returns the unique instance of the AxiliceInitView class. Implements the singleton design pattern.

- **Returns** - AxiliceInitView, unique instance of AxiliceInitView.

- *getRevision*

```
public static String getRevision( )
```

- **Usage**

- * Returns tag Id assigned to CVS source file.

METHODS INHERITED FROM CLASS `GenericHandler`

11.2.11 CLASS `AxiliceModuleView`

Helper class that implements the Axilice module GUI. It completes the glade file specification to add extra features. It also implements the event handlers for the Axilice module buttons.
Responsibilities:

- Definition and inicialization of the Axilice module GUI.
- Event management for the Axilice module widgets.

DECLARATION

```
public class AxiliceModuleView
extends GenericHandler
```

FIELDS

- public static final String `ERROR_ENTITY_NOHELPERCLASS`
 - Constant value with the logging message for an error ocurred during the loading of the entity's helper class.
- public static final String `ERROR_ENTITY_NOHELPERINSTANCE`
 - Constant value with the logging message for an error ocurred during the instantiation of the entity's helper class.

METHODS

- *getInstance*

```
public static final AxiliceModuleView getInstance( )
```

 - **Usage**
 - * Returns the unique instance of the AxiliceModuleView class. Implements the singleton dessign pattern.
 - **Returns** - AxiliceModuleView, unique instance of AxiliceModuleView.
- *getModuleContentsPanel*

```
public SwtWidget getModuleContentsPanel( )
```
- *getRevision*

```
public static String getRevision( )
```

 - **Usage**
 - * Returns tag Id assigned to CVS source file.

- *on_entityButton_clicked*

```
public void on_entityButton_clicked( Event e )
```

– **Usage**

* Handler for the entity button click event. Inits the entity and shows the entity main window on the screen.

– **Parameters**

* **e** - Event, entity button clicked event data.

-
- *showModule*

```
public void showModule( com.imagos.axilice.core.installer.ModuleOT  
moduleData, Logger logger )
```

– **Usage**

* Build and show the module data on the screen. It builds the module panel if it's the first time the module is accessed. Fills the module panel with the available entity buttons. Show the module panel to the user.

METHODS INHERITED FROM CLASS `GenericHandler`

11.2.12 CLASS `AxiliceValueInput`

DECLARATION

```
public class AxiliceValueInput  
extends GenericHandler
```

CONSTRUCTORS

- *AxiliceValueInput*

```
public AxiliceValueInput( Logger log, Gui parentGui, java.lang.String  
selectedMethod, GenericHandler selectedListener )
```

METHODS

- *build*

```
public void build( )
```

- *getValue*

```
public String getValue( )
```

- *on_cancel_button_clicked*

```
public void on_cancel_button_clicked( Event e )
```

- *on_ok_button_clicked*

```
public void on_ok_button_clicked( Event e )
```

- *setValue*
`public void setValue(java.lang.String value)`
- *show*
`public void show(java.lang.String value)`

METHODS INHERITED FROM CLASS `GenericHandler`

11.2.13 CLASS `EntityExtensionCatalogListener`

DECLARATION

```
public class EntityExtensionCatalogListener
extends java.lang.Object
```

CONSTRUCTORS

- *EntityExtensionCatalogListener*
`public EntityExtensionCatalogListener(Tree tree)`

METHODS

- *handleEvent*
`public void handleEvent(Event event)`

11.2.14 CLASS `EntityExtensionCatalogResizeListener`

DECLARATION

```
public class EntityExtensionCatalogResizeListener
extends java.lang.Object
```

CONSTRUCTORS

- *EntityExtensionCatalogResizeListener*
`public EntityExtensionCatalogResizeListener(Composite composite, Text text, int inset)`

METHODS

- *handleEvent*
public void **handleEvent**(Event e)

11.2.15 CLASS EntityExtensionCatalogTextListener

DECLARATION

```
public class EntityExtensionCatalogTextListener
extends java.lang.Object
```

CONSTRUCTORS

- *EntityExtensionCatalogTextListener*
public **EntityExtensionCatalogTextListener**(Tree tree, TreeItem item,
Composite composite, Text text, TreeEditor editor, int inset)

METHODS

- *handleEvent*
public void **handleEvent**(Event e)

11.2.16 CLASS GUIOperation

DECLARATION

```
public class GUIOperation
extends java.lang.Object
implements java.lang.Runnable
```

CONSTRUCTORS

- *GUIOperation*
public **GUIOperation**(java.lang.reflect.Method method, java.lang.Object
instance, java.lang.Object [] params)

METHODS

- *run*
public void **run**()

11.2.17 CLASS ImageUtils

DECLARATION

```
public class ImageUtils
extends java.lang.Object
```

METHODS

- *convertImage*
public Image convertImage(Image img)
- *getImageContents*
public byte getImageContents(java.lang.String filename)
- *getInstance*
public static ImageUtils getInstance()
- *getScaledImage*
public Image getScaledImage(byte [] imgContents, int maxWidth, int maxHeight)
- *getScaledImage*
public Image getScaledImage(ImageData imgData, int maxWidth, int maxHeight)
- *getScaledImage*
public Image getScaledImage(java.io.InputStream imgIs, int maxWidth, int maxHeight)
- *getScaledImage*
public Image getScaledImage(java.lang.String imgFile, int maxWidth, int maxHeight)
- *loadImage*
public Image loadImage(byte [] imgContents)
- *loadImage*
public Image loadImage(ImageData imgData)
- *loadImage*
public Image loadImage(java.io.InputStream imgIs)
- *loadImage*
public Image loadImage(java.lang.String imgFile)
- *loadImage*
public Image loadImage(java.net.URL url)
- *resize*
public Image resize(Image image, int width, int height)
- *scale*
public Image scale(Image image, int width, int height)

Chapter 12

Package

com.imagos.axilice.server.management

Package Contents

Page

Classes

ManagementManagerBuilder	312
<i>...no description...</i>	
ManagementManagerImpl	312
<i>...no description...</i>	

12.1 Classes

12.1.1 CLASS ManagementManagerBuilder

DECLARATION

```
public class ManagementManagerBuilder
extends java.lang.Object
```

CONSTRUCTORS

- *ManagementManagerBuilder*
public **ManagementManagerBuilder**()

METHODS

- *initManagementManager*
public static void **initManagementManager**()
– **Usage**
* Initializes the common manager remote object. Makes the registration and binding of the remote ManagementManager service.

12.1.2 CLASS ManagementManagerImpl

DECLARATION

```
public class ManagementManagerImpl
extends java.rmi.server.UnicastRemoteObject
implements com.imagos.axilice.core.management.ManagementManager
```

CONSTRUCTORS

- *ManagementManagerImpl*
public **ManagementManagerImpl**()

METHODS

- *deleteMovements*
public **OperationResult deleteMovements**(
com.imagos.axilice.core.management.DocumentLineOT **entityData**)

- *getProductStock*
public ArrayList getProductStock(
com.imagos.axilice.core.warehouse.ProductOT product,
com.imagos.axilice.core.warehouse.WarehouseOT warehouse)
- *nextSerialNumber*
public long nextSerialNumber(
com.imagos.axilice.core.management.SerialNumberOT serial)

METHODS INHERITED FROM CLASS java.rmi.server.UnicastRemoteObject

- *clone*
public Object clone()
- *exportObject*
public static RemoteStub exportObject(java.rmi.Remote arg0)
- *exportObject*
public static Remote exportObject(java.rmi.Remote arg0, int arg1)
- *exportObject*
public static Remote exportObject(java.rmi.Remote arg0, int arg1,
java.rmi.server.RMIClientSocketFactory arg2, java.rmi.server.RMIServerSocketFactory
arg3)
- *unexportObject*
public static boolean unexportObject(java.rmi.Remote arg0, boolean arg1)

METHODS INHERITED FROM CLASS java.rmi.server.RemoteServer

- *getClientHost*
public static String getClientHost()
- *getLog*
public static PrintStream getLog()
- *setLog*
public static void setLog(java.io.OutputStream arg0)

METHODS INHERITED FROM CLASS java.rmi.server.RemoteObject

- *equals*
public boolean equals(java.lang.Object arg0)
- *getRef*
public RemoteRef getRef()
- *hashCode*
public int hashCode()
- *toString*
public String toString()
- *toStub*
public static Remote toStub(java.rmi.Remote arg0)

Chapter 13

Package com.imagos.axilice.server

<i>Package Contents</i>	<i>Page</i>
<hr/>	
Classes	
ServerContext	315
<i>This class lays on the server side of the Axilice application and works as an interface for initializing the server generic instances used by the server objects.</i>	
ServerManager	316
<i>Implements the main class of the server side of the application.</i>	
<hr/>	

13.1 Classes

13.1.1 CLASS ServerContext

This class lays on the server side of the Axilice application and works as an interface for initializing the server generic instances used by the server objects. It also gives access to the generic properties and systems, as for example, logging, internationalization, etc.
Responsibilities:

Initialize server logging system.

Initialize server internationalization system.

Build the catalog of messages used by the system.

Give accessors for all these general objects.

DECLARATION

```
public class ServerContext
extends java.lang.Object
```

METHODS

- *delImage*
public OperationResult delImage(java.lang.String imageFile)
- *getI18n*
public I18n getI18n()
 - **Usage**
 - * Gets the internationalization object used by the server objects for translating messages.
 - **Returns** - I18n, Axilice server internationalization instance.
- *getImage*
public ImageData getImage(java.lang.String imageFile)
- *getInstance*
public static final ServerContext getInstance()
 - **Usage**
 - * Returns the unique instance of the ServerContext class. Implements the singleton design pattern.
 - **Returns** - ServerContext, unique instance of ServerContext.
- *getLogger*
public Logger getLogger()

- **Usage**
 - * Gets the logger instance used by the server objects for system logging messages.
 - **Returns** - Logger, Axilice server logger instance.
-
- *getLogger*

```
public Logger getLogger( java.lang.String category )
```

 - **Usage**
 - * Gets the logger instance corresponding to the specified category.
 - **Returns** - Logger, category logger instance.
-
- *getMsgCatalog*

```
public ResultMessageCatalog getMsgCatalog( )
```

 - **Usage**
 - * Gets the catalog of result and error messages showed to the user as the result of the operations made by the server objects.
 - **Returns** - ResultMessageCatalog, Axilice server messages catalog.
-
- *getRevision*

```
public static String getRevision( )
```

 - **Usage**
 - * Returns tag Id assigned to CVS source file.
-
- *initServerContext*

```
public void initServerContext( )
```

 - **Usage**
 - * Inits the server context and builds all the generic objects.
-
- *loadImage*

```
public byte loadImage( java.lang.String imageFile )
```

 - **Usage**
 - * Loads an image from the server resources path.
 - **Parameters**
 - * **imageFile** - String, filename of the image file.
 - **Returns** - byte[], rendered loaded image.
-
- *saveImage*

```
public OperationResult saveImage( java.lang.String imageFile, byte [] contents )
```

13.1.2 CLASS ServerManager

Implements the main class of the server side of the application.
Responsibilities:

- Start the server of the Axilice application.

DECLARATION

```
public class ServerManager
extends java.lang.Object
```

CONSTRUCTORS

- *ServerManager*
public **ServerManager**()

METHODS

- *getRevision*
public static String **getRevision**()
 - **Usage**
 - * Devuelve el tag Id de CVS asociado al fuente del fichero.
- *main*
public static void **main**(java.lang.String [] args)
 - **Parameters**
 - * args -

Chapter 14

Package com.imagos.axilice.core

<i>Package Contents</i>	<i>Page</i>
<hr/>	
Classes	
AxiliceException	319
<i>This class encapsulates and holds all the errors that might have taken place during any of the application operations.</i>	
HibernateEntityResolver	321
<i>This class implements a SAX Entity Resolver to update the DTD used in the Hibernate XML Parsing to the local version.</i>	
<hr/>	

14.1 Classes

14.1.1 CLASS AxiliceException

This class encapsulates and holds all the errors that might have taken place during any of the application operations.
Responsibilities:

- Encapsulate and pass on the error messages to the upper layers.

DECLARATION

```
public class AxiliceException
extends java.lang.Exception
```

SERIALIZABLE FIELDS

- private ResultMessage resMsg
 - ResultMessage instance with the message build as the result of the operation that raised the exception.

CONSTRUCTORS

- *AxiliceException*
 public **AxiliceException**()
 - **Usage**
 - * Default AxiliceException constructor.
- *AxiliceException*
 public **AxiliceException**(com.imagos.axilice.core.util.ResultMessage resMsg)
 - **Usage**
 - * AxiliceException constructor with the message to show to the end user.
 - **Parameters**
 - * resMsg - ResultMessage, operation result message.

METHODS

- *getMessage*
 public String **getMessage**()
 - **Usage**
- *getResMsg*
 public ResultMessage **getResMsg**()
 - **Usage**

- * Gets the message taken as the result of the operation that raised the exception.
- **Returns** - ResultMessage, operation result message.

- *getRevision*

`public static String getRevision()`

- **Usage**

- * Returns tag Id assigned to CVS source file.
-

- *setResMsg*

`public void setResMsg(com.imagos.axilice.core.util.ResultMessage resMsg)`

- **Usage**

- * Sets up the message taken as the result of the operation that raised the exception.

- **Parameters**

- * **resMsg** - ResultMessage, operation result message.

METHODS INHERITED FROM CLASS java.lang.Exception

METHODS INHERITED FROM CLASS java.lang.Throwable

- *fillInStackTrace*
`public synchronized native Throwable fillInStackTrace()`
- *getCause*
`public Throwable getCause()`
- *getLocalizedMessage*
`public String getLocalizedMessage()`
- *getMessage*
`public String getMessage()`
- *getStackTrace*
`public StackTraceElement getStackTrace()`
- *initCause*
`public synchronized Throwable initCause(java.lang.Throwable arg0)`
- *printStackTrace*
`public void printStackTrace()`
- *printStackTrace*
`public void printStackTrace(java.io.PrintStream arg0)`
- *printStackTrace*
`public void printStackTrace(java.io.PrintWriter arg0)`
- *setStackTrace*
`public void setStackTrace(java.lang.StackTraceElement [] arg0)`
- *toString*
`public String toString()`

14.1.2 CLASS **HibernateEntityResolver**

This class implements a SAX Entity Resolver to update the DTD used in the Hibernate XML Parsing to the local version. That way the application can work even with no Internet connection and the parsing is faster.
Responsibilities:

- Translate the web URL of the DTD to a local URI.

DECLARATION

```
public class HibernateEntityResolver
extends java.lang.Object
implements org.xml.sax.EntityResolver
```

CONSTRUCTORS

- *HibernateEntityResolver*
public HibernateEntityResolver()

METHODS

- *getRevision*
public static String getRevision()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
- *resolveEntity*
public InputSource resolveEntity(java.lang.String publicId, java.lang.String systemId)
 - **Usage**
 - * Allow the application to resolve external entities. Its the implementation of the resolveEntity method of the EntityResolver interface. In this method the DTD external entity will be redirected to its local version.
 - **Parameters**
 - * **publicId** - String, The public identifier of the external entity being referenced, or null if none was supplied.
 - * **systemId** - String, The system identifier of the external entity being referenced.
 - **Returns** - InputSource An InputSource object describing the new input source, or null to request that the parser open a regular URI connection to the system identifier.

Chapter 15

Package com.imagos.axilice.client.management

<i>Package Contents</i>	<i>Page</i>
<hr/>	
Classes	
DocumentClient <i>...no description...</i>	323
DocumentLineClient <i>...no description...</i>	336
DocumentTypeClient <i>...no description...</i>	349
LotOutItemListener <i>...no description...</i>	361
LotOutListener <i>...no description...</i>	362
LotOutSelector <i>...no description...</i>	363
MovementClient <i>...no description...</i>	364
SerialNumberClient <i>...no description...</i>	377
<hr/>	

15.1 Classes

15.1.1 CLASS DocumentClient

DECLARATION

```
public class DocumentClient
extends com.imagos.axilice.client.gui.common.CommonClient
implements com.imagos.axilice.client.gui.AxiliceEventListener
```

CONSTRUCTORS

- *DocumentClient*
public **DocumentClient**()

METHODS

- *axiliceViewModified*
public void **axiliceViewModified**(com.imagos.axilice.client.gui.AxiliceEvent
e)
- *build_extension_fields_view*
public void **build_extension_fields_view**()
- *buildSelectionSets*
public void **buildSelectionSets**()
- *fillEntityData*
public void **fillEntityData**()
- *getEntityData*
public BaseVO **getEntityData**()
- *getWidgets*
public void **getWidgets**()
- *init_specific_components*
public void **init_specific_components**()
- *on_combo_country_billing_changed*
public void **on_combo_country_billing_changed**(Event e)
- *on_combo_country_shipping_changed*
public void **on_combo_country_shipping_changed**(Event e)
- *on_date_selected*
public void **on_date_selected**(Event e)
- *on_document_changed*
public void **on_document_changed**(Event e)

- *on_select_date_clicked*
public void on_select_date_clicked(Event e)
- *printEntity*
public OperationResult printEntity()
- *selectSubentity*
public void selectSubentity(java.lang.String parentId,
com.imagos.axilice.core.common.BaseVO subentityOT)
- *setEditionMode*
public void setEditionMode(boolean enabled)
- *setWidgetsValues*
public void setWidgetsValues(com.imagos.axilice.core.common.BaseVO entity
)
- *validateEntity*
public OperationResult validateEntity()

METHODS INHERITED FROM CLASS com.imagos.axilice.client.gui.common.CommonClient

(in 17.1.2, page 402)

- *addAxiliceEventListener*
public void addAxiliceEventListener(
com.imagos.axilice.client.gui.AxiliceEventListener l)
- *addEntity*
public ArrayList addEntity()
- *addExistingSubentity*
public void addExistingSubentity(com.imagos.axilice.core.installer.SubentityOT
ent)
 - **Usage**
 - * Shows a dialog for selection of a new subentity
 - **Parameters**
 - * ent - SubentityOT. Encapsulate data type of subentity
- *addSubentity*
public void addSubentity(com.imagos.axilice.core.common.BaseVO subentity,
com.imagos.axilice.core.common.MultiSelectionSubentityOT subE)
 - **Usage**
 - * Adds a existent entity to a multiselection list
 - **Parameters**
 - * subentity - Subentity to be added
 - * subE - Object that encapsulates information about the subentity
- *addTrazabilityTree*
public void addTrazabilityTree(TabFolder tabFolder)
- *build_extension_fields_view*
public abstract void build_extension_fields_view()
 - **Usage**
 - * Build the view of the extension fields.

It must implement only those clients who require it

-
- *buildEntity*
`public AxiliceEntityView buildEntity()`
 - *buildEntityDetail*
`public AxiliceEntityView buildEntityDetail(
com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData)`
 - *buildEntitySelectionView*
`public AxiliceEntitySelectionView buildEntitySelectionView()`
 - **Usage**
 - * Build view of selection entities
 - **Returns** - AxilicePartnerSelectionView. SelectionView created
-
- *buildSelectionSets*
`public abstract void buildSelectionSets()`
 - **Usage**
 - * Build 2 sets of SubentityOT: lSelectionSubentities and lMultiSelectionSubentities

These sets are used to connect the widgets with the handlers for the selection buttons and the multiselection panels

It must implement only those clients who require it
-
- *buildSubentityMultiSelection*
`public void buildSubentityMultiSelection()`
 - *clearEntityData*
`public void clearEntityData()`
 - *clearParentField*
`public OperationResult clearParentField(java.util.ArrayList selRegistries,
java.lang.String parentField)`
 - **Usage**
 - * Clears the value of the field indicated
 - **Parameters**
 - * selRegistries - ArrayList. Index list of records to process
 - * parentField - Name of the field to clear
 - **Returns** - OperationResult. Result of operation
-
- *delEntity*
`public OperationResult delEntity()`
 - *delEntity*
`public OperationResult delEntity(java.util.ArrayList selRegistries)`
 - *delEntity*
`public OperationResult delEntity(com.imagos.axilice.core.common.BaseVO b)`
 - **Usage**
 - * Delete a entity from database
 - **Parameters**
 - * b - BaseVO. Entity to delete
 - **Returns** - OperationResult. Result of the operation
-
- *delSubentities*
`public OperationResult delSubentities(java.util.ArrayList selRegistries)`
 - **Usage**

* Delete a set of subentities from record list of controller, not from database

Set the parent field of subentity to null

– **Parameters**

* **selRegistries** - List of index to delete

– **Returns** - `OperationResult`. Result of the operation

• *editEntity*

`public OperationResult editEntity()`

• *fillEntityData*

`public void fillEntityData()`

• *fillEntityExtensionTable*

`public void fillEntityExtensionTable(java.lang.Long id)`

– **Usage**

* Retrieve extension field values from the server for a given entity and displays them in the view

– **Parameters**

* **id** - Long. Key of the entity whose values are to be displayed

• *fillEntityTable*

`public void fillEntityTable()`

– **Usage**

* Load records from database and fill the entity view table and controller record list

• *fillEntityTable*

`public void fillEntityTable(java.util.ArrayList lR)`

– **Usage**

* Fill the entity view table with the records given

– **Parameters**

* **lR** - ArrayList. Records to be displayed

• *getClassEntity*

`public Class getClassEntity()`

– **Returns** - Class of the entity for which the controller is built

• *getDialogName*

`public String getDialogName()`

• *getEditGladePath*

`public String getEditGladePath()`

• *getEditionGui*

`public Gui getEditionGui()`

– **Usage**

* Returns the edition Gui instance.

– **Returns** - Gui, entity edition Gui.

• *getEditWindow*

`public SwtTopLevelWindow getEditWindow()`

• *getEntityData*

`public abstract BaseVO getEntityData()`

• *getEntityFields*

`public ArrayList getEntityFields()`

- *getLogger*
public Logger getLogger()
- *getParentEntityLauncher*
public EntityLauncher getParentEntityLauncher()
 – **Returns** - EntityLauncher of parent entity if is a embedded entity or null otherwise
- *getRegistries*
public Collection getRegistries()
 – **Returns** - lRegistries
- *getRegistries*
public List getRegistries(java.util.ArrayList lIds)
 – **Usage**
 * Get a list of entities stored by the controller
 – **Parameters**
 * lIds - Index list
 – **Returns** - List of entities
- *getView*
public AxiliceEntityView getView()
 – **Returns** - AxiliceEntityView. Main view or Detail view if is a embedded entity.
- *getWidgets*
public abstract void getWidgets()
 – **Usage**
 * Get the widgets from the glade and stored in local variables.

 Must be overwritten
- *init_specific_components*
public abstract void init_specific_components()
 – **Usage**
 * Initializes the specific components of the client
- *printEntity*
public OperationResult printEntity()
- *printEntity*
public OperationResult printEntity(java.lang.String reportPath, java.util.HashMap parameters)
- *queryEntity*
public ArrayList queryEntity()
- *queryEntity*
public ArrayList queryEntity(java.lang.String fieldId, java.lang.String value)
- *queryEntityWithFormat*
public ArrayList queryEntityWithFormat(java.lang.String fieldId, java.lang.String value)
- *queryFreeEntities*
public ArrayList queryFreeEntities(com.imagos.axilice.core.common.MultiSelectionSubentityOT mSubE)
 – **Usage**
 * Search entities not associated with a parent entity (null parent field)
 – **Parameters**

- * mSubE - SubentityOT for the parent entity that must be null
 - **Returns** - ArrayList. List of entities found

- *querySubentities*
 public ArrayList **querySubentities**(com.imagos.axilice.core.installer.SubentityOT entData)

- *refreshEntity*
 public OperationResult **refreshEntity**()

- *refreshEntity*
 public BaseVO **refreshEntity**(com.imagos.axilice.core.common.BaseVO entity)

- *refreshParentSelectedEntity*
 public OperationResult **refreshParentSelectedEntity**()
 - **Usage**
 - * Reload from database the value of the parent entity of the selected entity
 - **Returns** - OperationResult. The result of the operation.

- *removeRegistries*
 public OperationResult **removeRegistries**(java.util.ArrayList selRegistries, java.lang.String parentField)
 - **Usage**
 - * Deletes a list of the records stored by the controller. Also clean your parent fields and updates the database
 - **Parameters**
 - * selRegistries - Index list to remove
 - * parentField - Field name that links to the parent entity
 - **Returns** - OperationResult. Result of the operation

- *saveEntity*
 public OperationResult **saveEntity**()

- *saveEntity*
 public OperationResult **saveEntity**(java.util.ArrayList lIds)

- *saveEntity*
 public OperationResult **saveEntity**(com.imagos.axilice.core.common.BaseVO b)
 - **Usage**
 - * Save or update a entity to database
 - **Parameters**
 - * b - BaseVO. Entity to save or update
 - **Returns** - OperationResult. Result of the operation

- *selectEntity*
 public void **selectEntity**(java.lang.Long idEntity)

- *selectSubentity*
 public void **selectSubentity**(java.lang.String parentFieldId, com.imagos.axilice.core.common.BaseVO subentityOT)

- *selectTreeItem*
 public void **selectTreeItem**(java.lang.Long id)
 - **Usage**
 - * Select a item in the entity tree. No me gusta esto aqui
 - **Parameters**
 - * id - Long. Index of item to select

- *setDialogName*
public void setDialogName(java.lang.String dialogName)
- *setEditGladePath*
public void setEditGladePath(java.lang.String editGladePath)
- *setEditionMode*
public abstract void setEditionMode(boolean enabled)
- *setRegistries*
public void setRegistries(java.util.Collection collection)
 - **Usage**
 - * Setter for IRegistries

- *setWidgetsValues*
public abstract void setWidgetsValues(com.imagos.axilice.core.common.BaseVO entity)
 - **Usage**
 - * Set the values of the widgets on the edit dialog.
 - Must be overwritten.
 - **Parameters**
 - * **entity** - BaseVO. Entity that contains the values to be displayed

- *showErrorMessages*
public void showErrorMessages(com.imagos.axilice.core.util.OperationResult result)

- *updateEntityData*
public void updateEntityData()
- *updateRegistry*
public void updateRegistry(com.imagos.axilice.core.common.BaseVO ent)
 - **Usage**
 - * Add or update an entity (to the controller records)
 - **Parameters**
 - * **ent** - Entity to add or update

- *updateViews*
public void updateViews(boolean updateFromParent)
 - **Usage**
 - * Force the update of the views associated
 - **Parameters**
 - * **updateFromParent** - Indicates whether the view must be updated by reading the values from the parent entity. takes effect only when the entity is detail (is embedded in another)

- *validateEntity*
public abstract OperationResult validateEntity()
 - **Usage**
 - * Validate the values of the widgets in the edit dialog.
 - Must be overwritten.
 - **Returns** - OperationResult. Result of the validation.

- *viewEntity*
public OperationResult viewEntity()

METHODS INHERITED FROM CLASS com.imagos.axilice.core.installer.EntityLauncher

(in 24.2.2, page 532)

- *addEntity*
 public abstract ArrayList **addEntity**()
 – **Usage**
 * Add entity button clicked handler. Show the entity content screen ready to start inserting data for a new instance of the entity (new registry).
 – **Returns** - ArrayList<ResultMessage>, list of errors if any has taken place, null in any other case.

- *buildEditEntityGUI*
 public void **buildEditEntityGUI**()
 – **Usage**
 * Builds the edit entity dialog window from the corresponding glade interface file. Stores the entity dialog instance for posterior accesses.

- *buildEntity*
 public AxiliceEntityView **buildEntity**()
 – **Usage**
 * Builds the entity GUI and show it on screen to the user.

- *buildEntity*
 public AxiliceDetailEntityView **buildEntity**(SwtWidget **parent**,
 com.imagos.axilice.core.common.MultiSelectionSubentityOT **subentityData**,
 java.lang.String **rootName**)
 – **Usage**
 * Builds the entity GUI and show it on screen to the user. The entity GUI will be embedded into the parent composite widget.
 – **Parameters**
 * **parent** - Composite, parent holder for embedding the entity view.
 * **subentityData** - SubentityOT, data of the subentity to build if the entity launcher will be used to handle a multi-selection detail entity view.
 * **rootName** - String, name of the root widget that will be embedded into the master entity edition panel.
 – **Returns** - AxiliceDetailEntityView, the entity detail view built.

- *cancelEntityEdition*
 public void **cancelEntityEdition**()

- *clearEntityData*
 public abstract void **clearEntityData**()
 – **Usage**
 * Clears the entity content panel by resetting the content widgets.

- *delEntity*
 public abstract OperationResult **delEntity**()
 – **Usage**
 * Delete entity button clicked handler. Deletes the selected instance of the entity (selected registry) from the database.
 – **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *delEntity*

```
public abstract OperationResult delEntity( java.util.ArrayList lSelRegistries )
```

- **Usage**

- * Delete entity button clicked handler. Deletes the selected instances of the entity (selected registries) from the database. Its a multiple delete method.

- **Parameters**

- * lSelRegistries - ArrayList<Long>, list of selected instances of the entity (selected registries). The list will contain only the internal id of the selected registries.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *deselectSubentity*

```
public void deselectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

- **Usage**

- * Updates an entity instance by removing the selected value for a subentity field.

- **Parameters**

- * parentFieldId - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * subentityOT - BaseVO, subentity's deselected instance (registry) value.

- *editEntity*

```
public abstract OperationResult editEntity( )
```

- **Usage**

- * Edit entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), and allows the user to modify any content of the entity registry.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *fillEntityData*

```
public abstract void fillEntityData( )
```

- **Usage**

- * Fills the entity content panel with the data of the selected instance (registry) of the entity.

- *fireExceptionAndExit*

```
public void fireExceptionAndExit( java.lang.String msg )
```

- **Usage**

- * Fire a exception and stop the program

- **Parameters**

- * msg -

- *formatRegistries*

```
public ArrayList formatRegistries( )
```

- *formatRegistries*

```
public ArrayList formatRegistries( java.util.ArrayList lRegistries )
```

- *getDialog*

```
public SwtDialog getDialog( java.lang.String dialogName )
```

- *getEntity*

```
public EntityOT getEntity( )
```

- **Usage**

- * Gets the entity data.

- **Returns** - EntityOT, entity data.

- *getEntityData*
`public abstract BaseVO getEntityData()`
 - **Usage**
 - * Gets the content panel's data typed by the user formatted as an entity VO object.

- *getEntityEditViewName*
`public String getEntityEditViewName(java.lang.String parentGuiName)`

- *getEntityFields*
`public abstract ArrayList getEntityFields()`
 - **Usage**
 - * Gets the list of defined fields for an entity
 - **Returns** - ArrayList, list of entity fields.

- *getEntityViewName*
`public String getEntityViewName(java.lang.String parentGuiName)`

- *getLRegistries*
`public ArrayList getLRegistries()`
 - **Usage**
 - * Getter for lRegistries

- *getRegistry*
`public BaseVO getRegistry(long id)`

- *getRevision*
`public static String getRevision()`
 - **Usage**
 - * Returns tag Id assigned to CVS source file.

- *getSelectedEntity*
`public BaseVO getSelectedEntity()`
 - **Usage**
 - * Gets the selected instance (registry) of the entity.
 - **Returns** - BaseVO, selected instance (registry) of the entity.

- *getSelectedId*
`public Long getSelectedId()`
 - **Usage**
 - * Gets the internal id of the selected entity registry.
 - **Returns** - Long, internal id of the selected entity registry, null in case no entity registry has been selected yet.

- *getSwtWidget*
`public SwtWidget getSwtWidget(java.lang.String widgetName)`

- *getWidget*
`public Widget getWidget(java.lang.String widgetName)`

- *initEntityLauncher*
`public void initEntityLauncher(Logger logger, Gui parentGui, SwtWidget parentPanel)`
 - **Usage**
 - * Inits the entity launcher context and prepares the launcher to be started.
 - **Parameters**

- * **logger** - Logger, logger instance for printing logging messages.
- * **parentGuiName** - String, name of the JavaGlade instance that built the parent where the entity will be embedded.
- * **parentPanel** - Composite, parent panel where the entity contents will be embedded.

- *isDetail*

public boolean **isDetail**()

– **Usage**

- * Indicates whether is a main entity or is embedded in another

- *on_entitySelectionButton_clicked*

public void **on_entitySelectionButton_clicked**(Event e)

– **Usage**

- * Handler for the subentity selection buttons (magnifying glass buttons) clicked. Shows the selection dialog for selecting a valid instance of a subentity.

– **Parameters**

- * **e** - Event, magnifying glass button clicked event data.

- *printEntity*

public abstract **OperationResult printEntity**()

– **Usage**

- * Print entity button clicked handler. Builds the report with the data of the selected instance of the entity (selected registry), and allows the user to print the elaborated report.

– **Returns** - **OperationResult**, result of the delete operation and list of messages generated during the operation.

- *printWidgets*

public void **printWidgets**()

- *queryEntity*

public abstract **ArrayList queryEntity**()

– **Usage**

- * Search entity button clicked handler. Searches the system for the entity instances (registries) available.

– **Returns** - **ArrayList**, list of entity instances (registries) taken as the result of the query operation.

- *queryEntity*

public abstract **ArrayList queryEntity**(java.lang.String **fieldId**, java.lang.String **value**)

– **Usage**

- * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria.

– **Parameters**

- * **fieldId** - String, name of the entity field that will be queried.
- * **value** - String, value of the entity field that will be searched for.

– **Returns** - **ArrayList**, list of entity instances (registries) taken as the result of the query operation.

- *queryEntityWithFormat*

public abstract **ArrayList queryEntityWithFormat**(java.lang.String **fieldId**, java.lang.String **value**)

– **Usage**

- * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria and show them (formatted as a list) to the user in the entity's screen.
 - **Parameters**
 - * **fieldId** - String, name of the entity field that will be queried.
 - * **value** - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList<RegistryOT>, list of entity instances (registries) taken as the result of the query operation.

- *querySubentities*

```
public abstract ArrayList querySubentities(
com.imagos.axilice.core.installer.SubentityOT entData )
```

 - **Usage**
 - * Queries the selected entity for all the detail subentities associated with it.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to query for.
 - **Returns** - ArrayList, list of detail subentity registries assigned to the master detail registry.

- *saveEntity*

```
public abstract OperationResult saveEntity( )
```

 - **Usage**
 - * Save entity button clicked handler. Update the system with the data of the edited entity registry. It's called for saving new entity registries (add a new registry) or for updating the contents of existing entity registries.
 - **Returns** - OperationResult, Result of the operation and list of messages generated through the operation.

- *saveEntityRegistry*

```
public void saveEntityRegistry( )
```
- *selectEntity*

```
public abstract void selectEntity( java.lang.Long idEntity )
```

 - **Usage**
 - * Marks an instance (registry) of the entity as the selected.
 - **Parameters**
 - * **idEntity** - Long, internal id of the selected instance (registry).

- *selectSubentity*

```
public void selectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

 - **Usage**
 - * Updates an entity instance by setting up the selected value for a subentity field.
 - **Parameters**
 - * **parentFieldId** - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * **subentityOT** - BaseVO, subentity's selected instance (registry) value.

- *setEntity*

```
public void setEntity( com.imagos.axilice.core.installer.EntityOT entity )
```

 - **Usage**
 - * Sets up the entity name.
 - **Parameters**
 - * **name** - String, entity name.

- *setRegistries*

```
public void setRegistries( java.util.ArrayList IRegistries )
```

- Usage

- * Sets the list of registries that are showed in the entity view controlled by the Entity Launcher instance. It's used for detail entities where queries are made in the master entity launcher.

- Parameters

- * lRegistries - ArrayList<BaseVO>, list of shown registries of the entity.

- *showSelectionDialog*

```
public void showSelectionDialog( com.imagos.axilice.core.installer.SubentityOT entData )
```

- Usage

- * Opens the selection dialog that enables the user to select an instance of a subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.

- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT entData, long selEntity, boolean readonly )
```

- Usage

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.
 - * selEntity - long, id of the selected instance of the subentity.
 - * readonly - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT entData, java.lang.String parentGuiName, long selEntity, boolean readonly )
```

- Usage

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.
 - * parentGuiName - String,
 - * selEntity - long, id of the selected instance of the subentity.
 - * readonly - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

- *showSubentitySelectionDialog*

```
public void showSubentitySelectionDialog( com.imagos.axilice.core.installer.SubentityOT entData )
```

- Usage

- * Opens the selection dialog that enables the user to select an instance of a multiple selection detail subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.

- *undoEntityEdition*

```
public void undoEntityEdition( )
```

- *updateEntityData*
 public abstract void **updateEntityData**()
 – **Usage**
 * Updates the selected instance (registry) data with the content panel's data typed by the user.

- *viewEntity*
 public abstract **OperationResult** **viewEntity**()
 – **Usage**
 * View entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), in read-only mode.
 – **Returns** - **OperationResult**, result of the delete operation and list of messages generated during the operation.

15.1.2 CLASS DocumentLineClient

DECLARATION

```
public class DocumentLineClient
extends com.imagos.axilice.client.gui.common.CommonClient
```

CONSTRUCTORS

- *DocumentLineClient*
 public **DocumentLineClient**()

METHODS

- *build_extension_fields_view*
 public void **build_extension_fields_view**()

- *buildSelectionSets*
 public void **buildSelectionSets**()

- *clearEntityData*
 public void **clearEntityData**()

- *fillEntityData*
 public void **fillEntityData**()

- *generateMovement*
 public **MovementOT** **generateMovement**(
 com.imagos.axilice.core.management.DocumentLineOT line, java.math.BigDecimal
quantity, com.imagos.axilice.core.warehouse.LotOT lot)

- *generateMovements*
 public **OperationResult** **generateMovements**(java.util.HashMap assignments)

- *getEntityData*
public BaseVO **getEntityData**()
- *getWidgets*
public void **getWidgets**()
- *init_specific_components*
public void **init_specific_components**()
- *on_data_changed*
public void **on_data_changed**(Event e)
- *on_lot_clicked*
public void **on_lot_clicked**(Event e)
- *selectSubentity*
public void **selectSubentity**(java.lang.String parentId,
com.imagos.axilice.core.common.BaseVO subentityOT)
- *setEditionMode*
public void **setEditionMode**(boolean enabled)
- *setWidgetsValues*
public void **setWidgetsValues**(com.imagos.axilice.core.common.BaseVO entity
)
- *validateEntity*
public OperationResult **validateEntity**()

METHODS INHERITED FROM CLASS com.imagos.axilice.client.gui.common.CommonClient

(in 17.1.2, page 402)

- *addAxiliceEventListener*
public void **addAxiliceEventListener**(
com.imagos.axilice.client.gui.AxiliceEventListener l)
 - *addEntity*
public ArrayList **addEntity**()
 - *addExistingSubentity*
public void **addExistingSubentity**(com.imagos.axilice.core.installer.SubentityOT
ent)
 - **Usage**
 - * Shows a dialog for selection of a new subentity
 - **Parameters**
 - * ent - SubentityOT. Encapsulate data type of subentity
 - *addSubentity*
public void **addSubentity**(com.imagos.axilice.core.common.BaseVO subentity,
com.imagos.axilice.core.common.MultiSelectionSubentityOT subE)
 - **Usage**
 - * Adds a existent entity to a multiselection list
 - **Parameters**
 - * subentity - Subentity to be added
 - * subE - Object that encapsulates information about the subentity
-

- *addTrazabilityTree*
public void addTrazabilityTree(TabFolder tabFolder)
- *build_extension_fields_view*
public abstract void build_extension_fields_view()
 - **Usage**
 - * Build the view of the extension fields.

It must implement only those clients who require it
- *buildEntity*
public AxiliceEntityView buildEntity()
- *buildEntityDetail*
public AxiliceEntityView buildEntityDetail(com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData)
- *buildEntitySelectionView*
public AxiliceEntitySelectionView buildEntitySelectionView()
 - **Usage**
 - * Build view of selection entities
 - **Returns** - AxilicePartnerSelectionView. SelectionView created
- *buildSelectionSets*
public abstract void buildSelectionSets()
 - **Usage**
 - * Build 2 sets of SubentityOT: lSelectionSubentities and lMultiSelectionSubentities

These sets are used to connect the widgets with the handlers for the selection buttons and the multiseleccion panels

It must implement only those clients who require it
- *buildSubentityMultiSelection*
public void buildSubentityMultiSelection()
- *clearEntityData*
public void clearEntityData()
- *clearParentField*
public OperationResult clearParentField(java.util.ArrayList selRegistries, java.lang.String parentField)
 - **Usage**
 - * Clears the value of the field indicated
 - **Parameters**
 - * **selRegistries** - ArrayList. Index list of records to process
 - * **parentField** - Name of the field to clear
 - **Returns** - OperationResult. Result of operation
- *delEntity*
public OperationResult delEntity()
- *delEntity*
public OperationResult delEntity(java.util.ArrayList selRegistries)
- *delEntity*
public OperationResult delEntity(com.imagos.axilice.core.common.BaseVO b)
 - **Usage**
 - * Delete a entity from database

- **Parameters**
 - * **b** - BaseVO. Entity to delete
 - **Returns** - OperationResult. Result of the operation
-
- *delSubentities*
 public OperationResult **delSubentities**(java.util.ArrayList selRegistries)
 - **Usage**
 - * Delete a set of subentities from record list of controller, not from database
 - Set the parent field of subentity to null
 - **Parameters**
 - * **selRegistries** - List of index to delete
 - **Returns** - OperationResult. Result of the operation
-
- *editEntity*
 public OperationResult **editEntity**()
 - *fillEntityData*
 public void **fillEntityData**()
 - *fillEntityExtensionTable*
 public void **fillEntityExtensionTable**(java.lang.Long id)
 - **Usage**
 - * Retrieve extension field values from the server for a given entity and displays them in the view
 - **Parameters**
 - * **id** - Long. Key of the entity whose values are to be displayed
-
- *fillEntityTable*
 public void **fillEntityTable**()
 - **Usage**
 - * Load records from database and fill the entity view table and controller record list
-
- *fillEntityTable*
 public void **fillEntityTable**(java.util.ArrayList lR)
 - **Usage**
 - * Fill the entity view table with the records given
 - **Parameters**
 - * **lR** - ArrayList. Records to be displayed
-
- *getClassEntity*
 public Class **getClassEntity**()
 - **Returns** - Class of the entity for which the controller is built
-
- *getDialogName*
 public String **getDialogName**()
 - *getEditGladePath*
 public String **getEditGladePath**()
 - *getEditionGui*
 public Gui **getEditionGui**()
 - **Usage**
 - * Returns the edition Gui instance.
 - **Returns** - Gui, entity edition Gui.

- *getEditWindow*
public SwtTopLevelWindow **getEditWindow**()
- *getEntityData*
public abstract BaseVO **getEntityData**()
- *getEntityFields*
public ArrayList **getEntityFields**()
- *getLogger*
public Logger **getLogger**()
- *getParentEntityLauncher*
public EntityLauncher **getParentEntityLauncher**()
 - **Returns** - EntityLauncher of parent entity if is a embedded entity or null otherwise
- *getRegistries*
public Collection **getRegistries**()
 - **Returns** - IRegistries
- *getRegistries*
public List **getRegistries**(java.util.ArrayList lIds)
 - **Usage**
 - * Get a list of entities stored by the controller
 - **Parameters**
 - * lIds - Index list
 - **Returns** - List of entities
- *getView*
public AxiliceEntityView **getView**()
 - **Returns** - AxiliceEntityView. Main view or Detail view if is a embedded entity.
- *getWidgets*
public abstract void **getWidgets**()
 - **Usage**
 - * Get the widgets from the glade and stored in local variables.
 - Must be overwritten
- *init_specific_components*
public abstract void **init_specific_components**()
 - **Usage**
 - * Initializes the specific components of the client
- *printEntity*
public OperationResult **printEntity**()
- *printEntity*
public OperationResult **printEntity**(java.lang.String reportPath, java.util.HashMap parameters)
- *queryEntity*
public ArrayList **queryEntity**()
- *queryEntity*
public ArrayList **queryEntity**(java.lang.String fieldId, java.lang.String value)
- *queryEntityWithFormat*
public ArrayList **queryEntityWithFormat**(java.lang.String fieldId, java.lang.String value)

- *queryFreeEntities*

```
public ArrayList queryFreeEntities(
com.imagos.axilice.core.common.MultiSelectionSubentityOT mSubE )
```

 - **Usage**
 - * Search entities not associated with a parent entity (null parent field)
 - **Parameters**
 - * mSubE - SubentityOT for the parent entity that must be null
 - **Returns** - ArrayList. List of entities found

- *querySubentities*

```
public ArrayList querySubentities( com.imagos.axilice.core.installer.SubentityOT
entData )
```

- *refreshEntity*

```
public OperationResult refreshEntity( )
```

- *refreshEntity*

```
public BaseVO refreshEntity( com.imagos.axilice.core.common.BaseVO entity )
```

- *refreshParentSelectedEntity*

```
public OperationResult refreshParentSelectedEntity( )
```

 - **Usage**
 - * Reload from database the value of the parent entity of the selected entity
 - **Returns** - OperationResult. The result of the operation.

- *removeRegistries*

```
public OperationResult removeRegistries( java.util.ArrayList selRegistries,
java.lang.String parentField )
```

 - **Usage**
 - * Deletes a list of the records stored by the controller. Also clean your parent fields and updates the database
 - **Parameters**
 - * selRegistries - Index list to remove
 - * parentField - Field name that links to the parent entity
 - **Returns** - OperationResult. Result of the operation

- *saveEntity*

```
public OperationResult saveEntity( )
```

- *saveEntity*

```
public OperationResult saveEntity( java.util.ArrayList lIds )
```

- *saveEntity*

```
public OperationResult saveEntity( com.imagos.axilice.core.common.BaseVO b )
```

 - **Usage**
 - * Save or update a entity to database
 - **Parameters**
 - * b - BaseVO. Entity to save or update
 - **Returns** - OperationResult. Result of the operation

- *selectEntity*

```
public void selectEntity( java.lang.Long idEntity )
```

- *selectSubentity*

```
public void selectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

- *selectTreeItem*

```
public void selectTreeItem( java.lang.Long id )
```

- **Usage**
 - * Select a item in the entity tree. No me gusta esto aqui
- **Parameters**
 - * **id** - Long. Index of item to select
- *setDialogName*

```
public void setDialogName( java.lang.String dialogName )
```
- *setEditGladePath*

```
public void setEditGladePath( java.lang.String editGladePath )
```
- *setEditionMode*

```
public abstract void setEditionMode( boolean enabled )
```
- *setRegistries*

```
public void setRegistries( java.util.Collection collection )
```

 - **Usage**
 - * Setter for IRegistries
- *setWidgetsValues*

```
public abstract void setWidgetsValues( com.imagos.axilice.core.common.BaseVO entity )
```

 - **Usage**
 - * Set the values of the widgets on the edit dialog.
 - Must be overwritten.
 - **Parameters**
 - * **entity** - BaseVO. Entity that contains the values to be displayed
- *showErrorMessages*

```
public void showErrorMessages( com.imagos.axilice.core.util.OperationResult result )
```
- *updateEntityData*

```
public void updateEntityData( )
```
- *updateRegistry*

```
public void updateRegistry( com.imagos.axilice.core.common.BaseVO ent )
```

 - **Usage**
 - * Add or update an entity (to the controller records)
 - **Parameters**
 - * **ent** - Entity to add or update
- *updateViews*

```
public void updateViews( boolean updateFromParent )
```

 - **Usage**
 - * Force the update of the views associated
 - **Parameters**
 - * **updateFromParent** - Indicates whether the view must be updated by reading the values from the parent entity. takes effect only when the entity is detail (is embedded in another)
- *validateEntity*

```
public abstract OperationResult validateEntity( )
```

 - **Usage**
 - * Validate the values of the widgets in the edit dialog.
 - Must be overwritten.
 - **Returns** - OperationResult. Result of the validation.
- *viewEntity*

```
public OperationResult viewEntity( )
```

METHODS INHERITED FROM CLASS com.imagos.axilice.core.installer.EntityLauncher

(in 24.2.2, page 532)

- *addEntity*
 public abstract ArrayList **addEntity**()
 – **Usage**
 * Add entity button clicked handler. Show the entity content screen ready to start inserting data for a new instance of the entity (new registry).
 – **Returns** - ArrayList<ResultMessage>, list of errors if any has taken place, null in any other case.

- *buildEditEntityGUI*
 public void **buildEditEntityGUI**()
 – **Usage**
 * Builds the edit entity dialog window from the corresponding glade interface file. Stores the entity dialog instance for posterior accesses.

- *buildEntity*
 public AxiliceEntityView **buildEntity**()
 – **Usage**
 * Builds the entity GUI and show it on screen to the user.

- *buildEntity*
 public AxiliceDetailEntityView **buildEntity**(SwtWidget **parent**,
 com.imagos.axilice.core.common.MultiSelectionSubentityOT **subentityData**,
 java.lang.String **rootName**)
 – **Usage**
 * Builds the entity GUI and show it on screen to the user. The entity GUI will be embedded into the parent composite widget.
 – **Parameters**
 * **parent** - Composite, parent holder for embedding the entity view.
 * **subentityData** - SubentityOT, data of the subentity to build if the entity launcher will be used to handle a multi-selection detail entity view.
 * **rootName** - String, name of the root widget that will be embedded into the master entity edition panel.
 – **Returns** - AxiliceDetailEntityView, the entity detail view built.

- *cancelEntityEdition*
 public void **cancelEntityEdition**()

- *clearEntityData*
 public abstract void **clearEntityData**()
 – **Usage**
 * Clears the entity content panel by resetting the content widgets.

- *delEntity*
 public abstract OperationResult **delEntity**()
 – **Usage**
 * Delete entity button clicked handler. Deletes the selected instance of the entity (selected registry) from the database.
 – **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *delEntity*

```
public abstract OperationResult delEntity( java.util.ArrayList lSelRegistries )
```

- **Usage**

- * Delete entity button clicked handler. Deletes the selected instances of the entity (selected registries) from the database. Its a multiple delete method.

- **Parameters**

- * lSelRegistries - ArrayList<Long>, list of selected instances of the entity (selected registries). The list will contain only the internal id of the selected registries.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *deselectSubentity*

```
public void deselectSubentity( java.lang.String parentFieldId,  
com.imagos.axilice.core.common.BaseVO subentityOT )
```

- **Usage**

- * Updates an entity instance by removing the selected value for a subentity field.

- **Parameters**

- * parentFieldId - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * subentityOT - BaseVO, subentity's deselected instance (registry) value.

- *editEntity*

```
public abstract OperationResult editEntity( )
```

- **Usage**

- * Edit entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), and allows the user to modify any content of the entity registry.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *fillEntityData*

```
public abstract void fillEntityData( )
```

- **Usage**

- * Fills the entity content panel with the data of the selected instance (registry) of the entity.

- *fireExceptionAndExit*

```
public void fireExceptionAndExit( java.lang.String msg )
```

- **Usage**

- * Fire a exception and stop the program

- **Parameters**

- * msg -

- *formatRegistries*

```
public ArrayList formatRegistries( )
```

- *formatRegistries*

```
public ArrayList formatRegistries( java.util.ArrayList lRegistries )
```

- *getDialog*

```
public SwtDialog getDialog( java.lang.String dialogName )
```

- *getEntity*

```
public EntityOT getEntity( )
```

- **Usage**

- * Gets the entity data.

- **Returns** - EntityOT, entity data.

- *getEntityData*
`public abstract BaseVO getEntityData()`
 - **Usage**
 - * Gets the content panel's data typed by the user formatted as an entity VO object.

- *getEntityEditViewName*
`public String getEntityEditViewName(java.lang.String parentGuiName)`

- *getEntityFields*
`public abstract ArrayList getEntityFields()`
 - **Usage**
 - * Gets the list of defined fields for an entity
 - **Returns** - ArrayList, list of entity fields.

- *getEntityViewName*
`public String getEntityViewName(java.lang.String parentGuiName)`

- *getLRegistries*
`public ArrayList getLRegistries()`
 - **Usage**
 - * Getter for lRegistries

- *getRegistry*
`public BaseVO getRegistry(long id)`

- *getRevision*
`public static String getRevision()`
 - **Usage**
 - * Returns tag Id assigned to CVS source file.

- *getSelectedEntity*
`public BaseVO getSelectedEntity()`
 - **Usage**
 - * Gets the selected instance (registry) of the entity.
 - **Returns** - BaseVO, selected instance (registry) of the entity.

- *getSelectedId*
`public Long getSelectedId()`
 - **Usage**
 - * Gets the internal id of the selected entity registry.
 - **Returns** - Long, internal id of the selected entity registry, null in case no entity registry has been selected yet.

- *getSwtWidget*
`public SwtWidget getSwtWidget(java.lang.String widgetName)`

- *getWidget*
`public Widget getWidget(java.lang.String widgetName)`

- *initEntityLauncher*
`public void initEntityLauncher(Logger logger, Gui parentGui, SwtWidget parentPanel)`
 - **Usage**
 - * Inits the entity launcher context and prepares the launcher to be started.
 - **Parameters**

- * **logger** - Logger, logger instance for printing logging messages.
- * **parentGuiName** - String, name of the JavaGlade instance that built the parent where the entity will be embedded.
- * **parentPanel** - Composite, parent panel where the entity contents will be embedded.

- *isDetail*

public boolean **isDetail**()

– **Usage**

- * Indicates whether is a main entity or is embedded in another

- *on_entitySelectionButton_clicked*

public void **on_entitySelectionButton_clicked**(Event **e**)

– **Usage**

- * Handler for the subentity selection buttons (magnifying glass buttons) clicked. Shows the selection dialog for selecting a valid instance of a subentity.

– **Parameters**

- * **e** - Event, magnifying glass button clicked event data.

- *printEntity*

public abstract **OperationResult printEntity**()

– **Usage**

- * Print entity button clicked handler. Builds the report with the data of the selected instance of the entity (selected registry), and allows the user to print the elaborated report.

– **Returns** - **OperationResult**, result of the delete operation and list of messages generated during the operation.

- *printWidgets*

public void **printWidgets**()

- *queryEntity*

public abstract **ArrayList queryEntity**()

– **Usage**

- * Search entity button clicked handler. Searches the system for the entity instances (registries) available.

– **Returns** - **ArrayList**, list of entity instances (registries) taken as the result of the query operation.

- *queryEntity*

public abstract **ArrayList queryEntity**(java.lang.String **fieldId**, java.lang.String **value**)

– **Usage**

- * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria.

– **Parameters**

- * **fieldId** - String, name of the entity field that will be queried.
- * **value** - String, value of the entity field that will be searched for.

– **Returns** - **ArrayList**, list of entity instances (registries) taken as the result of the query operation.

- *queryEntityWithFormat*

public abstract **ArrayList queryEntityWithFormat**(java.lang.String **fieldId**, java.lang.String **value**)

– **Usage**

- * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria and show them (formatted as a list) to the user in the entity's screen.
 - **Parameters**
 - * **fieldId** - String, name of the entity field that will be queried.
 - * **value** - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList<RegistryOT>, list of entity instances (registries) taken as the result of the query operation.

- *querySubentities*

```
public abstract ArrayList querySubentities(
com.imagos.axilice.core.installer.SubentityOT entData )
```

 - **Usage**
 - * Queries the selected entity for all the detail subentities associated with it.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to query for.
 - **Returns** - ArrayList, list of detail subentity registries assigned to the master detail registry.

- *saveEntity*

```
public abstract OperationResult saveEntity( )
```

 - **Usage**
 - * Save entity button clicked handler. Update the system with the data of the edited entity registry. It's called for saving new entity registries (add a new registry) or for updating the contents of existing entity registries.
 - **Returns** - OperationResult, Result of the operation and list of messages generated through the operation.

- *saveEntityRegistry*

```
public void saveEntityRegistry( )
```
- *selectEntity*

```
public abstract void selectEntity( java.lang.Long idEntity )
```

 - **Usage**
 - * Marks an instance (registry) of the entity as the selected.
 - **Parameters**
 - * **idEntity** - Long, internal id of the selected instance (registry).

- *selectSubentity*

```
public void selectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

 - **Usage**
 - * Updates an entity instance by setting up the selected value for a subentity field.
 - **Parameters**
 - * **parentFieldId** - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * **subentityOT** - BaseVO, subentity's selected instance (registry) value.

- *setEntity*

```
public void setEntity( com.imagos.axilice.core.installer.EntityOT entity )
```

 - **Usage**
 - * Sets up the entity name.
 - **Parameters**
 - * **name** - String, entity name.

- *setRegistries*

```
public void setRegistries( java.util.ArrayList lRegistries )
```

- Usage

- * Sets the list of registries that are showed in the entity view controlled by the Entity Launcher instance. It's used for detail entities where queries are made in the master entity launcher.

- Parameters

- * lRegistries - ArrayList<BaseVO>, list of shown registries of the entity.

- *showSelectionDialog*

```
public void showSelectionDialog( com.imagos.axilice.core.installer.SubentityOT entData )
```

- Usage

- * Opens the selection dialog that enables the user to select an instance of a subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.

- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT entData, long selEntity, boolean readonly )
```

- Usage

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.
 - * selEntity - long, id of the selected instance of the subentity.
 - * readonly - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT entData, java.lang.String parentGuiName, long selEntity, boolean readonly )
```

- Usage

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.
 - * parentGuiName - String,
 - * selEntity - long, id of the selected instance of the subentity.
 - * readonly - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

- *showSubentitySelectionDialog*

```
public void showSubentitySelectionDialog( com.imagos.axilice.core.installer.SubentityOT entData )
```

- Usage

- * Opens the selection dialog that enables the user to select an instance of a multiple selection detail subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.

- *undoEntityEdition*

```
public void undoEntityEdition( )
```

- *updateEntityData*
 public abstract void **updateEntityData**()
 – **Usage**
 * Updates the selected instance (registry) data with the content panel’s data typed by the user.

- *viewEntity*
 public abstract **OperationResult** **viewEntity**()
 – **Usage**
 * View entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), in read-only mode.
 – **Returns** - **OperationResult**, result of the delete operation and list of messages generated during the operation.

15.1.3 CLASS DocumentTypeClient

DECLARATION

```
public class DocumentTypeClient
extends com.imagos.axilice.client.gui.common.CommonClient
```

CONSTRUCTORS

- *DocumentTypeClient*
 public **DocumentTypeClient**()

METHODS

- *build_extension_fields_view*
 public void **build_extension_fields_view**()

- *buildSelectionSets*
 public void **buildSelectionSets**()

- *fillEntityData*
 public void **fillEntityData**()

- *getEntityData*
 public **BaseVO** **getEntityData**()

- *getWidgets*
 public void **getWidgets**()

- *init_specific_components*
 public void **init_specific_components**()

- *setEditionMode*
 public void **setEditionMode**(boolean enabled)

- *setWidgetsValues*

```
public void setWidgetsValues( com.imagos.axilice.core.common.BaseVO entity )
```

- *validateEntity*

```
public OperationResult validateEntity( )
```

METHODS INHERITED FROM CLASS com.imagos.axilice.client.gui.common.CommonClient

(in 17.1.2, page 402)

- *addAxiliceEventListener*

```
public void addAxiliceEventListener( com.imagos.axilice.client.gui.AxiliceEventListener l )
```

- *addEntity*

```
public ArrayList addEntity( )
```

- *addExistingSubentity*

```
public void addExistingSubentity( com.imagos.axilice.core.installer.SubentityOT ent )
```

 - **Usage**
 - * Shows a dialog for selection of a new subentity
 - **Parameters**
 - * ent - SubentityOT. Encapsulate data type of subentity

- *addSubentity*

```
public void addSubentity( com.imagos.axilice.core.common.BaseVO subentity, com.imagos.axilice.core.common.MultiSelectionSubentityOT subE )
```

 - **Usage**
 - * Adds a existent entity to a multiselection list
 - **Parameters**
 - * subentity - Subentity to be added
 - * subE - Object that encapsulates information about the subentity

- *addTrazabilityTree*

```
public void addTrazabilityTree( TabFolder tabFolder )
```

- *build_extension_fields_view*

```
public abstract void build_extension_fields_view( )
```

 - **Usage**
 - * Build the view of the extension fields.

It must implement only those clients who require it

- *buildEntity*

```
public AxiliceEntityView buildEntity( )
```

- *buildEntityDetail*

```
public AxiliceEntityView buildEntityDetail( com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData )
```

- *buildEntitySelectionView*

```
public AxiliceEntitySelectionView buildEntitySelectionView( )
```

 - **Usage**
 - * Build view of selection entities
 - **Returns** - AxilicePartnerSelectionView. SelectionView created

- *buildSelectionSets*
 public abstract void **buildSelectionSets**()
 – **Usage**
 * Build 2 sets of SubentityOT: lSelectionSubentities and lMultiSelectionSubentities
 These sets are used to connect the widgets with the handlers for the selection buttons and the multiseleccion panels
 It must implement only those clients who require it

- *buildSubentityMultiSelection*
 public void **buildSubentityMultiSelection**()
- *clearEntityData*
 public void **clearEntityData**()
- *clearParentField*
 public OperationResult **clearParentField**(java.util.ArrayList selRegistries,
 java.lang.String parentField)
 – **Usage**
 * Clears the value of the field indicated
 – **Parameters**
 * selRegistries - ArrayList. Index list of records to process
 * parentField - Name of the field to clear
 – **Returns** - OperationResult. Result of operation

- *delEntity*
 public OperationResult **delEntity**()
- *delEntity*
 public OperationResult **delEntity**(java.util.ArrayList selRegistries)
- *delEntity*
 public OperationResult **delEntity**(com.imagos.axilice.core.common.BaseVO b)
 – **Usage**
 * Delete a entity from database
 – **Parameters**
 * b - BaseVO. Entity to delete
 – **Returns** - OperationResult. Result of the operation

- *delSubentities*
 public OperationResult **delSubentities**(java.util.ArrayList selRegistries)
 – **Usage**
 * Delete a set of subentities from record list of controller, not from database
 Set the parent field of subentity to null
 – **Parameters**
 * selRegistries - List of index to delete
 – **Returns** - OperationResult. Result of the operation

- *editEntity*
 public OperationResult **editEntity**()
- *fillEntityData*
 public void **fillEntityData**()
- *fillEntityExtensionTable*
 public void **fillEntityExtensionTable**(java.lang.Long id)

- **Usage**
 - * Retrieve extension field values from the server for a given entity and displays them in the view
- **Parameters**
 - * `id` - Long. Key of the entity whose values are to be displayed

- *fillEntityTable*
public void fillEntityTable()
 - **Usage**
 - * Load records from database and fill the entity view table and controller record list

- *fillEntityTable*
public void fillEntityTable(java.util.ArrayList lR)
 - **Usage**
 - * Fill the entity view table with the records given
 - **Parameters**
 - * `lR` - ArrayList. Records to be displayed

- *getClassEntity*
public Class getClassEntity()
 - **Returns** - Class of the entity for which the controller is built

- *getDialogName*
public String getDialogName()
- *getEditGladePath*
public String getEditGladePath()
- *getEditionGui*
public Gui getEditionGui()
 - **Usage**
 - * Returns the edition Gui instance.
 - **Returns** - Gui, entity edition Gui.

- *getEditWindow*
public SwtTopLevelWindow getEditWindow()
- *getEntityData*
public abstract BaseVO getEntityData()
- *getEntityFields*
public ArrayList getEntityFields()
- *getLogger*
public Logger getLogger()
- *getParentEntityLauncher*
public EntityLauncher getParentEntityLauncher()
 - **Returns** - EntityLauncher of parent entity if is a embedded entity or null otherwise

- *getRegistries*
public Collection getRegistries()
 - **Returns** - lRegistries

- *getRegistries*
public List getRegistries(java.util.ArrayList lIds)
 - **Usage**
 - * Get a list of entities stored by the controller

- **Parameters**
 - * lIds - Index list
 - **Returns** - List of entities
-
- *getView*
 public AxiliceEntityView **getView**()
 – **Returns** - AxiliceEntityView. Main view or Detail view if is a embedded entity.
-
- *getWidgets*
 public abstract void **getWidgets**()
 – **Usage**
 - * Get the widgets from the glade and stored in local variables.

Must be overwritten
-
- *init_specific_components*
 public abstract void **init_specific_components**()
 – **Usage**
 - * Initializes the specific components of the client
-
- *printEntity*
 public OperationResult **printEntity**()
 - *printEntity*
 public OperationResult **printEntity**(java.lang.String reportPath, java.util.HashMap parameters)
-
- *queryEntity*
 public ArrayList **queryEntity**()
 - *queryEntity*
 public ArrayList **queryEntity**(java.lang.String fieldId, java.lang.String value)
 - *queryEntityWithFormat*
 public ArrayList **queryEntityWithFormat**(java.lang.String fieldId, java.lang.String value)
-
- *queryFreeEntities*
 public ArrayList **queryFreeEntities**(com.imagos.axilice.core.common.MultiSelectionSubentityOT mSubE)
 – **Usage**
 - * Search entities not associated with a parent entity (null parent field)
 – **Parameters**
 - * mSubE - SubentityOT for the parent entity that must be null
 – **Returns** - ArrayList. List of entities found
-
- *querySubentities*
 public ArrayList **querySubentities**(com.imagos.axilice.core.installer.SubentityOT entData)
 - *refreshEntity*
 public OperationResult **refreshEntity**()
 - *refreshEntity*
 public BaseVO **refreshEntity**(com.imagos.axilice.core.common.BaseVO entity)
 - *refreshParentSelectedEntity*
 public OperationResult **refreshParentSelectedEntity**()
 – **Usage**
 - * Reload from database the value of the parent entity of the selected entity

- **Returns** - OperationResult. The result of the operation.
- - *removeRegistries*
 public OperationResult removeRegistries(java.util.ArrayList selRegistries,
 java.lang.String parentField)
 - **Usage**
 - * Deletes a list of the records stored by the controller. Also clean your parent fields and updates the database
 - **Parameters**
 - * selRegistries - Index list to remove
 - * parentField - Field name that links to the parent entity
 - **Returns** - OperationResult. Result of the operation
- - *saveEntity*
 public OperationResult saveEntity()
- - *saveEntity*
 public OperationResult saveEntity(java.util.ArrayList lIds)
- - *saveEntity*
 public OperationResult saveEntity(com.imagos.axilice.core.common.BaseVO b)
 - **Usage**
 - * Save or update a entity to database
 - **Parameters**
 - * b - BaseVO. Entity to save or update
 - **Returns** - OperationResult. Result of the operation
- - *selectEntity*
 public void selectEntity(java.lang.Long idEntity)
- - *selectSubentity*
 public void selectSubentity(java.lang.String parentFieldId,
 com.imagos.axilice.core.common.BaseVO subentityOT)
- - *selectTreeItem*
 public void selectTreeItem(java.lang.Long id)
 - **Usage**
 - * Select a item in the entity tree. No me gusta esto aqui
 - **Parameters**
 - * id - Long. Index of item to select
- - *setDialogName*
 public void setDialogName(java.lang.String dialogName)
- - *setEditGladePath*
 public void setEditGladePath(java.lang.String editGladePath)
- - *setEditionMode*
 public abstract void setEditionMode(boolean enabled)
- - *setRegistries*
 public void setRegistries(java.util.Collection collection)
 - **Usage**
 - * Setter for lRegistries
- - *setWidgetsValues*
 public abstract void setWidgetsValues(com.imagos.axilice.core.common.BaseVO entity)
 - **Usage**

- * Set the values of the widgets on the edit dialog.
- Must be overwritten.
- **Parameters**
 - * **entity** - BaseVO. Entity that contains the values to be displayed
- *showErrorMessages*

```
public void showErrorMessages( com.imagos.axilice.core.util.OperationResult  result )
```

- *updateEntityData*

```
public void updateEntityData( )
```

- *updateRegistry*

```
public void updateRegistry( com.imagos.axilice.core.common.BaseVO  ent )
```

 - **Usage**
 - * Add or update an entity (to the controller records)
 - **Parameters**
 - * **ent** - Entity to add or update

- *updateViews*

```
public void updateViews( boolean  updateFromParent )
```

 - **Usage**
 - * Force the update of the views associated
 - **Parameters**
 - * **updateFromParent** - Indicates whether the view must be updated by reading the values from the parent entity. takes effect only when the entity is detail (is embedded in another)

- *validateEntity*

```
public abstract OperationResult validateEntity( )
```

 - **Usage**
 - * Validate the values of the widgets in the edit dialog.
 - Must be overwritten.
 - **Returns** - OperationResult. Result of the validation.

- *viewEntity*

```
public OperationResult viewEntity( )
```

METHODS INHERITED FROM CLASS com.imagos.axilice.core.installer.EntityLauncher

(in 24.2.2, page 532)

- *addEntity*

```
public abstract ArrayList addEntity( )
```

 - **Usage**
 - * Add entity button clicked handler. Show the entity content screen ready to start inserting data for a new instance of the entity (new registry).
 - **Returns** - ArrayList<ResultMessage>, list of errors if any has taken place, null in any other case.

- *buildEditEntityGUI*

```
public void buildEditEntityGUI( )
```

 - **Usage**

- * Builds the edit entity dialog window from the corresponding glade interface file. Stores the entity dialog instance for posterior accesses.

- *buildEntity*

```
public AxiliceEntityView buildEntity( )
```

- **Usage**

- * Builds the entity GUI and show it on screen to the user.

- *buildEntity*

```
public AxiliceDetailEntityView buildEntity( SwtWidget parent,
com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData,
java.lang.String rootName )
```

- **Usage**

- * Builds the entity GUI and show it on screen to the user. The entity GUI will be embedded into the parent composite widget.

- **Parameters**

- * **parent** - Composite, parent holder for embedding the entity view.
- * **subentityData** - SubentityOT, data of the subentity to build if the entity launcher will be used to handle a multi-selection detail entity view.
- * **rootName** - String, name of the root widget that will be embedded into the master entity edition panel.

- **Returns** - AxiliceDetailEntityView, the entity detail view built.

- *cancelEntityEdition*

```
public void cancelEntityEdition( )
```

- *clearEntityData*

```
public abstract void clearEntityData( )
```

- **Usage**

- * Clears the entity content panel by resetting the content widgets.

- *delEntity*

```
public abstract OperationResult delEntity( )
```

- **Usage**

- * Delete entity button clicked handler. Deletes the selected instance of the entity (selected registry) from the database.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *delEntity*

```
public abstract OperationResult delEntity( java.util.ArrayList lSelRegistries )
```

- **Usage**

- * Delete entity button clicked handler. Deletes the selected instances of the entity (selected registries) from the database. Its a multiple delete method.

- **Parameters**

- * **lSelRegistries** - ArrayList<Long>, list of selected instances of the entity (selected registries). The list will contain only the internal id of the selected registries.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *deselectSubentity*

```
public void deselectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

- **Usage**

- * Updates an entity instance by removing the selected value for a subentity field.
- **Parameters**
 - * **parentFieldId** - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * **subentityOT** - BaseVO, subentity's deselected instance (registry) value.

- *editEntity*
 public abstract OperationResult **editEntity**()
 - **Usage**
 - * Edit entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), and allows the user to modify any content of the entity registry.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *fillEntityData*
 public abstract void **fillEntityData**()
 - **Usage**
 - * Fills the entity content panel with the data of the selected instance (registry) of the entity.

- *fireExceptionAndExit*
 public void **fireExceptionAndExit**(java.lang.String msg)
 - **Usage**
 - * Fire a exception and stop the program
 - **Parameters**
 - * **msg** -

- *formatRegistries*
 public ArrayList **formatRegistries**()

- *formatRegistries*
 public ArrayList **formatRegistries**(java.util.ArrayList lRegistries)

- *getDialog*
 public SwtDialog **getDialog**(java.lang.String dialogName)

- *getEntity*
 public EntityOT **getEntity**()
 - **Usage**
 - * Gets the entity data.
 - **Returns** - EntityOT, entity data.

- *getEntityData*
 public abstract BaseVO **getEntityData**()
 - **Usage**
 - * Gets the content panel's data typed by the user formatted as an entity VO object.

- *getEntityEditViewName*
 public String **getEntityEditViewName**(java.lang.String parentGuiName)

- *getEntityFields*
 public abstract ArrayList **getEntityFields**()
 - **Usage**
 - * Gets the list of defined fields for an entity
 - **Returns** - ArrayList, list of entity fields.

- *getEntityViewName*
public String **getEntityViewName**(java.lang.String parentGuiName)
- *getLRegistries*
public ArrayList **getLRegistries**()
 - **Usage**
 - * Getter for IRegistries
- *getRegistry*
public BaseVO **getRegistry**(long id)
- *getRevision*
public static String **getRevision**()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
- *getSelectedEntity*
public BaseVO **getSelectedEntity**()
 - **Usage**
 - * Gets the selected instance (registry) of the entity.
 - **Returns** - BaseVO, selected instance (registry) of the entity.
- *getSelectedId*
public Long **getSelectedId**()
 - **Usage**
 - * Gets the internal id of the selected entity registry.
 - **Returns** - Long, internal id of the selected entity registry, null in case no entity registry has been selected yet.
- *getSwtWidget*
public SwtWidget **getSwtWidget**(java.lang.String widgetName)
- *getWidget*
public Widget **getWidget**(java.lang.String widgetName)
- *initEntityLauncher*
public void **initEntityLauncher**(Logger logger, Gui parentGui, SwtWidget parentPanel)
 - **Usage**
 - * Inits the entity launcher context and prepares the launcher to be started.
 - **Parameters**
 - * **logger** - Logger, logger instance for printing logging messages.
 - * **parentGuiName** - String, name of the JavaGlade instance that built the parent where the entity will be embedded.
 - * **parentPanel** - Composite, parent panel where the entity contents will be embedded.
- *isDetail*
public boolean **isDetail**()
 - **Usage**
 - * Indicates whether is a main entity or is embedded in another
- *on_entitySelectionButton_clicked*
public void **on_entitySelectionButton_clicked**(Event e)
 - **Usage**
 - * Handler for the subentity selection buttons (magnifying glass buttons) clicked. Shows the selection dialog for selecting a valid instance of a subentity.

- **Parameters**
 - * *e* - Event, magnifying glass button clicked event data.

- *printEntity*

```
public abstract OperationResult printEntity( )
```

 - **Usage**
 - * Print entity button clicked handler. Builds the report with the data of the selected instance of the entity (selected registry), and allows the user to print the elaborated report.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *printWidgets*

```
public void printWidgets( )
```

- *queryEntity*

```
public abstract ArrayList queryEntity( )
```

 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) available.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.

- *queryEntity*

```
public abstract ArrayList queryEntity( java.lang.String fieldId, java.lang.String value )
```

 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria.
 - **Parameters**
 - * *fieldId* - String, name of the entity field that will be queried.
 - * *value* - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.

- *queryEntityWithFormat*

```
public abstract ArrayList queryEntityWithFormat( java.lang.String fieldId, java.lang.String value )
```

 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria and show them (formatted as a list) to the user in the entity's screen.
 - **Parameters**
 - * *fieldId* - String, name of the entity field that will be queried.
 - * *value* - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList<RegistryOT>, list of entity instances (registries) taken as the result of the query operation.

- *querySubentities*

```
public abstract ArrayList querySubentities( com.imagos.axilice.core.installer.SubentityOT entData )
```

 - **Usage**
 - * Queries the selected entity for all the detail subentities associated with it.
 - **Parameters**
 - * *entData* - SubentityOT, data of the detail subentity to query for.

- **Returns** - ArrayList, list of detail subentity registries assigned to the master detail registry.

- *saveEntity*

```
public abstract OperationResult saveEntity( )
```

 - **Usage**
 - * Save entity button clicked handler. Update the system with the data of the edited entity registry. It's called for saving new entity registries (add a new registry) or for updating the contents of existing entity registries.
 - **Returns** - OperationResult, Result of the operation and list of messages generated through the operation.

- *saveEntityRegistry*

```
public void saveEntityRegistry( )
```
- *selectEntity*

```
public abstract void selectEntity( java.lang.Long idEntity )
```

 - **Usage**
 - * Marks an instance (registry) of the entity as the selected.
 - **Parameters**
 - * idEntity - Long, internal id of the selected instance (registry).

- *selectSubentity*

```
public void selectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

 - **Usage**
 - * Updates an entity instance by setting up the selected value for a subentity field.
 - **Parameters**
 - * parentFieldId - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * subentityOT - BaseVO, subentity's selected instance (registry) value.

- *setEntity*

```
public void setEntity( com.imagos.axilice.core.installer.EntityOT entity )
```

 - **Usage**
 - * Sets up the entity name.
 - **Parameters**
 - * name - String, entity name.

- *setRegistries*

```
public void setRegistries( java.util.ArrayList lRegistries )
```

 - **Usage**
 - * Sets the list of registries that are showed in the entity view controlled by the Entity Launcher instance. It's used for detail entities where queries are made in the master entity launcher.
 - **Parameters**
 - * lRegistries - ArrayList<BaseVO>, list of shown registries of the entity.

- *showSelectionDialog*

```
public void showSelectionDialog( com.imagos.axilice.core.installer.SubentityOT
entData )
```

 - **Usage**
 - * Opens the selection dialog that enables the user to select an instance of a subentity.
 - **Parameters**
 - * entData - SubentityOT, data of the detail subentity to select.

- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT
entData, long selEntity, boolean readonly )
```

- Usage

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

- Parameters

- * **entData** - SubentityOT, data of the detail subentity to select.
 - * **selEntity** - long, id of the selected instance of the subentity.
 - * **readonly** - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT
entData, java.lang.String parentGuiName, long selEntity, boolean readonly )
```

- Usage

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

- Parameters

- * **entData** - SubentityOT, data of the detail subentity to select.
 - * **parentGuiName** - String,
 - * **selEntity** - long, id of the selected instance of the subentity.
 - * **readonly** - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

- *showSubentitySelectionDialog*

```
public void showSubentitySelectionDialog(
com.imagos.axilice.core.installer.SubentityOT entData )
```

- Usage

- * Opens the selection dialog that enables the user to select an instance of a multiple selection detail subentity.

- Parameters

- * **entData** - SubentityOT, data of the detail subentity to select.

- *undoEntityEdition*

```
public void undoEntityEdition( )
```

- *updateEntityData*

```
public abstract void updateEntityData( )
```

- Usage

- * Updates the selected instance (registry) data with the content panel's data typed by the user.

- *viewEntity*

```
public abstract OperationResult viewEntity( )
```

- Usage

- * View entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), in read-only mode.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

DECLARATION

```
public class LotOutItemListener
extends GenericHandler
```

CONSTRUCTORS

- *LotOutItemListener*
public LotOutItemListener(Text text)
- *LotOutItemListener*
public LotOutItemListener(TreeItem item, Text text)

METHODS

- *on_focusLost*
public void on_focusLost(Event event)
- *on_keyPressed*
public void on_keyPressed(Event event)
- *setItem*
public void setItem(TreeItem item)

METHODS INHERITED FROM CLASS GenericHandler

15.1.5 CLASS LotOutListener

DECLARATION

```
public class LotOutListener
extends GenericHandler
```

CONSTRUCTORS

- *LotOutListener*
public LotOutListener(Gui parentGui, Tree w_tree)

METHODS

- *handleEvent*
public void handleEvent(Event e)
- *on_value_selected*
public void on_value_selected(Event e)

METHODS INHERITED FROM CLASS GenericHandler

15.1.6 CLASS LotOutSelector

DECLARATION

```
public class LotOutSelector
extends GenericHandler
```

CONSTRUCTORS

- *LotOutSelector*

```
public LotOutSelector(
    com.imagos.axilice.client.management.DocumentLineClient entL )
```

METHODS

- *build*

```
public void build( )
```
- *getLotsAsignement*

```
public HashMap getLotsAsignement( )
```
- *getProduct*

```
public ProductOT getProduct( )
```
- *getQuantity*

```
public BigDecimal getQuantity( )
```
- *getWarehouse*

```
public WarehouseOT getWarehouse( )
```
- *getWidgets*

```
public void getWidgets( )
```
- *isLots_selected*

```
public boolean isLots_selected( )
```
- *loadStocksFromDatabase*

```
public OperationResult loadStocksFromDatabase( )
```
- *on_cancel_button_clicked*

```
public void on_cancel_button_clicked( Event e )
```
- *on_ok_button_clicked*

```
public void on_ok_button_clicked( Event e )
```
- *on_undo_button_clicked*

```
public void on_undo_button_clicked( Event e )
```

- *setLots_selected*
public void setLots_selected(boolean lotsSelected)
- *setLotsAsignement*
public void setLotsAsignement(java.util.HashMap lotsAsignement)
- *setProduct*
public void setProduct(com.imagos.axilice.core.warehouse.ProductOT product)
- *setQuantity*
public void setQuantity(java.math.BigDecimal quantity)
- *setWarehouse*
public void setWarehouse(com.imagos.axilice.core.warehouse.WarehouseOT warehouse)
- *show*
public void show()
- *updateStocksFromDatabase*
public OperationResult updateStocksFromDatabase()
- *validate*
public OperationResult validate()

METHODS INHERITED FROM CLASS GenericHandler

15.1.7 CLASS MovementClient

DECLARATION

```
public class MovementClient
extends com.imagos.axilice.client.gui.common.CommonClient
```

CONSTRUCTORS

- *MovementClient*
public MovementClient()

METHODS

- *build_extension_fields_view*
public void build_extension_fields_view()
- *buildSelectionSets*
public void buildSelectionSets()

- *clearEntityData*
public void clearEntityData()
- *fillEntityData*
public void fillEntityData()
- *getEntityData*
public BaseVO getEntityData()
- *getWidgets*
public void getWidgets()
- *init_specific_components*
public void init_specific_components()
- *on_date_selected*
public void on_date_selected(Event e)
- *on_select_date_clicked*
public void on_select_date_clicked(Event e)
- *setEditionMode*
public void setEditionMode(boolean enabled)
- *setWidgetsValues*
public void setWidgetsValues(com.imagos.axilice.core.common.BaseVO entity)
- *validateEntity*
public OperationResult validateEntity()

METHODS INHERITED FROM CLASS com.imagos.axilice.client.gui.common.CommonClient

(in 17.1.2, page 402)

- *addAxiliceEventListener*
public void addAxiliceEventListener(com.imagos.axilice.client.gui.AxiliceEventListener l)
- *addEntity*
public ArrayList addEntity()
- *addExistingSubentity*
public void addExistingSubentity(com.imagos.axilice.core.installer.SubentityOT ent)
 - **Usage**
 - * Shows a dialog for selection of a new subentity
 - **Parameters**
 - * ent - SubentityOT. Encapsulate data type of subentity
- *addSubentity*
public void addSubentity(com.imagos.axilice.core.common.BaseVO subentity, com.imagos.axilice.core.common.MultiSelectionSubentityOT subE)
 - **Usage**
 - * Adds a existent entity to a multiselection list
 - **Parameters**
 - * subentity - Subentity to be added

* **subE** - Object that encapsulates information about the subentity

- *addTrazabilityTree*

public void addTrazabilityTree(TabFolder tabFolder)

- *build_extension_fields_view*

public abstract void build_extension_fields_view()

– **Usage**

* Build the view of the extension fields.

It must implement only those clients who require it

- *buildEntity*

public AxiliceEntityView buildEntity()

- *buildEntityDetail*

**public AxiliceEntityView buildEntityDetail(
com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData)**

- *buildEntitySelectionView*

public AxiliceEntitySelectionView buildEntitySelectionView()

– **Usage**

* Build view of selection entities

– **Returns** - AxilicePartnerSelectionView. SelectionView created

- *buildSelectionSets*

public abstract void buildSelectionSets()

– **Usage**

* Build 2 sets of SubentityOT: lSelectionSubentities and lMultiSelectionSubentities

These sets are used to connect the widgets with the handlers for the selection buttons and the multiselection panels

It must implement only those clients who require it

- *buildSubentityMultiSelection*

public void buildSubentityMultiSelection()

- *clearEntityData*

public void clearEntityData()

- *clearParentField*

**public OperationResult clearParentField(java.util.ArrayList selRegistries,
java.lang.String parentField)**

– **Usage**

* Clears the value of the field indicated

– **Parameters**

* **selRegistries** - ArrayList. Index list of records to process

* **parentField** - Name of the field to clear

– **Returns** - OperationResult. Result of operation

- *delEntity*

public OperationResult delEntity()

- *delEntity*

public OperationResult delEntity(java.util.ArrayList selRegistries)

- *delEntity*

public OperationResult delEntity(com.imagos.axilice.core.common.BaseVO b)

– **Usage**

- * Delete a entity from database
 - **Parameters**
 - * **b** - BaseVO. Entity to delete
 - **Returns** - OperationResult. Result of the operation

- *delSubentities*
 public OperationResult **delSubentities**(java.util.ArrayList **selRegistries**)
 - **Usage**
 - * Delete a set of subentities from record list of controller, not from database
 - Set the parent field of subentity to null
 - **Parameters**
 - * **selRegistries** - List of index to delete
 - **Returns** - OperationResult. Result of the operation

- *editEntity*
 public OperationResult **editEntity**()

- *fillEntityData*
 public void **fillEntityData**()

- *fillEntityExtensionTable*
 public void **fillEntityExtensionTable**(java.lang.Long **id**)
 - **Usage**
 - * Retrieve extension field values from the server for a given entity and displays them in the view
 - **Parameters**
 - * **id** - Long. Key of the entity whose values are to be displayed

- *fillEntityTable*
 public void **fillEntityTable**()
 - **Usage**
 - * Load records from database and fill the entity view table and controller record list

- *fillEntityTable*
 public void **fillEntityTable**(java.util.ArrayList **lR**)
 - **Usage**
 - * Fill the entity view table with the records given
 - **Parameters**
 - * **lR** - ArrayList. Records to be displayed

- *getClassEntity*
 public Class **getClassEntity**()
 - **Returns** - Class of the entity for which the controller is built

- *getDialogName*
 public String **getDialogName**()

- *getEditGladePath*
 public String **getEditGladePath**()

- *getEditionGui*
 public Gui **getEditionGui**()
 - **Usage**
 - * Returns the edition Gui instance.

- **Returns** - Gui, entity edition Gui.
- *getEditWindow*
public SwtTopLevelWindow **getEditWindow**()
- *getEntityData*
public abstract BaseVO **getEntityData**()
- *getEntityFields*
public ArrayList **getEntityFields**()
- *getLogger*
public Logger **getLogger**()
- *getParentEntityLauncher*
public EntityLauncher **getParentEntityLauncher**()
 - **Returns** - EntityLauncher of parent entity if is a embedded entity or null otherwise
- *getRegistries*
public Collection **getRegistries**()
 - **Returns** - IRegistries
- *getRegistries*
public List **getRegistries**(java.util.ArrayList lIds)
 - **Usage**
 - * Get a list of entities stored by the controller
 - **Parameters**
 - * lIds - Index list
 - **Returns** - List of entities
- *getView*
public AxiliceEntityView **getView**()
 - **Returns** - AxiliceEntityView. Main view or Detail view if is a embedded entity.
- *getWidgets*
public abstract void **getWidgets**()
 - **Usage**
 - * Get the widgets from the glade and stored in local variables.
 - Must be overwritten
- *init_specific_components*
public abstract void **init_specific_components**()
 - **Usage**
 - * Initializes the specific components of the client
- *printEntity*
public OperationResult **printEntity**()
- *printEntity*
public OperationResult **printEntity**(java.lang.String reportPath, java.util.HashMap parameters)
- *queryEntity*
public ArrayList **queryEntity**()
- *queryEntity*
public ArrayList **queryEntity**(java.lang.String fieldId, java.lang.String value)

- *queryEntityWithFormat*
public ArrayList queryEntityWithFormat(java.lang.String fieldId, java.lang.String value)
- *queryFreeEntities*
public ArrayList queryFreeEntities(com.imagos.axilice.core.common.MultiSelectionSubentityOT mSubE)
 - **Usage**
 - * Search entities not associated with a parent entity (null parent field)
 - **Parameters**
 - * mSubE - SubentityOT for the parent entity that must be null
 - **Returns** - ArrayList. List of entities found
- *querySubentities*
public ArrayList querySubentities(com.imagos.axilice.core.installer.SubentityOT entData)
- *refreshEntity*
public OperationResult refreshEntity()
- *refreshEntity*
public BaseVO refreshEntity(com.imagos.axilice.core.common.BaseVO entity)
- *refreshParentSelectedEntity*
public OperationResult refreshParentSelectedEntity()
 - **Usage**
 - * Reload from database the value of the parent entity of the selected entity
 - **Returns** - OperationResult. The result of the operation.
- *removeRegistries*
public OperationResult removeRegistries(java.util.ArrayList selRegistries, java.lang.String parentField)
 - **Usage**
 - * Deletes a list of the records stored by the controller. Also clean your parent fields and updates the database
 - **Parameters**
 - * selRegistries - Index list to remove
 - * parentField - Field name that links to the parent entity
 - **Returns** - OperationResult. Result of the operation
- *saveEntity*
public OperationResult saveEntity()
- *saveEntity*
public OperationResult saveEntity(java.util.ArrayList lIds)
- *saveEntity*
public OperationResult saveEntity(com.imagos.axilice.core.common.BaseVO b)
 - **Usage**
 - * Save or update a entity to database
 - **Parameters**
 - * b - BaseVO. Entity to save or update
 - **Returns** - OperationResult. Result of the operation
- *selectEntity*
public void selectEntity(java.lang.Long idEntity)

- *selectSubentity*
 public void **selectSubentity**(java.lang.String parentFieldId,
 com.imagos.axilice.core.common.BaseVO subentityOT)
- *selectTreeItem*
 public void **selectTreeItem**(java.lang.Long id)
 - **Usage**
 - * Select a item in the entity tree. No me gusta esto aqui
 - **Parameters**
 - * id - Long. Index of item to select
- *setDialogName*
 public void **setDialogName**(java.lang.String dialogName)
- *setEditGladePath*
 public void **setEditGladePath**(java.lang.String editGladePath)
- *setEditionMode*
 public abstract void **setEditionMode**(boolean enabled)
- *setRegistries*
 public void **setRegistries**(java.util.Collection collection)
 - **Usage**
 - * Setter for IRegistries
- *setWidgetsValues*
 public abstract void **setWidgetsValues**(com.imagos.axilice.core.common.BaseVO
 entity)
 - **Usage**
 - * Set the values of the widgets on the edit dialog.
 - Must be overwritten.
 - **Parameters**
 - * entity - BaseVO. Entity that contains the values to be displayed
- *showErrorMessages*
 public void **showErrorMessages**(com.imagos.axilice.core.util.OperationResult result)
- *updateEntityData*
 public void **updateEntityData**()
- *updateRegistry*
 public void **updateRegistry**(com.imagos.axilice.core.common.BaseVO ent)
 - **Usage**
 - * Add or update an entity (to the controller records)
 - **Parameters**
 - * ent - Entity to add or update
- *updateViews*
 public void **updateViews**(boolean updateFromParent)
 - **Usage**
 - * Force the update of the views associated
 - **Parameters**
 - * updateFromParent - Indicates whether the view must be updated by reading the values from the parent entity. takes effect only when the entity is detail (is embedded in another)

- *validateEntity*
 public abstract **OperationResult validateEntity()**
 - **Usage**
 - * Validate the values of the widgets in the edit dialog.
 - **Must be overwritten.**
 - **Returns** - **OperationResult**. Result of the validation.
-
- *viewEntity*
 public **OperationResult viewEntity()**

METHODS INHERITED FROM CLASS com.imagos.axilice.core.installer.EntityLauncher

(in 24.2.2, page 532)

- *addEntity*
 public abstract **ArrayList addEntity()**
 - **Usage**
 - * Add entity button clicked handler. Show the entity content screen ready to start inserting data for a new instance of the entity (new registry).
 - **Returns** - **ArrayList<ResultMessage>**, list of errors if any has taken place, null in any other case.
-
- *buildEditEntityGUI*
 public void **buildEditEntityGUI()**
 - **Usage**
 - * Builds the edit entity dialog window from the corresponding glade interface file. Stores the entity dialog instance for posterior accesses.
-
- *buildEntity*
 public **AxiliceEntityView buildEntity()**
 - **Usage**
 - * Builds the entity GUI and show it on screen to the user.
-
- *buildEntity*
 public **AxiliceDetailEntityView buildEntity(SwtWidget parent, com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData, java.lang.String rootName)**
 - **Usage**
 - * Builds the entity GUI and show it on screen to the user. The entity GUI will be embedded into the parent composite widget.
 - **Parameters**
 - * **parent** - Composite, parent holder for embedding the entity view.
 - * **subentityData** - SubentityOT, data of the subentity to build if the entity launcher will be used to handle a multi-selection detail entity view.
 - * **rootName** - String, name of the root widget that will be embedded into the master entity edition panel.
 - **Returns** - **AxiliceDetailEntityView**, the entity detail view built.
-
- *cancelEntityEdition*
 public void **cancelEntityEdition()**
 - *clearEntityData*
 public abstract void **clearEntityData()**

- **Usage**
 - * Clears the entity content panel by resetting the content widgets.

- *delEntity*

```
public abstract OperationResult delEntity( )
```

 - **Usage**
 - * Delete entity button clicked handler. Deletes the selected instance of the entity (selected registry) from the database.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *delEntity*

```
public abstract OperationResult delEntity( java.util.ArrayList lSelRegistries )
```

 - **Usage**
 - * Delete entity button clicked handler. Deletes the selected instances of the entity (selected registries) from the database. Its a multiple delete method.
 - **Parameters**
 - * lSelRegistries - ArrayList<Long>, list of selected instances of the entity (selected registries). The list will contain only the internal id of the selected registries.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *deselectSubentity*

```
public void deselectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

 - **Usage**
 - * Updates an entity instance by removing the selected value for a subentity field.
 - **Parameters**
 - * parentFieldId - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * subentityOT - BaseVO, subentity's deselected instance (registry) value.

- *editEntity*

```
public abstract OperationResult editEntity( )
```

 - **Usage**
 - * Edit entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), and allows the user to modify any content of the entity registry.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *fillEntityData*

```
public abstract void fillEntityData( )
```

 - **Usage**
 - * Fills the entity content panel with the data of the selected instance (registry) of the entity.

- *fireExceptionAndExit*

```
public void fireExceptionAndExit( java.lang.String msg )
```

 - **Usage**
 - * Fire a exception and stop the program
 - **Parameters**
 - * msg -

- *formatRegistries*
public ArrayList formatRegistries()
- *formatRegistries*
public ArrayList formatRegistries(java.util.ArrayList lRegistries)
- *getDialog*
public SwtDialog getDialog(java.lang.String dialogName)
- *getEntity*
public EntityOT getEntity()
 - **Usage**
 - * Gets the entity data.
 - **Returns** - EntityOT, entity data.

- *getEntityData*
public abstract BaseVO getEntityData()
 - **Usage**
 - * Gets the content panel's data typed by the user formatted as an entity VO object.

- *getEntityEditViewName*
public String getEntityEditViewName(java.lang.String parentGuiName)
- *getEntityFields*
public abstract ArrayList getEntityFields()
 - **Usage**
 - * Gets the list of defined fields for an entity
 - **Returns** - ArrayList, list of entity fields.

- *getEntityViewName*
public String getEntityViewName(java.lang.String parentGuiName)
- *getLRegistries*
public ArrayList getLRegistries()
 - **Usage**
 - * Getter for lRegistries

- *getRegistry*
public BaseVO getRegistry(long id)
- *getRevision*
public static String getRevision()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.

- *getSelectedEntity*
public BaseVO getSelectedEntity()
 - **Usage**
 - * Gets the selected instance (registry) of the entity.
 - **Returns** - BaseVO, selected instance (registry) of the entity.

- *getSelectedId*
public Long getSelectedId()
 - **Usage**
 - * Gets the internal id of the selected entity registry.
 - **Returns** - Long, internal id of the selected entity registry, null in case no entity registry has been selected yet.

- *getSwtWidget*
public SwtWidget getSwtWidget(java.lang.String widgetName)
- *getWidget*
public Widget getWidget(java.lang.String widgetName)
- *initEntityLauncher*
public void initEntityLauncher(Logger logger, Gui parentGui, SwtWidget parentPanel)
 - **Usage**
 - * Initiates the entity launcher context and prepares the launcher to be started.
 - **Parameters**
 - * **logger** - Logger, logger instance for printing logging messages.
 - * **parentGuiName** - String, name of the JavaGlade instance that built the parent where the entity will be embedded.
 - * **parentPanel** - Composite, parent panel where the entity contents will be embedded.
- *isDetail*
public boolean isDetail()
 - **Usage**
 - * Indicates whether is a main entity or is embedded in another
- *on_entitySelectionButton_clicked*
public void on_entitySelectionButton_clicked(Event e)
 - **Usage**
 - * Handler for the subentity selection buttons (magnifying glass buttons) clicked. Shows the selection dialog for selecting a valid instance of a subentity.
 - **Parameters**
 - * **e** - Event, magnifying glass button clicked event data.
- *printEntity*
public abstract OperationResult printEntity()
 - **Usage**
 - * Print entity button clicked handler. Builds the report with the data of the selected instance of the entity (selected registry), and allows the user to print the elaborated report.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.
- *printWidgets*
public void printWidgets()
- *queryEntity*
public abstract ArrayList queryEntity()
 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) available.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.
- *queryEntity*
public abstract ArrayList queryEntity(java.lang.String fieldId, java.lang.String value)
 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria.
 - **Parameters**

- * **fieldId** - String, name of the entity field that will be queried.
 - * **value** - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.

- *queryEntityWithFormat*

```
public abstract ArrayList queryEntityWithFormat( java.lang.String fieldId,
java.lang.String value )
```

 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria and show them (formatted as a list) to the user in the entity's screen.
 - **Parameters**
 - * **fieldId** - String, name of the entity field that will be queried.
 - * **value** - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList<RegistryOT>, list of entity instances (registries) taken as the result of the query operation.

- *querySubentities*

```
public abstract ArrayList querySubentities(
com.imagos.axilice.core.installer.SubentityOT entData )
```

 - **Usage**
 - * Queries the selected entity for all the detail subentities associated with it.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to query for.
 - **Returns** - ArrayList, list of detail subentity registries assigned to the master detail registry.

- *saveEntity*

```
public abstract OperationResult saveEntity( )
```

 - **Usage**
 - * Save entity button clicked handler. Update the system with the data of the edited entity registry. It's called for saving new entity registries (add a new registry) or for updating the contents of existing entity registries.
 - **Returns** - OperationResult, Result of the operation and list of messages generated through the operation.

- *saveEntityRegistry*

```
public void saveEntityRegistry( )
```

- *selectEntity*

```
public abstract void selectEntity( java.lang.Long idEntity )
```

 - **Usage**
 - * Marks an instance (registry) of the entity as the selected.
 - **Parameters**
 - * **idEntity** - Long, internal id of the selected instance (registry).

- *selectSubentity*

```
public void selectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

 - **Usage**
 - * Updates an entity instance by setting up the selected value for a subentity field.
 - **Parameters**
 - * **parentFieldId** - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.

-
- * subentityOT - BaseVO, subentity's selected instance (registry) value.
-
- *setEntity*
 public void **setEntity**(com.imagos.axilice.core.installer.EntityOT entity)
 – **Usage**
 * Sets up the entity name.
 – **Parameters**
 * name - String, entity name.
-
- *setRegistries*
 public void **setRegistries**(java.util.ArrayList lRegistries)
 – **Usage**
 * Sets the list of registries that are showed in the entity view controlled by the Entity Launcher instance. It's used for detail entities where queries are made in the master entity launcher.
 – **Parameters**
 * lRegistries - ArrayList<BaseVO>, list of shown registries of the entity.
-
- *showSelectionDialog*
 public void **showSelectionDialog**(com.imagos.axilice.core.installer.SubentityOT entData)
 – **Usage**
 * Opens the selection dialog that enables the user to select an instance of a subentity.
 – **Parameters**
 * entData - SubentityOT, data of the detail subentity to select.
-
- *showSubentityDialog*
 public void **showSubentityDialog**(com.imagos.axilice.core.installer.SubentityOT entData, long selEntity, boolean readonly)
 – **Usage**
 * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.
 – **Parameters**
 * entData - SubentityOT, data of the detail subentity to select.
 * selEntity - long, id of the selected instance of the subentity.
 * readonly - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.
-
- *showSubentityDialog*
 public void **showSubentityDialog**(com.imagos.axilice.core.installer.SubentityOT entData, java.lang.String parentGuiName, long selEntity, boolean readonly)
 – **Usage**
 * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.
 – **Parameters**
 * entData - SubentityOT, data of the detail subentity to select.
 * parentGuiName - String,
 * selEntity - long, id of the selected instance of the subentity.
 * readonly - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.
-
- *showSubentitySelectionDialog*
 public void **showSubentitySelectionDialog**(
 com.imagos.axilice.core.installer.SubentityOT entData)

- **Usage**
 - * Opens the selection dialog that enables the user to select an instance of a multiple selection detail subentity.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to select.
-
- *undoEntityEdition*
public void undoEntityEdition()
 - *updateEntityData*
public abstract void updateEntityData()
- **Usage**
 - * Updates the selected instance (registry) data with the content panel's data typed by the user.
-
- *viewEntity*
public abstract OperationResult viewEntity()
- **Usage**
 - * View entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), in read-only mode.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

15.1.8 CLASS SerialNumberClient

DECLARATION

```
public class SerialNumberClient
extends com.imagos.axilice.client.gui.common.CommonClient
```

CONSTRUCTORS

- *SerialNumberClient*
public SerialNumberClient()

METHODS

- *build_extension_fields_view*
public void build_extension_fields_view()
- *buildSelectionSets*
public void buildSelectionSets()
- *fillEntityData*
public void fillEntityData()
- *getEntityData*
public BaseVO getEntityData()

- *getWidgets*
public void **getWidgets**()
- *init_specific_components*
public void **init_specific_components**()
- *setEditionMode*
public void **setEditionMode**(boolean **enabled**)
- *setWidgetsValues*
public void **setWidgetsValues**(com.imagos.axilice.core.common.BaseVO **entity**)
- *validateEntity*
public **OperationResult** **validateEntity**()

METHODS INHERITED FROM CLASS com.imagos.axilice.client.gui.common.CommonClient

(in 17.1.2, page 402)

- *addAxiliceEventListener*
public void **addAxiliceEventListener**(
com.imagos.axilice.client.gui.AxiliceEventListener **l**)
- *addEntity*
public **ArrayList** **addEntity**()
- *addExistingSubentity*
public void **addExistingSubentity**(com.imagos.axilice.core.installer.SubentityOT
ent)
 - **Usage**
 - * Shows a dialog for selection of a new subentity
 - **Parameters**
 - * **ent** - SubentityOT. Encapsulate data type of subentity
- *addSubentity*
public void **addSubentity**(com.imagos.axilice.core.common.BaseVO **subentity**,
com.imagos.axilice.core.common.MultiSelectionSubentityOT **subE**)
 - **Usage**
 - * Adds a existent entity to a multiselection list
 - **Parameters**
 - * **subentity** - Subentity to be added
 - * **subE** - Object that encapsulates information about the subentity
- *addTrazabilityTree*
public void **addTrazabilityTree**(**TabFolder** **tabFolder**)
- *build_extension_fields_view*
public abstract void **build_extension_fields_view**()
 - **Usage**
 - * Build the view of the extension fields.

It must implement only those clients who require it
- *buildEntity*
public **AxiliceEntityView** **buildEntity**()

- *buildEntityDetail*
public AxiliceEntityView buildEntityDetail(
com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData)
- *buildEntitySelectionView*
public AxiliceEntitySelectionView buildEntitySelectionView()
 - **Usage**
 - * Build view of selection entities
 - **Returns** - AxilicePartnerSelectionView. SelectionView created

- *buildSelectionSets*
public abstract void buildSelectionSets()
 - **Usage**
 - * Build 2 sets of SubentityOT: lSelectionSubentities and lMultiSelectionSubentities

These sets are used to connect the widgets with the handlers for the selection buttons and the multiseleccion panels

It must implement only those clients who require it

- *buildSubentityMultiSelection*
public void buildSubentityMultiSelection()
- *clearEntityData*
public void clearEntityData()
- *clearParentField*
public OperationResult clearParentField(java.util.ArrayList selRegistries,
java.lang.String parentField)
 - **Usage**
 - * Clears the value of the field indicated
 - **Parameters**
 - * selRegistries - ArrayList. Index list of records to process
 - * parentField - Name of the field to clear
 - **Returns** - OperationResult. Result of operation

- *delEntity*
public OperationResult delEntity()
- *delEntity*
public OperationResult delEntity(java.util.ArrayList selRegistries)
- *delEntity*
public OperationResult delEntity(com.imagos.axilice.core.common.BaseVO b)
 - **Usage**
 - * Delete a entity from database
 - **Parameters**
 - * b - BaseVO. Entity to delete
 - **Returns** - OperationResult. Result of the operation

- *delSubentities*
public OperationResult delSubentities(java.util.ArrayList selRegistries)
 - **Usage**
 - * Delete a set of subentities from record list of controller, not from database

Set the parent field of subentity to null

- **Parameters**
 - * selRegistries - List of index to delete
 - **Returns** - OperationResult. Result of the operation

- *editEntity*
public OperationResult editEntity()
- *fillEntityData*
public void fillEntityData()
- *fillEntityExtensionTable*
public void fillEntityExtensionTable(java.lang.Long id)
 - **Usage**
 - * Retrieve extension field values from the server for a given entity and displays them in the view
 - **Parameters**
 - * id - Long. Key of the entity whose values are to be displayed

- *fillEntityTable*
public void fillEntityTable()
 - **Usage**
 - * Load records from database and fill the entity view table and controller record list

- *fillEntityTable*
public void fillEntityTable(java.util.ArrayList lR)
 - **Usage**
 - * Fill the entity view table with the records given
 - **Parameters**
 - * lR - ArrayList. Records to be displayed

- *getClassEntity*
public Class getClassEntity()
 - **Returns** - Class of the entity for which the controller is built

- *getDialogName*
public String getDialogName()
- *getEditGladePath*
public String getEditGladePath()
- *getEditionGui*
public Gui getEditionGui()
 - **Usage**
 - * Returns the edition Gui instance.
 - **Returns** - Gui, entity edition Gui.

- *getEditWindow*
public SwtTopLevelWindow getEditWindow()
- *getEntityData*
public abstract BaseVO getEntityData()
- *getEntityFields*
public ArrayList getEntityFields()
- *getLogger*
public Logger getLogger()
- *getParentEntityLauncher*
public EntityLauncher getParentEntityLauncher()

- **Returns** - EntityLauncher of parent entity if is a embedded entity or null otherwise
- - *getRegistries*
 public Collection **getRegistries**()
 - **Returns** - IRegistries

- *getRegistries*
 public List **getRegistries**(java.util.ArrayList lIds)
 - **Usage**
 * Get a list of entities stored by the controller
 - **Parameters**
 * lIds - Index list
 - **Returns** - List of entities

- *getView*
 public AxiliceEntityView **getView**()
 - **Returns** - AxiliceEntityView. Main view or Detail view if is a embedded entity.

- *getWidgets*
 public abstract void **getWidgets**()
 - **Usage**
 * Get the widgets from the glade and stored in local variables.

Must be overwritten

- *init_specific_components*
 public abstract void **init_specific_components**()
 - **Usage**
 * Initializes the specific components of the client

- *printEntity*
 public OperationResult **printEntity**()

- *printEntity*
 public OperationResult **printEntity**(java.lang.String reportPath, java.util.HashMap parameters)

- *queryEntity*
 public ArrayList **queryEntity**()

- *queryEntity*
 public ArrayList **queryEntity**(java.lang.String fieldId, java.lang.String value)

- *queryEntityWithFormat*
 public ArrayList **queryEntityWithFormat**(java.lang.String fieldId, java.lang.String value)

- *queryFreeEntities*
 public ArrayList **queryFreeEntities**(com.imagos.axilice.core.common.MultiSelectionSubentityOT mSubE)
 - **Usage**
 * Search entities not associated with a parent entity (null parent field)
 - **Parameters**
 * mSubE - SubentityOT for the parent entity that must be null
 - **Returns** - ArrayList. List of entities found

- *querySubentities*
public ArrayList querySubentities(com.imagos.axilice.core.installer.SubentityOT entData)
- *refreshEntity*
public OperationResult refreshEntity()
- *refreshEntity*
public BaseVO refreshEntity(com.imagos.axilice.core.common.BaseVO entity)
- *refreshParentSelectedEntity*
public OperationResult refreshParentSelectedEntity()
 - Usage
 - * Reload from database the value of the parent entity of the selected entity
 - Returns - OperationResult. The result of the operation.
- *removeRegistries*
public OperationResult removeRegistries(java.util.ArrayList selRegistries, java.lang.String parentField)
 - Usage
 - * Deletes a list of the records stored by the controller. Also clean your parent fields and updates the database
 - Parameters
 - * selRegistries - Index list to remove
 - * parentField - Field name that links to the parent entity
 - Returns - OperationResult. Result of the operation
- *saveEntity*
public OperationResult saveEntity()
- *saveEntity*
public OperationResult saveEntity(java.util.ArrayList lIds)
- *saveEntity*
public OperationResult saveEntity(com.imagos.axilice.core.common.BaseVO b)
 - Usage
 - * Save or update a entity to database
 - Parameters
 - * b - BaseVO. Entity to save or update
 - Returns - OperationResult. Result of the operation
- *selectEntity*
public void selectEntity(java.lang.Long idEntity)
- *selectSubentity*
public void selectSubentity(java.lang.String parentFieldId, com.imagos.axilice.core.common.BaseVO subentityOT)
- *selectTreeItem*
public void selectTreeItem(java.lang.Long id)
 - Usage
 - * Select a item in the entity tree. No me gusta esto aqui
 - Parameters
 - * id - Long. Index of item to select
- *setDialogName*
public void setDialogName(java.lang.String dialogName)
- *setEditGladePath*
public void setEditGladePath(java.lang.String editGladePath)

- *setEditionMode*

public abstract void setEditionMode(boolean enabled)
- *setRegistries*
public void setRegistries(java.util.Collection collection)
 - **Usage**
 - * Setter for IRegistries

- *setWidgetsValues*
public abstract void setWidgetsValues(com.imagos.axilice.core.common.BaseVO entity)
 - **Usage**
 - * Set the values of the widgets on the edit dialog.
 - Must be overwritten.
 - **Parameters**
 - * **entity** - BaseVO. Entity that contains the values to be displayed

- *showErrorMessages*
public void showErrorMessages(com.imagos.axilice.core.util.OperationResult result)

- *updateEntityData*
public void updateEntityData()

- *updateRegistry*
public void updateRegistry(com.imagos.axilice.core.common.BaseVO ent)
 - **Usage**
 - * Add or update an entity (to the controller records)
 - **Parameters**
 - * **ent** - Entity to add or update

- *updateViews*
public void updateViews(boolean updateFromParent)
 - **Usage**
 - * Force the update of the views associated
 - **Parameters**
 - * **updateFromParent** - Indicates whether the view must be updated by reading the values from the parent entity. takes effect only when the entity is detail (is embedded in another)

- *validateEntity*
public abstract OperationResult validateEntity()
 - **Usage**
 - * Validate the values of the widgets in the edit dialog.
 - Must be overwritten.
 - **Returns** - OperationResult. Result of the validation.

- *viewEntity*
public OperationResult viewEntity()

METHODS INHERITED FROM CLASS com.imagos.axilice.core.installer.EntityLauncher

(in 24.2.2, page 532)

- *addEntity*
 public abstract ArrayList **addEntity**()
 – **Usage**
 * Add entity button clicked handler. Show the entity content screen ready to start inserting data for a new instance of the entity (new registry).
 – **Returns** - ArrayList<ResultMessage>, list of errors if any has taken place, null in any other case.

- *buildEditEntityGUI*
 public void **buildEditEntityGUI**()
 – **Usage**
 * Builds the edit entity dialog window from the corresponding glade interface file. Stores the entity dialog instance for posterior accesses.

- *buildEntity*
 public AxiliceEntityView **buildEntity**()
 – **Usage**
 * Builds the entity GUI and show it on screen to the user.

- *buildEntity*
 public AxiliceDetailEntityView **buildEntity**(SwtWidget **parent**,
 com.imagos.axilice.core.common.MultiSelectionSubentityOT **subentityData**,
 java.lang.String **rootName**)
 – **Usage**
 * Builds the entity GUI and show it on screen to the user. The entity GUI will be embedded into the parent composite widget.
 – **Parameters**
 * **parent** - Composite, parent holder for embedding the entity view.
 * **subentityData** - SubentityOT, data of the subentity to build if the entity launcher will be used to handle a multi-selection detail entity view.
 * **rootName** - String, name of the root widget that will be embedded into the master entity edition panel.
 – **Returns** - AxiliceDetailEntityView, the entity detail view built.

- *cancelEntityEdition*
 public void **cancelEntityEdition**()

- *clearEntityData*
 public abstract void **clearEntityData**()
 – **Usage**
 * Clears the entity content panel by resetting the content widgets.

- *delEntity*
 public abstract OperationResult **delEntity**()
 – **Usage**
 * Delete entity button clicked handler. Deletes the selected instance of the entity (selected registry) from the database.
 – **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *delEntity*

```
public abstract OperationResult delEntity( java.util.ArrayList lSelRegistries )
```

- **Usage**

- * Delete entity button clicked handler. Deletes the selected instances of the entity (selected registries) from the database. Its a multiple delete method.

- **Parameters**

- * lSelRegistries - ArrayList<Long>, list of selected instances of the entity (selected registries). The list will contain only the internal id of the selected registries.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *deselectSubentity*

```
public void deselectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

- **Usage**

- * Updates an entity instance by removing the selected value for a subentity field.

- **Parameters**

- * parentFieldId - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.

- * subentityOT - BaseVO, subentity's deselected instance (registry) value.

- *editEntity*

```
public abstract OperationResult editEntity( )
```

- **Usage**

- * Edit entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), and allows the user to modify any content of the entity registry.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *fillEntityData*

```
public abstract void fillEntityData( )
```

- **Usage**

- * Fills the entity content panel with the data of the selected instance (registry) of the entity.

- *fireExceptionAndExit*

```
public void fireExceptionAndExit( java.lang.String msg )
```

- **Usage**

- * Fire a exception and stop the program

- **Parameters**

- * msg -

- *formatRegistries*

```
public ArrayList formatRegistries( )
```

- *formatRegistries*

```
public ArrayList formatRegistries( java.util.ArrayList lRegistries )
```

- *getDialog*

```
public SwtDialog getDialog( java.lang.String dialogName )
```

- *getEntity*

```
public EntityOT getEntity( )
```

- **Usage**

- * Gets the entity data.

- **Returns** - EntityOT, entity data.
- - *getEntityData*
 public abstract BaseVO **getEntityData**()
 - **Usage**
 - * Gets the content panel's data typed by the user formatted as an entity VO object.
- - *getEntityEditViewName*
 public String **getEntityEditViewName**(java.lang.String parentGuiName)
- - *getEntityFields*
 public abstract ArrayList **getEntityFields**()
 - **Usage**
 - * Gets the list of defined fields for an entity
 - **Returns** - ArrayList, list of entity fields.
- - *getEntityViewName*
 public String **getEntityViewName**(java.lang.String parentGuiName)
- - *getLRegistries*
 public ArrayList **getLRegistries**()
 - **Usage**
 - * Getter for lRegistries
- - *getRegistry*
 public BaseVO **getRegistry**(long id)
- - *getRevision*
 public static String **getRevision**()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
- - *getSelectedEntity*
 public BaseVO **getSelectedEntity**()
 - **Usage**
 - * Gets the selected instance (registry) of the entity.
 - **Returns** - BaseVO, selected instance (registry) of the entity.
- - *getSelectedId*
 public Long **getSelectedId**()
 - **Usage**
 - * Gets the internal id of the selected entity registry.
 - **Returns** - Long, internal id of the selected entity registry, null in case no entity registry has been selected yet.
- - *getSwtWidget*
 public SwtWidget **getSwtWidget**(java.lang.String widgetName)
- - *getWidget*
 public Widget **getWidget**(java.lang.String widgetName)
- - *initEntityLauncher*
 public void **initEntityLauncher**(Logger logger, Gui parentGui, SwtWidget parentPanel)
 - **Usage**
 - * Inits the entity launcher context and prepares the launcher to be started.
 - **Parameters**

- * **logger** - Logger, logger instance for printing logging messages.
- * **parentGuiName** - String, name of the JavaGlade instance that built the parent where the entity will be embedded.
- * **parentPanel** - Composite, parent panel where the entity contents will be embedded.

- *isDetail*

public boolean **isDetail**()

– **Usage**

- * Indicates whether is a main entity or is embedded in another

- *on_entitySelectionButton_clicked*

public void **on_entitySelectionButton_clicked**(Event e)

– **Usage**

- * Handler for the subentity selection buttons (magnifying glass buttons) clicked. Shows the selection dialog for selecting a valid instance of a subentity.

– **Parameters**

- * **e** - Event, magnifying glass button clicked event data.

- *printEntity*

public abstract **OperationResult printEntity**()

– **Usage**

- * Print entity button clicked handler. Builds the report with the data of the selected instance of the entity (selected registry), and allows the user to print the elaborated report.

– **Returns** - **OperationResult**, result of the delete operation and list of messages generated during the operation.

- *printWidgets*

public void **printWidgets**()

- *queryEntity*

public abstract **ArrayList queryEntity**()

– **Usage**

- * Search entity button clicked handler. Searches the system for the entity instances (registries) available.

– **Returns** - **ArrayList**, list of entity instances (registries) taken as the result of the query operation.

- *queryEntity*

public abstract **ArrayList queryEntity**(java.lang.String **fieldId**, java.lang.String **value**)

– **Usage**

- * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria.

– **Parameters**

- * **fieldId** - String, name of the entity field that will be queried.
- * **value** - String, value of the entity field that will be searched for.

– **Returns** - **ArrayList**, list of entity instances (registries) taken as the result of the query operation.

- *queryEntityWithFormat*

public abstract **ArrayList queryEntityWithFormat**(java.lang.String **fieldId**, java.lang.String **value**)

– **Usage**

- * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria and show them (formatted as a list) to the user in the entity's screen.

– **Parameters**

- * **fieldId** - String, name of the entity field that will be queried.
- * **value** - String, value of the entity field that will be searched for.

- **Returns** - ArrayList<RegistryOT>, list of entity instances (registries) taken as the result of the query operation.

• *querySubentities*

```
public abstract ArrayList querySubentities(
com.imagos.axilice.core.installer.SubentityOT entData )
```

– **Usage**

- * Queries the selected entity for all the detail subentities associated with it.

– **Parameters**

- * **entData** - SubentityOT, data of the detail subentity to query for.

- **Returns** - ArrayList, list of detail subentity registries assigned to the master detail registry.

• *saveEntity*

```
public abstract OperationResult saveEntity( )
```

– **Usage**

- * Save entity button clicked handler. Update the system with the data of the edited entity registry. It's called for saving new entity registries (add a new registry) or for updating the contents of existing entity registries.

- **Returns** - OperationResult, Result of the operation and list of messages generated through the operation.

• *saveEntityRegistry*

```
public void saveEntityRegistry( )
```

• *selectEntity*

```
public abstract void selectEntity( java.lang.Long idEntity )
```

– **Usage**

- * Marks an instance (registry) of the entity as the selected.

– **Parameters**

- * **idEntity** - Long, internal id of the selected instance (registry).

• *selectSubentity*

```
public void selectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

– **Usage**

- * Updates an entity instance by setting up the selected value for a subentity field.

– **Parameters**

- * **parentFieldId** - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
- * **subentityOT** - BaseVO, subentity's selected instance (registry) value.

• *setEntity*

```
public void setEntity( com.imagos.axilice.core.installer.EntityOT entity )
```

– **Usage**

- * Sets up the entity name.

– **Parameters**

- * **name** - String, entity name.
-

- *setRegistries*

```
public void setRegistries( java.util.ArrayList IRegistries )
```

- Usage

- * Sets the list of registries that are showed in the entity view controlled by the Entity Launcher instance. It's used for detail entities where queries are made in the master entity launcher.

- Parameters

- * lRegistries - ArrayList<BaseVO>, list of shown registries of the entity.

- *showSelectionDialog*

```
public void showSelectionDialog( com.imagos.axilice.core.installer.SubentityOT entData )
```

- Usage

- * Opens the selection dialog that enables the user to select an instance of a subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.

- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT entData, long selEntity, boolean readonly )
```

- Usage

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.
 - * selEntity - long, id of the selected instance of the subentity.
 - * readonly - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT entData, java.lang.String parentGuiName, long selEntity, boolean readonly )
```

- Usage

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.
 - * parentGuiName - String,
 - * selEntity - long, id of the selected instance of the subentity.
 - * readonly - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

- *showSubentitySelectionDialog*

```
public void showSubentitySelectionDialog( com.imagos.axilice.core.installer.SubentityOT entData )
```

- Usage

- * Opens the selection dialog that enables the user to select an instance of a multiple selection detail subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.

- *undoEntityEdition*

```
public void undoEntityEdition( )
```

- *updateEntityData*

```
public abstract void updateEntityData( )
```

- **Usage**

- * Updates the selected instance (registry) data with the content panel's data typed by the user.
-

- *viewEntity*

```
public abstract OperationResult viewEntity( )
```

- **Usage**

- * View entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), in read-only mode.

- **Returns** - **OperationResult**, result of the delete operation and list of messages generated during the operation.

Chapter 16

Package com.imagos.axilice.server.management.bl

<i>Package Contents</i>	<i>Page</i>
<hr/>	
Classes	
ManagementBLL.....	392
...no description...	
<hr/>	

16.1 Classes

16.1.1 CLASS ManagementBLL

DECLARATION

```
public class ManagementBLL
extends java.lang.Object
```

METHODS

- *delEntities*

```
public OperationResult delEntities( java.util.ArrayList  lCriteria,
    java.lang.Class  myClass )
```
- *getInstance*

```
public static final ManagementBLL getInstance( )
```

 - **Usage**
 - * Returns the unique instance of the ManagementBLL class. Implements the singleton design pattern.
 - **Returns** - ManagementBLL, unique instance of ManagementBLL.
- *getProductStock*

```
public ArrayList getProductStock(
    com.imagos.axilice.core.warehouse.ProductOT  product,
    com.imagos.axilice.core.warehouse.WarehouseOT  warehouse )
```
- *nextSerialNumber*

```
public long nextSerialNumber(
    com.imagos.axilice.core.management.SerialNumberOT  serial )
```


Chapter 17

Package com.imagos.axilice.client.gui.common

<i>Package Contents</i>	<i>Page</i>
<hr/>	
Classes	
BarcodeTypeClient	394
<i>...no description...</i>	
CommonClient	402
<i>...no description...</i>	
CurrencyClient	415
<i>...no description...</i>	
MapView	428
<i>...no description...</i>	
PriceListClient	429
<i>...no description...</i>	
PriceListLineClient	442
<i>...no description...</i>	
VatTypeClient	455
<i>...no description...</i>	

17.1 Classes

17.1.1 CLASS BarcodeTypeClient

DECLARATION

```
public class BarcodeTypeClient
extends com.imagos.axilice.core.installer.EntityLauncher
```

FIELDS

- public static final String AXILICE_LOG_EDITBARCODETYPE_GLADEPATH
– Message for logging the entity glade configuration file path.

CONSTRUCTORS

- *BarcodeTypeClient*
public **BarcodeTypeClient**()

METHODS

- *addEntity*
public ArrayList **addEntity**()
- *clearEntityData*
public void **clearEntityData**()
- *delEntity*
public **OperationResult** **delEntity**()
- *delEntity*
public **OperationResult** **delEntity**(java.util.ArrayList lSelRegistries)
- *editEntity*
public **OperationResult** **editEntity**()
- *fillEntityData*
public void **fillEntityData**()
- *getEntityData*
public BaseVO **getEntityData**()
- *getEntityFields*
public ArrayList **getEntityFields**()
- *getSelectedEntity*
public BaseVO **getSelectedEntity**()

- *getSelectedId*
public Long **getSelectedId**()
- *printEntity*
public OperationResult **printEntity**()
- *queryEntity*
public ArrayList **queryEntity**()
- *queryEntity*
public ArrayList **queryEntity**(java.lang.String fieldId, java.lang.String value)
- *queryEntityWithFormat*
public ArrayList **queryEntityWithFormat**(java.lang.String fieldId, java.lang.String value)
- *querySubentities*
public ArrayList **querySubentities**(
com.imagos.axilice.core.installer.SubentityOT entData)
 - **Usage**
 - * Queries the selected entity for all the detail subentities associated with it.
 - **Parameters**
 - * entData - SubentityOT, data of the detail subentity to query for.
 - **Returns** - ArrayList, list of detail subentity registries assigned to the master detail registry.
- *saveEntity*
public OperationResult **saveEntity**()
- *selectEntity*
public void **selectEntity**(java.lang.Long idEntity)
- *selectSubentity*
public void **selectSubentity**(java.lang.String parentFieldId, com.imagos.axilice.core.common.BaseV0 subentityOT)
- *updateEntityData*
public void **updateEntityData**()
- *viewEntity*
public OperationResult **viewEntity**()

METHODS INHERITED FROM CLASS com.imagos.axilice.core.installer.EntityLauncher

(in 24.2.2, page 532)

- *addEntity*
public abstract ArrayList **addEntity**()
 - **Usage**
 - * Add entity button clicked handler. Show the entity content screen ready to start inserting data for a new instance of the entity (new registry).
 - **Returns** - ArrayList<ResultMessage>, list of errors if any has taken place, null in any other case.

-
- *buildEditEntityGUI*
 public void **buildEditEntityGUI**()
 – **Usage**
 * Builds the edit entity dialog window from the corresponding glade interface file. Stores the entity dialog instance for posterior accesses.

 - *buildEntity*
 public AxiliceEntityView **buildEntity**()
 – **Usage**
 * Builds the entity GUI and show it on screen to the user.

 - *buildEntity*
 public AxiliceDetailEntityView **buildEntity**(SwtWidget parent,
 com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData,
 java.lang.String rootName)
 – **Usage**
 * Builds the entity GUI and show it on screen to the user. The entity GUI will be embedded into the parent composite widget.
 – **Parameters**
 * **parent** - Composite, parent holder for embedding the entity view.
 * **subentityData** - SubentityOT, data of the subentity to build if the entity launcher will be used to handle a multi-selection detail entity view.
 * **rootName** - String, name of the root widget that will be embedded into the master entity edition panel.
 – **Returns** - AxiliceDetailEntityView, the entity detail view built.

 - *cancelEntityEdition*
 public void **cancelEntityEdition**()

 - *clearEntityData*
 public abstract void **clearEntityData**()
 – **Usage**
 * Clears the entity content panel by resetting the content widgets.

 - *delEntity*
 public abstract OperationResult **delEntity**()
 – **Usage**
 * Delete entity button clicked handler. Deletes the selected instance of the entity (selected registry) from the database.
 – **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

 - *delEntity*
 public abstract OperationResult **delEntity**(java.util.ArrayList lSelRegistries)
 – **Usage**
 * Delete entity button clicked handler. Deletes the selected instances of the entity (selected registries) from the database. Its a multiple delete method.
 – **Parameters**
 * **lSelRegistries** - ArrayList<Long>, list of selected instances of the entity (selected registries). The list will contain only the internal id of the selected registries.
 – **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *deselectSubentity*

```
public void deselectSubentity( java.lang.String  parentFieldId,
com.imagos.axilice.core.common.BaseVO  subentityOT )
```

- **Usage**

- * Updates an entity instance by removing the selected value for a subentity field.

- **Parameters**

- * **parentFieldId** - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * **subentityOT** - BaseVO, subentity's deselected instance (registry) value.

- *editEntity*

```
public abstract OperationResult editEntity( )
```

- **Usage**

- * Edit entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), and allows the user to modify any content of the entity registry.

- **Returns** - **OperationResult**, result of the delete operation and list of messages generated during the operation.

- *fillEntityData*

```
public abstract void fillEntityData( )
```

- **Usage**

- * Fills the entity content panel with the data of the selected instance (registry) of the entity.

- *fireExceptionAndExit*

```
public void fireExceptionAndExit( java.lang.String  msg )
```

- **Usage**

- * Fire a exception and stop the program

- **Parameters**

- * **msg** -

- *formatRegistries*

```
public ArrayList formatRegistries( )
```

- *formatRegistries*

```
public ArrayList formatRegistries( java.util.ArrayList  IRegistries )
```

- *getDialog*

```
public SwtDialog getDialog( java.lang.String  dialogName )
```

- *getEntity*

```
public EntityOT getEntity( )
```

- **Usage**

- * Gets the entity data.

- **Returns** - **EntityOT**, entity data.

- *getEntityData*

```
public abstract BaseVO getEntityData( )
```

- **Usage**

- * Gets the content panel's data typed by the user formatted as an entity VO object.

- *getEntityEditViewName*

```
public String getEntityEditViewName( java.lang.String  parentGuiName )
```

- *getEntityFields*

```
public abstract ArrayList getEntityFields( )
```

- **Usage**
 - * Gets the list of defined fields for an entity
 - **Returns** - ArrayList, list of entity fields.

- *getEntityViewName*
public String getEntityViewName(java.lang.String parentGuiName)
- *getLRegistries*
public ArrayList getLRegistries()
 - **Usage**
 - * Getter for lRegistries

- *getRegistry*
public BaseVO getRegistry(long id)
- *getRevision*
public static String getRevision()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.

- *getSelectedEntity*
public BaseVO getSelectedEntity()
 - **Usage**
 - * Gets the selected instance (registry) of the entity.
 - **Returns** - BaseVO, selected instance (registry) of the entity.

- *getSelectedId*
public Long getSelectedId()
 - **Usage**
 - * Gets the internal id of the selected entity registry.
 - **Returns** - Long, internal id of the selected entity registry, null in case no entity registry has been selected yet.

- *getSwtWidget*
public SwtWidget getSwtWidget(java.lang.String widgetName)
- *getWidget*
public Widget getWidget(java.lang.String widgetName)
- *initEntityLauncher*
public void initEntityLauncher(Logger logger, Gui parentGui, SwtWidget parentPanel)
 - **Usage**
 - * Inits the entity launcher context and prepares the launcher to be started.
 - **Parameters**
 - * **logger** - Logger, logger instance for printing logging messages.
 - * **parentGuiName** - String, name of the JavaGlade instance that built the parent where the entity will be embedded.
 - * **parentPanel** - Composite, parent panel where the entity contents will be embedded.

- *isDetail*
public boolean isDetail()
 - **Usage**
 - * Indicates whether is a main entity or is embedded in another

- *on_entitySelectionButton_clicked*
public void on_entitySelectionButton_clicked(Event e)

- **Usage**
 - * Handler for the subentity selection buttons (magnifying glass buttons) clicked. Shows the selection dialog for selecting a valid instance of a subentity.
 - **Parameters**
 - * e - Event, magnifying glass button clicked event data.
-
- *printEntity*
 public abstract OperationResult **printEntity**()
 - **Usage**
 - * Print entity button clicked handler. Builds the report with the data of the selected instance of the entity (selected registry), and allows the user to print the elaborated report.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.
-
- *printWidgets*
 public void **printWidgets**()
-
- *queryEntity*
 public abstract ArrayList **queryEntity**()
 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) available.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.
-
- *queryEntity*
 public abstract ArrayList **queryEntity**(java.lang.String fieldId, java.lang.String value)
 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria.
 - **Parameters**
 - * fieldId - String, name of the entity field that will be queried.
 - * value - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.
-
- *queryEntityWithFormat*
 public abstract ArrayList **queryEntityWithFormat**(java.lang.String fieldId, java.lang.String value)
 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria and show them (formatted as a list) to the user in the entity's screen.
 - **Parameters**
 - * fieldId - String, name of the entity field that will be queried.
 - * value - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList<RegistryOT>, list of entity instances (registries) taken as the result of the query operation.
-
- *querySubentities*
 public abstract ArrayList **querySubentities**(com.imagos.axilice.core.installer.SubentityOT entData)
 - **Usage**

- * Queries the selected entity for all the detail subentities associated with it.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to query for.
 - **Returns** - ArrayList, list of detail subentity registries assigned to the master detail registry.

- *saveEntity*

```
public abstract OperationResult saveEntity( )
```

 - **Usage**
 - * Save entity button clicked handler. Update the system with the data of the edited entity registry. It's called for saving new entity registries (add a new registry) or for updating the contents of existing entity registries.
 - **Returns** - OperationResult, Result of the operation and list of messages generated through the operation.

- *saveEntityRegistry*

```
public void saveEntityRegistry( )
```
- *selectEntity*

```
public abstract void selectEntity( java.lang.Long idEntity )
```

 - **Usage**
 - * Marks an instance (registry) of the entity as the selected.
 - **Parameters**
 - * **idEntity** - Long, internal id of the selected instance (registry).

- *selectSubentity*

```
public void selectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

 - **Usage**
 - * Updates an entity instance by setting up the selected value for a subentity field.
 - **Parameters**
 - * **parentFieldId** - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been set up.
 - * **subentityOT** - BaseVO, subentity's selected instance (registry) value.

- *setEntity*

```
public void setEntity( com.imagos.axilice.core.installer.EntityOT entity )
```

 - **Usage**
 - * Sets up the entity name.
 - **Parameters**
 - * **name** - String, entity name.

- *setRegistries*

```
public void setRegistries( java.util.ArrayList lRegistries )
```

 - **Usage**
 - * Sets the list of registries that are showed in the entity view controlled by the Entity Launcher instance. It's used for detail entities where queries are made in the master entity launcher.
 - **Parameters**
 - * **lRegistries** - ArrayList<BaseVO>, list of shown registries of the entity.

- *showSelectionDialog*

```
public void showSelectionDialog( com.imagos.axilice.core.installer.SubentityOT
entData )
```

 - **Usage**

* Opens the selection dialog that enables the user to select an instance of a subentity.

– **Parameters**

* **entData** - SubentityOT, data of the detail subentity to select.

• *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT
entData, long selEntity, boolean readonly )
```

– **Usage**

* Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

– **Parameters**

* **entData** - SubentityOT, data of the detail subentity to select.

* **selEntity** - long, id of the selected instance of the subentity.

* **readonly** - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

• *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT
entData, java.lang.String parentGuiName, long selEntity, boolean readonly )
```

– **Usage**

* Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

– **Parameters**

* **entData** - SubentityOT, data of the detail subentity to select.

* **parentGuiName** - String,

* **selEntity** - long, id of the selected instance of the subentity.

* **readonly** - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

• *showSubentitySelectionDialog*

```
public void showSubentitySelectionDialog(
com.imagos.axilice.core.installer.SubentityOT entData )
```

– **Usage**

* Opens the selection dialog that enables the user to select an instance of a multiple selection detail subentity.

– **Parameters**

* **entData** - SubentityOT, data of the detail subentity to select.

• *undoEntityEdition*

```
public void undoEntityEdition( )
```

• *updateEntityData*

```
public abstract void updateEntityData( )
```

– **Usage**

* Updates the selected instance (registry) data with the content panel's data typed by the user.

• *viewEntity*

```
public abstract OperationResult viewEntity( )
```

– **Usage**

* View entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), in read-only mode.

– **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

17.1.2 CLASS *CommonClient*

DECLARATION

```
public abstract class CommonClient
extends com.imagos.axilice.core.installer.EntityLauncher
```

CONSTRUCTORS

- *CommonClient*
 public **CommonClient**()
- *CommonClient*
 public **CommonClient**(java.lang.Class **classEntity**, java.lang.String **editGlade**, java.lang.String **dialogN**)
 - **Usage**
 - * Default constructor
 - **Parameters**
 - * **classEntity** - Class (POJO) of the entity for which the client is built
 - * **editGlade** - Glade file name
 - * **dialogN** - Dialog name inside the glade file

METHODS

- *addAxiliceEventListener*
 public void **addAxiliceEventListener**(
 com.imagos.axilice.client.gui.AxiliceEventListener **l**)
- *addEntity*
 public ArrayList **addEntity**()
- *addExistingSubentity*
 public void **addExistingSubentity**(
 com.imagos.axilice.core.installer.SubentityOT **ent**)
 - **Usage**
 - * Shows a dialog for selection of a new subentity
 - **Parameters**
 - * **ent** - SubentityOT. Encapsulate data type of subentity
- *addSubentity*
 public void **addSubentity**(com.imagos.axilice.core.common.BaseVO **subentity**,
 com.imagos.axilice.core.common.MultiSelectionSubentityOT **subE**)
 - **Usage**
 - * Adds a existent entity to a multiselection list
 - **Parameters**

- * **subentity** - Subentity to be added
- * **subE** - Object that encapsulates information about the subentity

- *addTrazabilityTree*

public void addTrazabilityTree(TabFolder tabFolder)

- *build_extension_fields_view*

public abstract void build_extension_fields_view()

- **Usage**

- * Build the view of the extension fields.

It must implement only those clients who require it

- *buildEntity*

public AxiliceEntityView buildEntity()

- *buildEntityDetail*

**public AxiliceEntityView buildEntityDetail(
com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData)**

- *buildEntitySelectionView*

public AxiliceEntitySelectionView buildEntitySelectionView()

- **Usage**

- * Build view of selection entities

- **Returns** - AxilicePartnerSelectionView. SelectionView created

- *buildSelectionSets*

public abstract void buildSelectionSets()

- **Usage**

- * Build 2 sets of SubentityOT: lSelectionSubentities and lMultiSelectionSubentities

These sets are used to connect the widgets with the handlers for the selection buttons and the multiselección panels

It must implement only those clients who require it

- *buildSubentityMultiSelection*

public void buildSubentityMultiSelection()

- *clearEntityData*

public void clearEntityData()

- *clearParentField*

**public OperationResult clearParentField(java.util.ArrayList selRegistries,
java.lang.String parentField)**

- **Usage**

- * Clears the value of the field indicated

- **Parameters**

- * **selRegistries** - ArrayList. Index list of records to process
 - * **parentField** - Name of the field to clear

– **Returns** - OperationResult. Result of operation

• *delEntity*

public OperationResult delEntity()

• *delEntity*

public OperationResult delEntity(java.util.ArrayList selRegistries)

• *delEntity*

public OperationResult delEntity(com.imagos.axilice.core.common.BaseVO b)

– **Usage**

* Delete a entity from database

– **Parameters**

* b - BaseVO. Entity to delete

– **Returns** - OperationResult. Result of the operation

• *delSubentities*

public OperationResult delSubentities(java.util.ArrayList selRegistries)

– **Usage**

* Delete a set of subentities from record list of controller, not from database

Set the parent field of subentity to null

– **Parameters**

* selRegistries - List of index to delete

– **Returns** - OperationResult. Result of the operation

• *editEntity*

public OperationResult editEntity()

• *fillEntityData*

public void fillEntityData()

• *fillEntityExtensionTable*

public void fillEntityExtensionTable(java.lang.Long id)

– **Usage**

* Retrieve extension field values from the server for a given entity and displays them in the view

– **Parameters**

* id - Long. Key of the entity whose values are to be displayed

• *fillEntityTable*

public void fillEntityTable()

– **Usage**

* Load records from database and fill the entity view table and controller record list

• *fillEntityTable*

public void fillEntityTable(java.util.ArrayList IR)

- **Usage**
 - * Fill the entity view table with the records given
 - **Parameters**
 - * lR - ArrayList. Records to be displayed

- *getClassEntity*
 public Class **getClassEntity**()
 - **Returns** - Class of the entity for which the controller is built

- *getDialogName*
 public String **getDialogName**()

- *getEditGladePath*
 public String **getEditGladePath**()

- *getEditionGui*
 public Gui **getEditionGui**()
 - **Usage**
 - * Returns the edition Gui instance.
 - **Returns** - Gui, entity edition Gui.

- *getEditWindow*
 public SwtTopLevelWindow **getEditWindow**()

- *getEntityData*
 public abstract BaseV0 **getEntityData**()

- *getEntityFields*
 public ArrayList **getEntityFields**()

- *getLogger*
 public Logger **getLogger**()

- *getParentEntityLauncher*
 public EntityLauncher **getParentEntityLauncher**()
 - **Returns** - EntityLauncher of parent entity if is a embedded entity or null otherwise

- *getRegistries*
 public Collection **getRegistries**()
 - **Returns** - lRegistries

- *getRegistries*
 public List **getRegistries**(java.util.ArrayList lIds)
 - **Usage**
 - * Get a list of entities stored by the controller
 - **Parameters**
 - * lIds - Index list
 - **Returns** - List of entities

- *getView*
 public AxiliceEntityView **getView**()
 – **Returns** - AxiliceEntityView. Main view or Detail view if is a embedded entity.

- *getWidgets*
 public abstract void **getWidgets**()
 – **Usage**
 * Get the widgets from the glade and stored in local variables.

 Must be overwritten

- *init_specific_components*
 public abstract void **init_specific_components**()
 – **Usage**
 * Initializes the specific components of the client

- *printEntity*
 public OperationResult **printEntity**()

- *printEntity*
 public OperationResult **printEntity**(java.lang.String **reportPath**,
 java.util.HashMap **parameters**)

- *queryEntity*
 public ArrayList **queryEntity**()

- *queryEntity*
 public ArrayList **queryEntity**(java.lang.String **fieldId**, java.lang.String
value)

- *queryEntityWithFormat*
 public ArrayList **queryEntityWithFormat**(java.lang.String **fieldId**,
 java.lang.String **value**)

- *queryFreeEntities*
 public ArrayList **queryFreeEntities**(
 com.imagos.axilice.core.common.MultiSelectionSubentityOT **mSubE**)
 – **Usage**
 * Search entities not associated with a parent entity (null parent field)
 – **Parameters**
 * **mSubE** - SubentityOT for the parent entity that must be null
 – **Returns** - ArrayList. List of entities found

- *querySubentities*
 public ArrayList **querySubentities**(
 com.imagos.axilice.core.installer.SubentityOT **entData**)

- *refreshEntity*
 public OperationResult **refreshEntity**()

- *refreshEntity*

```
public BaseVO refreshEntity( com.imagos.axilice.core.common.BaseVO entity )
```

- *refreshParentSelectedEntity*

```
public OperationResult refreshParentSelectedEntity( )
```

- **Usage**

- * Reload from database the value of the parent entity of the selected entity

- **Returns** - OperationResult. The result of the operation.

- *removeRegistries*

```
public OperationResult removeRegistries( java.util.ArrayList selRegistries,
java.lang.String parentField )
```

- **Usage**

- * Deletes a list of the records stored by the controller. Also clean your parent fields and updates the database

- **Parameters**

- * selRegistries - Index list to remove

- * parentField - Field name that links to the parent entity

- **Returns** - OperationResult. Result of the operation

- *saveEntity*

```
public OperationResult saveEntity( )
```

- *saveEntity*

```
public OperationResult saveEntity( java.util.ArrayList lIds )
```

- *saveEntity*

```
public OperationResult saveEntity( com.imagos.axilice.core.common.BaseVO b
)
```

- **Usage**

- * Save or update a entity to database

- **Parameters**

- * b - BaseVO. Entity to save or update

- **Returns** - OperationResult. Result of the operation

- *selectEntity*

```
public void selectEntity( java.lang.Long idEntity )
```

- *selectSubentity*

```
public void selectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

- *selectTreeItem*

```
public void selectTreeItem( java.lang.Long id )
```

- **Usage**

- * Select a item in the entity tree. No me gusta esto aqui

- **Parameters**

- * id - Long. Index of item to select

- *setDialogName*

public void setDialogName(java.lang.String dialogName)
- *setEditGladePath*

public void setEditGladePath(java.lang.String editGladePath)
- *setEditionMode*

public abstract void setEditionMode(boolean enabled)
- *setRegistries*
public void setRegistries(java.util.Collection collection)
 - **Usage**
 - * Setter for IRegistries

- *setWidgetsValues*
public abstract void setWidgetsValues(com.imagos.axilice.core.common.BaseVO entity)
 - **Usage**
 - * Set the values of the widgets on the edit dialog.
 - Must be overwritten.
 - **Parameters**
 - * **entity** - BaseVO. Entity that contains the values to be displayed

- *showErrorMessages*

public void showErrorMessages(com.imagos.axilice.core.util.OperationResult result)

- *updateEntityData*

public void updateEntityData()

- *updateRegistry*
public void updateRegistry(com.imagos.axilice.core.common.BaseVO ent)
 - **Usage**
 - * Add or update an entity (to the controller records)
 - **Parameters**
 - * **ent** - Entity to add or update

- *updateViews*
public void updateViews(boolean updateFromParent)
 - **Usage**
 - * Force the update of the views associated
 - **Parameters**
 - * **updateFromParent** - Indicates whether the view must be updated by reading the values from the parent entity. takes effect only when the entity is detail (is embedded in another)

- *validateEntity*

public abstract OperationResult validateEntity()

- **Usage**

- * Validate the values of the widgets in the edit dialog.

Must be overwritten.

- **Returns** - `OperationResult`. Result of the validation.

- *viewEntity*

```
public OperationResult viewEntity( )
```

METHODS INHERITED FROM CLASS `com.imagos.axilice.core.installer.EntityLauncher`

(in 24.2.2, page 532)

- *addEntity*

```
public abstract ArrayList addEntity( )
```

- **Usage**

- * Add entity button clicked handler. Show the entity content screen ready to start inserting data for a new instance of the entity (new registry).

- **Returns** - `ArrayList<ResultMessage>`, list of errors if any has taken place, null in any other case.

- *buildEditEntityGUI*

```
public void buildEditEntityGUI( )
```

- **Usage**

- * Builds the edit entity dialog window from the corresponding glade interface file. Stores the entity dialog instance for posterior accesses.

- *buildEntity*

```
public AxiliceEntityView buildEntity( )
```

- **Usage**

- * Builds the entity GUI and show it on screen to the user.

- *buildEntity*

```
public AxiliceDetailEntityView buildEntity( SwtWidget parent,  
com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData,  
java.lang.String rootName )
```

- **Usage**

- * Builds the entity GUI and show it on screen to the user. The entity GUI will be embedded into the parent composite widget.

- **Parameters**

- * **parent** - Composite, parent holder for embedding the entity view.
- * **subentityData** - SubentityOT, data of the subentity to build if the entity launcher will be used to handle a multi-selection detail entity view.
- * **rootName** - String, name of the root widget that will be embedded into the master entity edition panel.

- **Returns** - `AxiliceDetailEntityView`, the entity detail view built.

- *cancelEntityEdition*

```
public void cancelEntityEdition( )
```

- *clearEntityData*

```
public abstract void clearEntityData( )
```

- **Usage**

* Clears the entity content panel by resetting the content widgets.

- *delEntity*

public abstract **OperationResult delEntity()**

- **Usage**

- * Delete entity button clicked handler. Deletes the selected instance of the entity (selected registry) from the database.

- **Returns** - **OperationResult**, result of the delete operation and list of messages generated during the operation.

- *delEntity*

public abstract **OperationResult delEntity(java.util.ArrayList lSelRegistries)**

- **Usage**

- * Delete entity button clicked handler. Deletes the selected instances of the entity (selected registries) from the database. Its a multiple delete method.

- **Parameters**

- * **lSelRegistries** - **ArrayList<Long>**, list of selected instances of the entity (selected registries). The list will contain only the internal id of the selected registries.

- **Returns** - **OperationResult**, result of the delete operation and list of messages generated during the operation.

- *deselectSubentity*

public void **deselectSubentity(java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT)**

- **Usage**

- * Updates an entity instance by removing the selected value for a subentity field.

- **Parameters**

- * **parentFieldId** - **String**, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.

- * **subentityOT** - **BaseVO**, subentity's deselected instance (registry) value.

- *editEntity*

public abstract **OperationResult editEntity()**

- **Usage**

- * Edit entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), and allows the user to modify any content of the entity registry.

- **Returns** - **OperationResult**, result of the delete operation and list of messages generated during the operation.

- *fillEntityData*

public abstract void **fillEntityData()**

- **Usage**

- * Fills the entity content panel with the data of the selected instance (registry) of the entity.

- *fireExceptionAndExit*

public void **fireExceptionAndExit(java.lang.String msg)**

- **Usage**

- * Fire a exception and stop the program

- **Parameters**

- * **msg** -

- *formatRegistries*
public ArrayList formatRegistries()
- *formatRegistries*
public ArrayList formatRegistries(java.util.ArrayList lRegistries)
- *getDialog*
public SwtDialog getDialog(java.lang.String dialogName)
- *getEntity*
public EntityOT getEntity()
 - **Usage**
 - * Gets the entity data.
 - **Returns** - EntityOT, entity data.

- *getEntityData*
public abstract BaseVO getEntityData()
 - **Usage**
 - * Gets the content panel's data typed by the user formatted as an entity VO object.

- *getEntityEditViewName*
public String getEntityEditViewName(java.lang.String parentGuiName)
- *getEntityFields*
public abstract ArrayList getEntityFields()
 - **Usage**
 - * Gets the list of defined fields for an entity
 - **Returns** - ArrayList, list of entity fields.

- *getEntityViewName*
public String getEntityViewName(java.lang.String parentGuiName)
- *getLRegistries*
public ArrayList getLRegistries()
 - **Usage**
 - * Getter for lRegistries

- *getRegistry*
public BaseVO getRegistry(long id)
- *getRevision*
public static String getRevision()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.

- *getSelectedEntity*
public BaseVO getSelectedEntity()
 - **Usage**
 - * Gets the selected instance (registry) of the entity.
 - **Returns** - BaseVO, selected instance (registry) of the entity.

- *getSelectedId*
public Long getSelectedId()
 - **Usage**
 - * Gets the internal id of the selected entity registry.
 - **Returns** - Long, internal id of the selected entity registry, null in case no entity registry has been selected yet.

- *getSwtWidget*
public SwtWidget **getSwtWidget**(java.lang.String widgetName)
- *getWidget*
public Widget **getWidget**(java.lang.String widgetName)
- *initEntityLauncher*
public void **initEntityLauncher**(Logger logger, Gui parentGui, SwtWidget parentPanel)
 - **Usage**
 - * Initiates the entity launcher context and prepares the launcher to be started.
 - **Parameters**
 - * **logger** - Logger, logger instance for printing logging messages.
 - * **parentGui** - String, name of the JavaGlade instance that built the parent where the entity will be embedded.
 - * **parentPanel** - Composite, parent panel where the entity contents will be embedded.
- *isDetail*
public boolean **isDetail**()
 - **Usage**
 - * Indicates whether is a main entity or is embedded in another
- *on_entitySelectionButton_clicked*
public void **on_entitySelectionButton_clicked**(Event e)
 - **Usage**
 - * Handler for the subentity selection buttons (magnifying glass buttons) clicked. Shows the selection dialog for selecting a valid instance of a subentity.
 - **Parameters**
 - * **e** - Event, magnifying glass button clicked event data.
- *printEntity*
public abstract OperationResult **printEntity**()
 - **Usage**
 - * Print entity button clicked handler. Builds the report with the data of the selected instance of the entity (selected registry), and allows the user to print the elaborated report.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.
- *printWidgets*
public void **printWidgets**()
- *queryEntity*
public abstract ArrayList **queryEntity**()
 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) available.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.
- *queryEntity*
public abstract ArrayList **queryEntity**(java.lang.String fieldId, java.lang.String value)
 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria.
 - **Parameters**

- * **fieldId** - String, name of the entity field that will be queried.
 - * **value** - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.

- *queryEntityWithFormat*

```
public abstract ArrayList queryEntityWithFormat( java.lang.String fieldId,
java.lang.String value )
```

 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria and show them (formatted as a list) to the user in the entity's screen.
 - **Parameters**
 - * **fieldId** - String, name of the entity field that will be queried.
 - * **value** - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList<RegistryOT>, list of entity instances (registries) taken as the result of the query operation.

- *querySubentities*

```
public abstract ArrayList querySubentities(
com.imagos.axilice.core.installer.SubentityOT entData )
```

 - **Usage**
 - * Queries the selected entity for all the detail subentities associated with it.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to query for.
 - **Returns** - ArrayList, list of detail subentity registries assigned to the master detail registry.

- *saveEntity*

```
public abstract OperationResult saveEntity( )
```

 - **Usage**
 - * Save entity button clicked handler. Update the system with the data of the edited entity registry. It's called for saving new entity registries (add a new registry) or for updating the contents of existing entity registries.
 - **Returns** - OperationResult, Result of the operation and list of messages generated through the operation.

- *saveEntityRegistry*

```
public void saveEntityRegistry( )
```
- *selectEntity*

```
public abstract void selectEntity( java.lang.Long idEntity )
```

 - **Usage**
 - * Marks an instance (registry) of the entity as the selected.
 - **Parameters**
 - * **idEntity** - Long, internal id of the selected instance (registry).

- *selectSubentity*

```
public void selectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseV0 subentityOT )
```

 - **Usage**
 - * Updates an entity instance by setting up the selected value for a subentity field.
 - **Parameters**
 - * **parentFieldId** - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.

-
- * subentityOT - BaseVO, subentity's selected instance (registry) value.
-
- *setEntity*
 public void **setEntity**(com.imagos.axilice.core.installer.EntityOT entity)
 – **Usage**
 * Sets up the entity name.
 – **Parameters**
 * **name** - String, entity name.
-
- *setRegistries*
 public void **setRegistries**(java.util.ArrayList lRegistries)
 – **Usage**
 * Sets the list of registries that are showed in the entity view controlled by the Entity Launcher instance. It's used for detail entities where queries are made in the master entity launcher.
 – **Parameters**
 * **lRegistries** - ArrayList<BaseVO>, list of shown registries of the entity.
-
- *showSelectionDialog*
 public void **showSelectionDialog**(com.imagos.axilice.core.installer.SubentityOT entData)
 – **Usage**
 * Opens the selection dialog that enables the user to select an instance of a subentity.
 – **Parameters**
 * **entData** - SubentityOT, data of the detail subentity to select.
-
- *showSubentityDialog*
 public void **showSubentityDialog**(com.imagos.axilice.core.installer.SubentityOT entData, long selEntity, boolean readonly)
 – **Usage**
 * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.
 – **Parameters**
 * **entData** - SubentityOT, data of the detail subentity to select.
 * **selEntity** - long, id of the selected instance of the subentity.
 * **readonly** - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.
-
- *showSubentityDialog*
 public void **showSubentityDialog**(com.imagos.axilice.core.installer.SubentityOT entData, java.lang.String parentGuiName, long selEntity, boolean readonly)
 – **Usage**
 * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.
 – **Parameters**
 * **entData** - SubentityOT, data of the detail subentity to select.
 * **parentGuiName** - String,
 * **selEntity** - long, id of the selected instance of the subentity.
 * **readonly** - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.
-
- *showSubentitySelectionDialog*
 public void **showSubentitySelectionDialog**(
 com.imagos.axilice.core.installer.SubentityOT entData)

- **Usage**
 - * Opens the selection dialog that enables the user to select an instance of a multiple selection detail subentity.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to select.
-
- *undoEntityEdition*
public void undoEntityEdition()
 - *updateEntityData*
public abstract void updateEntityData()
- **Usage**
 - * Updates the selected instance (registry) data with the content panel's data typed by the user.
-
- *viewEntity*
public abstract OperationResult viewEntity()
- **Usage**
 - * View entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), in read-only mode.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

17.1.3 CLASS CurrencyClient

DECLARATION

```
public class CurrencyClient
extends com.imagos.axilice.client.gui.common.CommonClient
```

CONSTRUCTORS

- *CurrencyClient*
public CurrencyClient()

METHODS

- *build_extension_fields_view*
public void build_extension_fields_view()
- *buildSelectionSets*
public void buildSelectionSets()
- *connectSubentityButtons*
public void connectSubentityButtons()
- *fillEntityData*
public void fillEntityData()

- *getEntityData*
public BaseVO **getEntityData**()
- *getWidgets*
public void **getWidgets**()
- *init_specific_components*
public void **init_specific_components**()
- *on_comboFlag_clicked*
public void **on_comboFlag_clicked**(Event e)
- *setEditionMode*
public void **setEditionMode**(boolean enabled)
- *setWidgetsValues*
public void **setWidgetsValues**(com.imagos.axilice.core.common.BaseVO base)
- *showFlagImage*
public void **showFlagImage**(com.imagos.axilice.core.common.FlagOT f)
- *validateEntity*
public OperationResult **validateEntity**()

METHODS INHERITED FROM CLASS com.imagos.axilice.client.gui.common.CommonClient

(in 17.1.2, page 402)

- *addAxiliceEventListener*
public void **addAxiliceEventListener**(
com.imagos.axilice.client.gui.AxiliceEventListener l)
- *addEntity*
public ArrayList **addEntity**()
- *addExistingSubentity*
public void **addExistingSubentity**(com.imagos.axilice.core.installer.SubentityOT
ent)
 - **Usage**
 - * Shows a dialog for selection of a new subentity
 - **Parameters**
 - * ent - SubentityOT. Encapsulate data type of subentity
- *addSubentity*
public void **addSubentity**(com.imagos.axilice.core.common.BaseVO subentity,
com.imagos.axilice.core.common.MultiSelectionSubentityOT subE)
 - **Usage**
 - * Adds a existent entity to a multiselection list
 - **Parameters**
 - * subentity - Subentity to be added
 - * subE - Object that encapsulates information about the subentity
- *addTrazabilityTree*
public void **addTrazabilityTree**(TabFolder tabFolder)
- *build_extension_fields_view*
public abstract void **build_extension_fields_view**()

– **Usage**

- * Build the view of the extension fields.

It must implement only those clients who require it

- *buildEntity*
public AxiliceEntityView buildEntity()
- *buildEntityDetail*
public AxiliceEntityView buildEntityDetail(
com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData)
- *buildEntitySelectionView*
public AxiliceEntitySelectionView buildEntitySelectionView()

– **Usage**

- * Build view of selection entities

– **Returns** - AxilicePartnerSelectionView. SelectionView created

- *buildSelectionSets*
public abstract void buildSelectionSets()

– **Usage**

- * Build 2 sets of SubentityOT: lSelectionSubentities and lMultiSelectionSubentities

These sets are used to connect the widgets with the handlers for the selection buttons and the multiselection panels

It must implement only those clients who require it

- *buildSubentityMultiSelection*
public void buildSubentityMultiSelection()
- *clearEntityData*
public void clearEntityData()
- *clearParentField*
public OperationResult clearParentField(java.util.ArrayList selRegistries,
java.lang.String parentField)

– **Usage**

- * Clears the value of the field indicated

– **Parameters**

- * selRegistries - ArrayList. Index list of records to process
- * parentField - Name of the field to clear

– **Returns** - OperationResult. Result of operation

- *delEntity*
public OperationResult delEntity()
- *delEntity*
public OperationResult delEntity(java.util.ArrayList selRegistries)
- *delEntity*
public OperationResult delEntity(com.imagos.axilice.core.common.BaseVO b)

– **Usage**

- * Delete a entity from database

– **Parameters**

- * b - BaseVO. Entity to delete

– **Returns** - OperationResult. Result of the operation

- *delSubentities*
public OperationResult delSubentities(java.util.ArrayList selRegistries)
 - **Usage**
 - * Delete a set of subentities from record list of controller, not from database
 - Set the parent field of subentity to null
 - **Parameters**
 - * selRegistries - List of index to delete
 - **Returns** - OperationResult. Result of the operation

- *editEntity*
public OperationResult editEntity()
- *fillEntityData*
public void fillEntityData()
- *fillEntityExtensionTable*
public void fillEntityExtensionTable(java.lang.Long id)
 - **Usage**
 - * Retrieve extension field values from the server for a given entity and displays them in the view
 - **Parameters**
 - * id - Long. Key of the entity whose values are to be displayed

- *fillEntityTable*
public void fillEntityTable()
 - **Usage**
 - * Load records from database and fill the entity view table and controller record list

- *fillEntityTable*
public void fillEntityTable(java.util.ArrayList lR)
 - **Usage**
 - * Fill the entity view table with the records given
 - **Parameters**
 - * lR - ArrayList. Records to be displayed

- *getClassEntity*
public Class getClassEntity()
 - **Returns** - Class of the entity for which the controller is built

- *getDialogName*
public String getDialogName()
- *getEditGladePath*
public String getEditGladePath()
- *getEditionGui*
public Gui getEditionGui()
 - **Usage**
 - * Returns the edition Gui instance.
 - **Returns** - Gui, entity edition Gui.

- *getEditWindow*
public SwtTopLevelWindow getEditWindow()

- *getEntityData*
public abstract BaseVO **getEntityData**()
- *getEntityFields*
public ArrayList **getEntityFields**()
- *getLogger*
public Logger **getLogger**()
- *getParentEntityLauncher*
public EntityLauncher **getParentEntityLauncher**()
 - **Returns** - EntityLauncher of parent entity if is a embedded entity or null otherwise
- *getRegistries*
public Collection **getRegistries**()
 - **Returns** - IRegistries
- *getRegistries*
public List **getRegistries**(java.util.ArrayList lIds)
 - **Usage**
 - * Get a list of entities stored by the controller
 - **Parameters**
 - * lIds - Index list
 - **Returns** - List of entities
- *getView*
public AxiliceEntityView **getView**()
 - **Returns** - AxiliceEntityView. Main view or Detail view if is a embedded entity.
- *getWidgets*
public abstract void **getWidgets**()
 - **Usage**
 - * Get the widgets from the glade and stored in local variables.
 - Must be overwritten
- *init_specific_components*
public abstract void **init_specific_components**()
 - **Usage**
 - * Initializes the specific components of the client
- *printEntity*
public OperationResult **printEntity**()
- *printEntity*
public OperationResult **printEntity**(java.lang.String reportPath, java.util.HashMap parameters)
- *queryEntity*
public ArrayList **queryEntity**()
- *queryEntity*
public ArrayList **queryEntity**(java.lang.String fieldId, java.lang.String value)
- *queryEntityWithFormat*
public ArrayList **queryEntityWithFormat**(java.lang.String fieldId, java.lang.String value)
- *queryFreeEntities*
public ArrayList **queryFreeEntities**(com.imagos.axilice.core.common.MultiSelectionSubentityOT mSubE)

- **Usage**
 - * Search entities not associated with a parent entity (null parent field)
 - **Parameters**
 - * mSubE - SubentityOT for the parent entity that must be null
 - **Returns** - ArrayList. List of entities found
-
- *querySubentities*
 public ArrayList **querySubentities**(com.imagos.axilice.core.installer.SubentityOT
 entData)
-
- *refreshEntity*
 public OperationResult **refreshEntity**()
-
- *refreshEntity*
 public BaseVO **refreshEntity**(com.imagos.axilice.core.common.BaseVO entity)
-
- *refreshParentSelectedEntity*
 public OperationResult **refreshParentSelectedEntity**()
- **Usage**
 - * Reload from database the value of the parent entity of the selected entity
 - **Returns** - OperationResult. The result of the operation.
-
- *removeRegistries*
 public OperationResult **removeRegistries**(java.util.ArrayList selRegistries,
 java.lang.String parentField)
- **Usage**
 - * Deletes a list of the records stored by the controller. Also clean your parent fields and updates the database
 - **Parameters**
 - * selRegistries - Index list to remove
 - * parentField - Field name that links to the parent entity
 - **Returns** - OperationResult. Result of the operation
-
- *saveEntity*
 public OperationResult **saveEntity**()
-
- *saveEntity*
 public OperationResult **saveEntity**(java.util.ArrayList lIds)
-
- *saveEntity*
 public OperationResult **saveEntity**(com.imagos.axilice.core.common.BaseVO b)
- **Usage**
 - * Save or update a entity to database
 - **Parameters**
 - * b - BaseVO. Entity to save or update
 - **Returns** - OperationResult. Result of the operation
-
- *selectEntity*
 public void **selectEntity**(java.lang.Long idEntity)
-
- *selectSubentity*
 public void **selectSubentity**(java.lang.String parentFieldId,
 com.imagos.axilice.core.common.BaseVO subentityOT)
-
- *selectTreeItem*
 public void **selectTreeItem**(java.lang.Long id)
- **Usage**
 - * Select a item in the entity tree. No me gusta esto aqui
 - **Parameters**

- * `id` - Long. Index of item to select
- - *setDialogName*
public void setDialogName(java.lang.String dialogName)
 - *setEditGladePath*
public void setEditGladePath(java.lang.String editGladePath)
 - *setEditionMode*
public abstract void setEditionMode(boolean enabled)
 - *setRegistries*
public void setRegistries(java.util.Collection collection)
 - **Usage**
 - * Setter for IRegistries

 - *setWidgetsValues*
public abstract void setWidgetsValues(com.imagos.axilice.core.common.BaseVO entity)
 - **Usage**
 - * Set the values of the widgets on the edit dialog.
 - Must be overwritten.
 - **Parameters**
 - * `entity` - BaseVO. Entity that contains the values to be displayed

 - *showErrorMessages*
public void showErrorMessages(com.imagos.axilice.core.util.OperationResult result)

 - *updateEntityData*
public void updateEntityData()
 - *updateRegistry*
public void updateRegistry(com.imagos.axilice.core.common.BaseVO ent)
 - **Usage**
 - * Add or update an entity (to the controller records)
 - **Parameters**
 - * `ent` - Entity to add or update

 - *updateViews*
public void updateViews(boolean updateFromParent)
 - **Usage**
 - * Force the update of the views associated
 - **Parameters**
 - * `updateFromParent` - Indicates whether the view must be updated by reading the values from the parent entity. takes effect only when the entity is detail (is embedded in another)

 - *validateEntity*
public abstract OperationResult validateEntity()
 - **Usage**
 - * Validate the values of the widgets in the edit dialog.
 - Must be overwritten.
 - **Returns** - OperationResult. Result of the validation.

 - *viewEntity*
public OperationResult viewEntity()

METHODS INHERITED FROM CLASS com.imagos.axilice.core.installer.EntityLauncher

(in 24.2.2, page 532)

- *addEntity*
 public abstract ArrayList **addEntity**()
 — **Usage**
 * Add entity button clicked handler. Show the entity content screen ready to start inserting data for a new instance of the entity (new registry).
 — **Returns** - ArrayList<ResultMessage>, list of errors if any has taken place, null in any other case.

- *buildEditEntityGUI*
 public void **buildEditEntityGUI**()
 — **Usage**
 * Builds the edit entity dialog window from the corresponding glade interface file. Stores the entity dialog instance for posterior accesses.

- *buildEntity*
 public AxiliceEntityView **buildEntity**()
 — **Usage**
 * Builds the entity GUI and show it on screen to the user.

- *buildEntity*
 public AxiliceDetailEntityView **buildEntity**(SwtWidget parent,
 com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData,
 java.lang.String rootName)
 — **Usage**
 * Builds the entity GUI and show it on screen to the user. The entity GUI will be embedded into the parent composite widget.
 — **Parameters**
 * **parent** - Composite, parent holder for embedding the entity view.
 * **subentityData** - SubentityOT, data of the subentity to build if the entity launcher will be used to handle a multi-selection detail entity view.
 * **rootName** - String, name of the root widget that will be embedded into the master entity edition panel.
 — **Returns** - AxiliceDetailEntityView, the entity detail view built.

- *cancelEntityEdition*
 public void **cancelEntityEdition**()

- *clearEntityData*
 public abstract void **clearEntityData**()
 — **Usage**
 * Clears the entity content panel by resetting the content widgets.

- *delEntity*
 public abstract OperationResult **delEntity**()
 — **Usage**
 * Delete entity button clicked handler. Deletes the selected instance of the entity (selected registry) from the database.
 — **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *delEntity*

```
public abstract OperationResult delEntity( java.util.ArrayList lSelRegistries )
```

- **Usage**

- * Delete entity button clicked handler. Deletes the selected instances of the entity (selected registries) from the database. Its a multiple delete method.

- **Parameters**

- * lSelRegistries - ArrayList<Long>, list of selected instances of the entity (selected registries). The list will contain only the internal id of the selected registries.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *deselectSubentity*

```
public void deselectSubentity( java.lang.String parentFieldId,  
com.imagos.axilice.core.common.BaseVO subentityOT )
```

- **Usage**

- * Updates an entity instance by removing the selected value for a subentity field.

- **Parameters**

- * parentFieldId - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.

- * subentityOT - BaseVO, subentity's deselected instance (registry) value.

- *editEntity*

```
public abstract OperationResult editEntity( )
```

- **Usage**

- * Edit entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), and allows the user to modify any content of the entity registry.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *fillEntityData*

```
public abstract void fillEntityData( )
```

- **Usage**

- * Fills the entity content panel with the data of the selected instance (registry) of the entity.

- *fireExceptionAndExit*

```
public void fireExceptionAndExit( java.lang.String msg )
```

- **Usage**

- * Fire a exception and stop the program

- **Parameters**

- * msg -

- *formatRegistries*

```
public ArrayList formatRegistries( )
```

- *formatRegistries*

```
public ArrayList formatRegistries( java.util.ArrayList lRegistries )
```

- *getDialog*

```
public SwtDialog getDialog( java.lang.String dialogName )
```

- *getEntity*

```
public EntityOT getEntity( )
```

- **Usage**

- * Gets the entity data.

- **Returns** - EntityOT, entity data.
- - *getEntityData*
 public abstract BaseVO **getEntityData**()
 - **Usage**
 - * Gets the content panel's data typed by the user formatted as an entity VO object.
- - *getEntityEditViewName*
 public String **getEntityEditViewName**(java.lang.String parentGuiName)
- - *getEntityFields*
 public abstract ArrayList **getEntityFields**()
 - **Usage**
 - * Gets the list of defined fields for an entity
 - **Returns** - ArrayList, list of entity fields.
- - *getEntityViewName*
 public String **getEntityViewName**(java.lang.String parentGuiName)
- - *getLRegistries*
 public ArrayList **getLRegistries**()
 - **Usage**
 - * Getter for lRegistries
- - *getRegistry*
 public BaseVO **getRegistry**(long id)
- - *getRevision*
 public static String **getRevision**()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
- - *getSelectedEntity*
 public BaseVO **getSelectedEntity**()
 - **Usage**
 - * Gets the selected instance (registry) of the entity.
 - **Returns** - BaseVO, selected instance (registry) of the entity.
- - *getSelectedId*
 public Long **getSelectedId**()
 - **Usage**
 - * Gets the internal id of the selected entity registry.
 - **Returns** - Long, internal id of the selected entity registry, null in case no entity registry has been selected yet.
- - *getSwtWidget*
 public SwtWidget **getSwtWidget**(java.lang.String widgetName)
- - *getWidget*
 public Widget **getWidget**(java.lang.String widgetName)
- - *initEntityLauncher*
 public void **initEntityLauncher**(Logger logger, Gui parentGui, SwtWidget parentPanel)
 - **Usage**
 - * Inits the entity launcher context and prepares the launcher to be started.
 - **Parameters**

- * **logger** - Logger, logger instance for printing logging messages.
- * **parentGuiName** - String, name of the JavaGlade instance that built the parent where the entity will be embedded.
- * **parentPanel** - Composite, parent panel where the entity contents will be embedded.

- *isDetail*

public boolean **isDetail**()

– **Usage**

- * Indicates whether is a main entity or is embedded in another

- *on_entitySelectionButton_clicked*

public void **on_entitySelectionButton_clicked**(Event e)

– **Usage**

- * Handler for the subentity selection buttons (magnifying glass buttons) clicked. Shows the selection dialog for selecting a valid instance of a subentity.

– **Parameters**

- * **e** - Event, magnifying glass button clicked event data.

- *printEntity*

public abstract **OperationResult printEntity**()

– **Usage**

- * Print entity button clicked handler. Builds the report with the data of the selected instance of the entity (selected registry), and allows the user to print the elaborated report.

- **Returns** - **OperationResult**, result of the delete operation and list of messages generated during the operation.

- *printWidgets*

public void **printWidgets**()

- *queryEntity*

public abstract **ArrayList queryEntity**()

– **Usage**

- * Search entity button clicked handler. Searches the system for the entity instances (registries) available.

- **Returns** - **ArrayList**, list of entity instances (registries) taken as the result of the query operation.

- *queryEntity*

public abstract **ArrayList queryEntity**(java.lang.String **fieldId**, java.lang.String **value**)

– **Usage**

- * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria.

– **Parameters**

- * **fieldId** - String, name of the entity field that will be queried.
- * **value** - String, value of the entity field that will be searched for.

- **Returns** - **ArrayList**, list of entity instances (registries) taken as the result of the query operation.

- *queryEntityWithFormat*

public abstract **ArrayList queryEntityWithFormat**(java.lang.String **fieldId**, java.lang.String **value**)

– **Usage**

- * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria and show them (formatted as a list) to the user in the entity's screen.
 - **Parameters**
 - * **fieldId** - String, name of the entity field that will be queried.
 - * **value** - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList<RegistryOT>, list of entity instances (registries) taken as the result of the query operation.

- *querySubentities*

```
public abstract ArrayList querySubentities(
com.imagos.axilice.core.installer.SubentityOT  entData )
```

 - **Usage**
 - * Queries the selected entity for all the detail subentities associated with it.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to query for.
 - **Returns** - ArrayList, list of detail subentity registries assigned to the master detail registry.

- *saveEntity*

```
public abstract OperationResult saveEntity( )
```

 - **Usage**
 - * Save entity button clicked handler. Update the system with the data of the edited entity registry. It's called for saving new entity registries (add a new registry) or for updating the contents of existing entity registries.
 - **Returns** - OperationResult, Result of the operation and list of messages generated through the operation.

- *saveEntityRegistry*

```
public void saveEntityRegistry( )
```
- *selectEntity*

```
public abstract void selectEntity( java.lang.Long  idEntity )
```

 - **Usage**
 - * Marks an instance (registry) of the entity as the selected.
 - **Parameters**
 - * **idEntity** - Long, internal id of the selected instance (registry).

- *selectSubentity*

```
public void selectSubentity( java.lang.String  parentFieldId,
com.imagos.axilice.core.common.BaseVO  subentityOT )
```

 - **Usage**
 - * Updates an entity instance by setting up the selected value for a subentity field.
 - **Parameters**
 - * **parentFieldId** - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * **subentityOT** - BaseVO, subentity's selected instance (registry) value.

- *setEntity*

```
public void setEntity( com.imagos.axilice.core.installer.EntityOT  entity )
```

 - **Usage**
 - * Sets up the entity name.
 - **Parameters**
 - * **name** - String, entity name.

- *setRegistries*

```
public void setRegistries( java.util.ArrayList IRegistries )
```

- Usage

- * Sets the list of registries that are showed in the entity view controlled by the Entity Launcher instance. It's used for detail entities where queries are made in the master entity launcher.

- Parameters

- * lRegistries - ArrayList<BaseVO>, list of shown registries of the entity.

- *showSelectionDialog*

```
public void showSelectionDialog( com.imagos.axilice.core.installer.SubentityOT entData )
```

- Usage

- * Opens the selection dialog that enables the user to select an instance of a subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.

- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT entData, long selEntity, boolean readonly )
```

- Usage

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.
 - * selEntity - long, id of the selected instance of the subentity.
 - * readonly - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT entData, java.lang.String parentGuiName, long selEntity, boolean readonly )
```

- Usage

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.
 - * parentGuiName - String,
 - * selEntity - long, id of the selected instance of the subentity.
 - * readonly - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

- *showSubentitySelectionDialog*

```
public void showSubentitySelectionDialog( com.imagos.axilice.core.installer.SubentityOT entData )
```

- Usage

- * Opens the selection dialog that enables the user to select an instance of a multiple selection detail subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.

- *undoEntityEdition*

```
public void undoEntityEdition( )
```

- *updateEntityData*
 public abstract void **updateEntityData**()
 – **Usage**
 * Updates the selected instance (registry) data with the content panel's data typed by the user.

- *viewEntity*
 public abstract **OperationResult** **viewEntity**()
 – **Usage**
 * View entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), in read-only mode.
 – **Returns** - **OperationResult**, result of the delete operation and list of messages generated during the operation.

17.1.4 CLASS MapViewer

DECLARATION

```
public class MapViewer
extends GenericHandler
```

CONSTRUCTORS

- *MapViewer*
 public **MapViewer**()

- *MapViewer*
 public **MapViewer**(com.imagos.axilice.core.partners.AddressOT address,
 com.imagos.axilice.client.gui.common.CommonClient entityLauncher, Logger
 log)
 – **Usage**
 * Default constructor
 – **Parameters**
 * address -
 * entityLauncher -

METHODS

- *build*
 public void **build**()

- *close*
 public void **close**(Event e)

- *getAddress*
 public AddressOT **getAddress**()

- *getWidgets*
public void **getWidgets**()
- *loadMap*
public **OperationResult** **loadMap**()
- *setAddress*
public void **setAddress**(com.imagos.axilice.core.partners.AddressOT address)
- *show*
public void **show**()

METHODS INHERITED FROM CLASS **GenericHandler**

17.1.5 CLASS **PriceListClient**

DECLARATION

```
public class PriceListClient
extends com.imagos.axilice.client.gui.common.CommonClient
```

CONSTRUCTORS

- *PriceListClient*
public **PriceListClient**()

METHODS

- *build_extension_fields_view*
public void **build_extension_fields_view**()
- *buildSelectionSets*
public void **buildSelectionSets**()
- *fillEntityData*
public void **fillEntityData**()
- *getEntityData*
public **BaseVO** **getEntityData**()
- *getWidgets*
public void **getWidgets**()
- *init_specific_components*
public void **init_specific_components**()

- *on_from_date_selected*
public void on_from_date_selected(Event e)
- *on_select_from_date_clicked*
public void on_select_from_date_clicked(Event e)
- *on_select_to_date_clicked*
public void on_select_to_date_clicked(Event e)
- *on_to_date_selected*
public void on_to_date_selected(Event e)
- *setEditionMode*
public void setEditionMode(boolean enabled)
- *setWidgetsValues*
public void setWidgetsValues(com.imagos.axilice.core.common.BaseVO entity)
- *validateEntity*
public OperationResult validateEntity()

METHODS INHERITED FROM CLASS com.imagos.axilice.client.gui.common.CommonClient

(in 17.1.2, page 402)

- *addAxiliceEventListener*
public void addAxiliceEventListener(com.imagos.axilice.client.gui.AxiliceEventListener l)
- *addEntity*
public ArrayList addEntity()
- *addExistingSubentity*
public void addExistingSubentity(com.imagos.axilice.core.installer.SubentityOT ent)
 - Usage
 - * Shows a dialog for selection of a new subentity
 - Parameters
 - * ent - SubentityOT. Encapsulate data type of subentity
- *addSubentity*
public void addSubentity(com.imagos.axilice.core.common.BaseVO subentity, com.imagos.axilice.core.common.MultiSelectionSubentityOT subE)
 - Usage
 - * Adds a existent entity to a multiselection list
 - Parameters
 - * subentity - Subentity to be added
 - * subE - Object that encapsulates information about the subentity
- *addTrazabilityTree*
public void addTrazabilityTree(TabFolder tabFolder)
- *build_extension_fields_view*
public abstract void build_extension_fields_view()
 - Usage

* Build the view of the extension fields.

It must implement only those clients who require it

- buildEntity
public AxiliceEntityView **buildEntity**()
- buildEntityDetail
public AxiliceEntityView **buildEntityDetail**(
com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData)
- buildEntitySelectionView
public AxiliceEntitySelectionView **buildEntitySelectionView**()
 - Usage
 - * Build view of selection entities
 - Returns - AxilicePartnerSelectionView. SelectionView created
- buildSelectionSets
public abstract void **buildSelectionSets**()
 - Usage
 - * Build 2 sets of SubentityOT: ISelectionSubentities and IMultiSelectionSubentities

These sets are used to connect the widgets with the handlers for the selection buttons and the multiseleccion panels

It must implement only those clients who require it

- buildSubentityMultiSelection
public void **buildSubentityMultiSelection**()
- clearEntityData
public void **clearEntityData**()
- clearParentField
public OperationResult **clearParentField**(java.util.ArrayList selRegistries,
java.lang.String parentField)
 - Usage
 - * Clears the value of the field indicated
 - Parameters
 - * selRegistries - ArrayList. Index list of records to process
 - * parentField - Name of the field to clear
 - Returns - OperationResult. Result of operation
- delEntity
public OperationResult **delEntity**()
- delEntity
public OperationResult **delEntity**(java.util.ArrayList selRegistries)
- delEntity
public OperationResult **delEntity**(com.imagos.axilice.core.common.BaseVO b)
 - Usage
 - * Delete a entity from database
 - Parameters
 - * b - BaseVO. Entity to delete
 - Returns - OperationResult. Result of the operation
- delSubentities
public OperationResult **delSubentities**(java.util.ArrayList selRegistries)

- **Usage**
 - * Delete a set of subentities from record list of controller, not from database
 - Set the parent field of subentity to null
 - **Parameters**
 - * `selRegistries` - List of index to delete
 - **Returns** - `OperationResult`. Result of the operation
-
- *editEntity*
public `OperationResult` **editEntity**()
 - *fillEntityData*
public void **fillEntityData**()
 - *fillEntityExtensionTable*
public void **fillEntityExtensionTable**(java.lang.Long id)
- **Usage**
 - * Retrieve extension field values from the server for a given entity and displays them in the view
 - **Parameters**
 - * `id` - Long. Key of the entity whose values are to be displayed
-
- *fillEntityTable*
public void **fillEntityTable**()
- **Usage**
 - * Load records from database and fill the entity view table and controller record list
-
- *fillEntityTable*
public void **fillEntityTable**(java.util.ArrayList lR)
- **Usage**
 - * Fill the entity view table with the records given
 - **Parameters**
 - * `lR` - ArrayList. Records to be displayed
-
- *getClassEntity*
public Class **getClassEntity**()
- **Returns** - Class of the entity for which the controller is built
-
- *getDialogName*
public String **getDialogName**()
 - *getEditGladePath*
public String **getEditGladePath**()
 - *getEditionGui*
public Gui **getEditionGui**()
- **Usage**
 - * Returns the edition Gui instance.
 - **Returns** - Gui, entity edition Gui.
-
- *getEditWindow*
public `SwtTopLevelWindow` **getEditWindow**()
 - *getEntityData*
public abstract BaseVO **getEntityData**()

- *getEntityFields*
public ArrayList **getEntityFields**()
- *getLogger*
public Logger **getLogger**()
- *getParentEntityLauncher*
public EntityLauncher **getParentEntityLauncher**()
 – **Returns** - EntityLauncher of parent entity if is a embedded entity or null otherwise
- *getRegistries*
public Collection **getRegistries**()
 – **Returns** - IRegistries
- *getRegistries*
public List **getRegistries**(java.util.ArrayList lIds)
 – **Usage**
 * Get a list of entities stored by the controller
 – **Parameters**
 * lIds - Index list
 – **Returns** - List of entities
- *getView*
public AxiliceEntityView **getView**()
 – **Returns** - AxiliceEntityView. Main view or Detail view if is a embedded entity.
- *getWidgets*
public abstract void **getWidgets**()
 – **Usage**
 * Get the widgets from the glade and stored in local variables.

 Must be overwritten
- *init_specific_components*
public abstract void **init_specific_components**()
 – **Usage**
 * Initializes the specific components of the client
- *printEntity*
public OperationResult **printEntity**()
- *printEntity*
public OperationResult **printEntity**(java.lang.String reportPath, java.util.HashMap parameters)
- *queryEntity*
public ArrayList **queryEntity**()
- *queryEntity*
public ArrayList **queryEntity**(java.lang.String fieldId, java.lang.String value)
- *queryEntityWithFormat*
public ArrayList **queryEntityWithFormat**(java.lang.String fieldId, java.lang.String value)
- *queryFreeEntities*
public ArrayList **queryFreeEntities**(com.imagos.axilice.core.common.MultiSelectionSubentityOT mSubE)
 – **Usage**

- * Search entities not associated with a parent entity (null parent field)
 - **Parameters**
 - * mSubE - SubentityOT for the parent entity that must be null
 - **Returns** - ArrayList. List of entities found

- *querySubentities*
public ArrayList querySubentities(com.imagos.axilice.core.installer.SubentityOT entData)

- *refreshEntity*
public OperationResult refreshEntity()

- *refreshEntity*
public BaseVO refreshEntity(com.imagos.axilice.core.common.BaseVO entity)

- *refreshParentSelectedEntity*
public OperationResult refreshParentSelectedEntity()
 - **Usage**
 - * Reload from database the value of the parent entity of the selected entity
 - **Returns** - OperationResult. The result of the operation.

- *removeRegistries*
public OperationResult removeRegistries(java.util.ArrayList selRegistries, java.lang.String parentField)
 - **Usage**
 - * Deletes a list of the records stored by the controller. Also clean your parent fields and updates the database
 - **Parameters**
 - * selRegistries - Index list to remove
 - * parentField - Field name that links to the parent entity
 - **Returns** - OperationResult. Result of the operation

- *saveEntity*
public OperationResult saveEntity()

- *saveEntity*
public OperationResult saveEntity(java.util.ArrayList lIds)

- *saveEntity*
public OperationResult saveEntity(com.imagos.axilice.core.common.BaseVO b)
 - **Usage**
 - * Save or update a entity to database
 - **Parameters**
 - * b - BaseVO. Entity to save or update
 - **Returns** - OperationResult. Result of the operation

- *selectEntity*
public void selectEntity(java.lang.Long idEntity)

- *selectSubentity*
public void selectSubentity(java.lang.String parentFieldId, com.imagos.axilice.core.common.BaseVO subentityOT)

- *selectTreeItem*
public void selectTreeItem(java.lang.Long id)
 - **Usage**
 - * Select a item in the entity tree. No me gusta esto aqui
 - **Parameters**
 - * id - Long. Index of item to select

-
- *setDialogName*
 public void **setDialogName**(java.lang.String dialogName)
 - *setEditGladePath*
 public void **setEditGladePath**(java.lang.String editGladePath)
 - *setEditionMode*
 public abstract void **setEditionMode**(boolean enabled)
 - *setRegistries*
 public void **setRegistries**(java.util.Collection collection)
 - **Usage**
 - * Setter for IRegistries
-
- *setWidgetsValues*
 public abstract void **setWidgetsValues**(com.imagos.axilice.core.common.BaseVO entity)
 - **Usage**
 - * Set the values of the widgets on the edit dialog.
 - Must be overwritten.
 - **Parameters**
 - * **entity** - BaseVO. Entity that contains the values to be displayed
-
- *showErrorMessages*
 public void **showErrorMessages**(com.imagos.axilice.core.util.OperationResult result)
-
- *updateEntityData*
 public void **updateEntityData**()
 - *updateRegistry*
 public void **updateRegistry**(com.imagos.axilice.core.common.BaseVO ent)
 - **Usage**
 - * Add or update an entity (to the controller records)
 - **Parameters**
 - * **ent** - Entity to add or update
-
- *updateViews*
 public void **updateViews**(boolean updateFromParent)
 - **Usage**
 - * Force the update of the views associated
 - **Parameters**
 - * **updateFromParent** - Indicates whether the view must be updated by reading the values from the parent entity. takes effect only when the entity is detail (is embedded in another)
-
- *validateEntity*
 public abstract OperationResult **validateEntity**()
 - **Usage**
 - * Validate the values of the widgets in the edit dialog.
 - Must be overwritten.
 - **Returns** - OperationResult. Result of the validation.
-
- *viewEntity*
 public OperationResult **viewEntity**()

METHODS INHERITED FROM CLASS com.imagos.axilice.core.installer.EntityLauncher

(in 24.2.2, page 532)

- *addEntity*
 public abstract ArrayList **addEntity**()
 — **Usage**
 * Add entity button clicked handler. Show the entity content screen ready to start inserting data for a new instance of the entity (new registry).
 — **Returns** - ArrayList<ResultMessage>, list of errors if any has taken place, null in any other case.

- *buildEditEntityGUI*
 public void **buildEditEntityGUI**()
 — **Usage**
 * Builds the edit entity dialog window from the corresponding glade interface file. Stores the entity dialog instance for posterior accesses.

- *buildEntity*
 public AxiliceEntityView **buildEntity**()
 — **Usage**
 * Builds the entity GUI and show it on screen to the user.

- *buildEntity*
 public AxiliceDetailEntityView **buildEntity**(SwtWidget parent,
 com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData,
 java.lang.String rootName)
 — **Usage**
 * Builds the entity GUI and show it on screen to the user. The entity GUI will be embedded into the parent composite widget.
 — **Parameters**
 * **parent** - Composite, parent holder for embedding the entity view.
 * **subentityData** - SubentityOT, data of the subentity to build if the entity launcher will be used to handle a multi-selection detail entity view.
 * **rootName** - String, name of the root widget that will be embedded into the master entity edition panel.
 — **Returns** - AxiliceDetailEntityView, the entity detail view built.

- *cancelEntityEdition*
 public void **cancelEntityEdition**()

- *clearEntityData*
 public abstract void **clearEntityData**()
 — **Usage**
 * Clears the entity content panel by resetting the content widgets.

- *delEntity*
 public abstract OperationResult **delEntity**()
 — **Usage**
 * Delete entity button clicked handler. Deletes the selected instance of the entity (selected registry) from the database.
 — **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *delEntity*

```
public abstract OperationResult delEntity( java.util.ArrayList lSelRegistries )
```

- **Usage**

- * Delete entity button clicked handler. Deletes the selected instances of the entity (selected registries) from the database. Its a multiple delete method.

- **Parameters**

- * lSelRegistries - ArrayList<Long>, list of selected instances of the entity (selected registries). The list will contain only the internal id of the selected registries.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *deselectSubentity*

```
public void deselectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

- **Usage**

- * Updates an entity instance by removing the selected value for a subentity field.

- **Parameters**

- * parentFieldId - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.

- * subentityOT - BaseVO, subentity's deselected instance (registry) value.

- *editEntity*

```
public abstract OperationResult editEntity( )
```

- **Usage**

- * Edit entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), and allows the user to modify any content of the entity registry.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *fillEntityData*

```
public abstract void fillEntityData( )
```

- **Usage**

- * Fills the entity content panel with the data of the selected instance (registry) of the entity.

- *fireExceptionAndExit*

```
public void fireExceptionAndExit( java.lang.String msg )
```

- **Usage**

- * Fire a exception and stop the program

- **Parameters**

- * msg -

- *formatRegistries*

```
public ArrayList formatRegistries( )
```

- *formatRegistries*

```
public ArrayList formatRegistries( java.util.ArrayList lRegistries )
```

- *getDialog*

```
public SwtDialog getDialog( java.lang.String dialogName )
```

- *getEntity*

```
public EntityOT getEntity( )
```

- **Usage**

- * Gets the entity data.

- **Returns** - EntityOT, entity data.
- - *getEntityData*
 public abstract BaseVO **getEntityData**()
 - **Usage**
 - * Gets the content panel's data typed by the user formatted as an entity VO object.
- - *getEntityEditViewName*
 public String **getEntityEditViewName**(java.lang.String parentGuiName)
- - *getEntityFields*
 public abstract ArrayList **getEntityFields**()
 - **Usage**
 - * Gets the list of defined fields for an entity
 - **Returns** - ArrayList, list of entity fields.
- - *getEntityViewName*
 public String **getEntityViewName**(java.lang.String parentGuiName)
- - *getLRegistries*
 public ArrayList **getLRegistries**()
 - **Usage**
 - * Getter for lRegistries
- - *getRegistry*
 public BaseVO **getRegistry**(long id)
- - *getRevision*
 public static String **getRevision**()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
- - *getSelectedEntity*
 public BaseVO **getSelectedEntity**()
 - **Usage**
 - * Gets the selected instance (registry) of the entity.
 - **Returns** - BaseVO, selected instance (registry) of the entity.
- - *getSelectedId*
 public Long **getSelectedId**()
 - **Usage**
 - * Gets the internal id of the selected entity registry.
 - **Returns** - Long, internal id of the selected entity registry, null in case no entity registry has been selected yet.
- - *getSwtWidget*
 public SwtWidget **getSwtWidget**(java.lang.String widgetName)
- - *getWidget*
 public Widget **getWidget**(java.lang.String widgetName)
- - *initEntityLauncher*
 public void **initEntityLauncher**(Logger logger, Gui parentGui, SwtWidget parentPanel)
 - **Usage**
 - * Inits the entity launcher context and prepares the launcher to be started.
 - **Parameters**

- * **logger** - Logger, logger instance for printing logging messages.
- * **parentGuiName** - String, name of the JavaGlade instance that built the parent where the entity will be embedded.
- * **parentPanel** - Composite, parent panel where the entity contents will be embedded.

- *isDetail*

public boolean **isDetail**()

– **Usage**

- * Indicates whether is a main entity or is embedded in another

- *on_entitySelectionButton_clicked*

public void **on_entitySelectionButton_clicked**(Event e)

– **Usage**

- * Handler for the subentity selection buttons (magnifying glass buttons) clicked. Shows the selection dialog for selecting a valid instance of a subentity.

– **Parameters**

- * **e** - Event, magnifying glass button clicked event data.

- *printEntity*

public abstract **OperationResult printEntity**()

– **Usage**

- * Print entity button clicked handler. Builds the report with the data of the selected instance of the entity (selected registry), and allows the user to print the elaborated report.

– **Returns** - **OperationResult**, result of the delete operation and list of messages generated during the operation.

- *printWidgets*

public void **printWidgets**()

- *queryEntity*

public abstract **ArrayList queryEntity**()

– **Usage**

- * Search entity button clicked handler. Searches the system for the entity instances (registries) available.

– **Returns** - **ArrayList**, list of entity instances (registries) taken as the result of the query operation.

- *queryEntity*

public abstract **ArrayList queryEntity**(java.lang.String **fieldId**, java.lang.String **value**)

– **Usage**

- * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria.

– **Parameters**

- * **fieldId** - String, name of the entity field that will be queried.
- * **value** - String, value of the entity field that will be searched for.

– **Returns** - **ArrayList**, list of entity instances (registries) taken as the result of the query operation.

- *queryEntityWithFormat*

public abstract **ArrayList queryEntityWithFormat**(java.lang.String **fieldId**, java.lang.String **value**)

– **Usage**

- * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria and show them (formatted as a list) to the user in the entity's screen.

– **Parameters**

- * **fieldId** - String, name of the entity field that will be queried.
- * **value** - String, value of the entity field that will be searched for.

- **Returns** - ArrayList<RegistryOT>, list of entity instances (registries) taken as the result of the query operation.

• *querySubentities*

```
public abstract ArrayList querySubentities(
com.imagos.axilice.core.installer.SubentityOT entData )
```

– **Usage**

- * Queries the selected entity for all the detail subentities associated with it.

– **Parameters**

- * **entData** - SubentityOT, data of the detail subentity to query for.

- **Returns** - ArrayList, list of detail subentity registries assigned to the master detail registry.

• *saveEntity*

```
public abstract OperationResult saveEntity( )
```

– **Usage**

- * Save entity button clicked handler. Update the system with the data of the edited entity registry. It's called for saving new entity registries (add a new registry) or for updating the contents of existing entity registries.

- **Returns** - OperationResult, Result of the operation and list of messages generated through the operation.

• *saveEntityRegistry*

```
public void saveEntityRegistry( )
```

• *selectEntity*

```
public abstract void selectEntity( java.lang.Long idEntity )
```

– **Usage**

- * Marks an instance (registry) of the entity as the selected.

– **Parameters**

- * **idEntity** - Long, internal id of the selected instance (registry).

• *selectSubentity*

```
public void selectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

– **Usage**

- * Updates an entity instance by setting up the selected value for a subentity field.

– **Parameters**

- * **parentFieldId** - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
- * **subentityOT** - BaseVO, subentity's selected instance (registry) value.

• *setEntity*

```
public void setEntity( com.imagos.axilice.core.installer.EntityOT entity )
```

– **Usage**

- * Sets up the entity name.

– **Parameters**

- * **name** - String, entity name.
-

- *setRegistries*

```
public void setRegistries( java.util.ArrayList lRegistries )
```

- Usage

- * Sets the list of registries that are showed in the entity view controlled by the Entity Launcher instance. It's used for detail entities where queries are made in the master entity launcher.

- Parameters

- * lRegistries - ArrayList<BaseVO>, list of shown registries of the entity.

- *showSelectionDialog*

```
public void showSelectionDialog( com.imagos.axilice.core.installer.SubentityOT entData )
```

- Usage

- * Opens the selection dialog that enables the user to select an instance of a subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.

- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT entData, long selEntity, boolean readonly )
```

- Usage

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.
 - * selEntity - long, id of the selected instance of the subentity.
 - * readonly - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT entData, java.lang.String parentGuiName, long selEntity, boolean readonly )
```

- Usage

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.
 - * parentGuiName - String,
 - * selEntity - long, id of the selected instance of the subentity.
 - * readonly - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

- *showSubentitySelectionDialog*

```
public void showSubentitySelectionDialog( com.imagos.axilice.core.installer.SubentityOT entData )
```

- Usage

- * Opens the selection dialog that enables the user to select an instance of a multiple selection detail subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.

- *undoEntityEdition*

```
public void undoEntityEdition( )
```

- *updateEntityData*
 public abstract void **updateEntityData**()
 – **Usage**
 * Updates the selected instance (registry) data with the content panel's data typed by the user.

- *viewEntity*
 public abstract **OperationResult viewEntity**()
 – **Usage**
 * View entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), in read-only mode.
 – **Returns** - **OperationResult**, result of the delete operation and list of messages generated during the operation.

17.1.6 CLASS PriceListLineClient

DECLARATION

```
public class PriceListLineClient
extends com.imagos.axilice.client.gui.common.CommonClient
```

CONSTRUCTORS

- *PriceListLineClient*
 public **PriceListLineClient**()

METHODS

- *build_extension_fields_view*
 public void **build_extension_fields_view**()

- *buildSelectionSets*
 public void **buildSelectionSets**()

- *clearEntityData*
 public void **clearEntityData**()

- *fillEntityData*
 public void **fillEntityData**()

- *getEntityData*
 public **BaseVO getEntityData**()

- *getWidgets*
 public void **getWidgets**()

- *init_specific_components*
 public void **init_specific_components**()

- *setEditionMode*
public void setEditionMode(boolean enabled)
- *setWidgetsValues*
public void setWidgetsValues(com.imagos.axilice.core.common.BaseVO entity)
- *validateEntity*
public **OperationResult** **validateEntity**()

METHODS INHERITED FROM CLASS `com.imagos.axilice.client.gui.common.CommonClient`

(in 17.1.2, page 402)

- *addAxiliceEventListener*
public void addAxiliceEventListener(com.imagos.axilice.client.gui.AxiliceEventListener l)
- *addEntity*
public ArrayList addEntity()
- *addExistingSubentity*
public void **addExistingSubentity**(com.imagos.axilice.core.installer.SubentityOT ent)

 - **Usage**
 - * Shows a dialog for selection of a new subentity
 - **Parameters**
 - * ent - SubentityOT. Encapsulate data type of subentity
- *addSubentity*
public void **addSubentity**(com.imagos.axilice.core.common.BaseVO subentity, com.imagos.axilice.core.common.MultiSelectionSubentityOT subE)

 - **Usage**
 - * Adds a existent entity to a multiselection list
 - **Parameters**
 - * subentity - Subentity to be added
 - * subE - Object that encapsulates information about the subentity
- *addTrazabilityTree*
public void addTrazabilityTree(TabFolder tabFolder)
- *build_extension_fields_view*
public abstract void **build_extension_fields_view**()

 - **Usage**
 - * Build the view of the extension fields.

It must implement only those clients who require it
- *buildEntity*
public AxiliceEntityView buildEntity()
- *buildEntityDetail*
public AxiliceEntityView buildEntityDetail(com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData)
- *buildEntitySelectionView*
public AxiliceEntitySelectionView buildEntitySelectionView()

 - **Usage**

- * Build view of selection entities
 - **Returns** - AxilicePartnerSelectionView. SelectionView created
- - *buildSelectionSets*
 public abstract void **buildSelectionSets**()
 - **Usage**
 - * Build 2 sets of SubentityOT: lSelectionSubentities and lMultiSelectionSubentities
 - These sets are used to connect the widgets with the handlers for the selection buttons and the multiseleccion panels

It must implement only those clients who require it
- - *buildSubentityMultiSelection*
 public void **buildSubentityMultiSelection**()
- - *clearEntityData*
 public void **clearEntityData**()
- - *clearParentField*
 public OperationResult **clearParentField**(java.util.ArrayList selRegistries, java.lang.String parentField)
 - **Usage**
 - * Clears the value of the field indicated
 - **Parameters**
 - * selRegistries - ArrayList. Index list of records to process
 - * parentField - Name of the field to clear
 - **Returns** - OperationResult. Result of operation
- - *delEntity*
 public OperationResult **delEntity**()
- - *delEntity*
 public OperationResult **delEntity**(java.util.ArrayList selRegistries)
- - *delEntity*
 public OperationResult **delEntity**(com.imagos.axilice.core.common.BaseVO b)
 - **Usage**
 - * Delete a entity from database
 - **Parameters**
 - * b - BaseVO. Entity to delete
 - **Returns** - OperationResult. Result of the operation
- - *delSubentities*
 public OperationResult **delSubentities**(java.util.ArrayList selRegistries)
 - **Usage**
 - * Delete a set of subentities from record list of controller, not from database

Set the parent field of subentity to null
 - **Parameters**
 - * selRegistries - List of index to delete
 - **Returns** - OperationResult. Result of the operation
- - *editEntity*
 public OperationResult **editEntity**()

- *fillEntityData*
public void fillEntityData()
- *fillEntityExtensionTable*
public void fillEntityExtensionTable(java.lang.Long id)
 - **Usage**
 - * Retrieve extension field values from the server for a given entity and displays them in the view
 - **Parameters**
 - * id - Long. Key of the entity whose values are to be displayed
- *fillEntityTable*
public void fillEntityTable()
 - **Usage**
 - * Load records from database and fill the entity view table and controller record list
- *fillEntityTable*
public void fillEntityTable(java.util.ArrayList lR)
 - **Usage**
 - * Fill the entity view table with the records given
 - **Parameters**
 - * lR - ArrayList. Records to be displayed
- *getClassEntity*
public Class getClassEntity()
 - **Returns** - Class of the entity for which the controller is built
- *getDialogName*
public String getDialogName()
- *getEditGladePath*
public String getEditGladePath()
- *getEditionGui*
public Gui getEditionGui()
 - **Usage**
 - * Returns the edition Gui instance.
 - **Returns** - Gui, entity edition Gui.
- *getEditWindow*
public SwtTopLevelWindow getEditWindow()
- *getEntityData*
public abstract BaseVO getEntityData()
- *getEntityFields*
public ArrayList getEntityFields()
- *getLogger*
public Logger getLogger()
- *getParentEntityLauncher*
public EntityLauncher getParentEntityLauncher()
 - **Returns** - EntityLauncher of parent entity if is a embedded entity or null otherwise
- *getRegistries*
public Collection getRegistries()
 - **Returns** - lRegistries

- *getRegistries*
 public List **getRegistries**(java.util.ArrayList lIds)
 – **Usage**
 * Get a list of entities stored by the controller
 – **Parameters**
 * lIds - Index list
 – **Returns** - List of entities

- *getView*
 public AxiliceEntityView **getView**()
 – **Returns** - AxiliceEntityView. Main view or Detail view if is a embedded entity.

- *getWidgets*
 public abstract void **getWidgets**()
 – **Usage**
 * Get the widgets from the glade and stored in local variables.

 Must be overwritten

- *init_specific_components*
 public abstract void **init_specific_components**()
 – **Usage**
 * Initializes the specific components of the client

- *printEntity*
 public OperationResult **printEntity**()

- *printEntity*
 public OperationResult **printEntity**(java.lang.String reportPath, java.util.HashMap parameters)

- *queryEntity*
 public ArrayList **queryEntity**()

- *queryEntity*
 public ArrayList **queryEntity**(java.lang.String fieldId, java.lang.String value)

- *queryEntityWithFormat*
 public ArrayList **queryEntityWithFormat**(java.lang.String fieldId, java.lang.String value)

- *queryFreeEntities*
 public ArrayList **queryFreeEntities**(com.imagos.axilice.core.common.MultiSelectionSubentityOT mSubE)
 – **Usage**
 * Search entities not associated with a parent entity (null parent field)
 – **Parameters**
 * mSubE - SubentityOT for the parent entity that must be null
 – **Returns** - ArrayList. List of entities found

- *querySubentities*
 public ArrayList **querySubentities**(com.imagos.axilice.core.installer.SubentityOT entData)

- *refreshEntity*
 public OperationResult **refreshEntity**()

- *refreshEntity*
 public BaseVO **refreshEntity**(com.imagos.axilice.core.common.BaseVO entity)

- *refreshParentSelectedEntity*
public OperationResult refreshParentSelectedEntity()
 - **Usage**
 - * Reload from database the value of the parent entity of the selected entity
 - **Returns** - OperationResult. The result of the operation.

- *removeRegistries*
public OperationResult removeRegistries(java.util.ArrayList selRegistries, java.lang.String parentField)
 - **Usage**
 - * Deletes a list of the records stored by the controller. Also clean your parent fields and updates the database
 - **Parameters**
 - * **selRegistries** - Index list to remove
 - * **parentField** - Field name that links to the parent entity
 - **Returns** - OperationResult. Result of the operation

- *saveEntity*
public OperationResult saveEntity()

- *saveEntity*
public OperationResult saveEntity(java.util.ArrayList lIds)

- *saveEntity*
public OperationResult saveEntity(com.imagos.axilice.core.common.BaseVO b)
 - **Usage**
 - * Save or update a entity to database
 - **Parameters**
 - * **b** - BaseVO. Entity to save or update
 - **Returns** - OperationResult. Result of the operation

- *selectEntity*
public void selectEntity(java.lang.Long idEntity)

- *selectSubentity*
public void selectSubentity(java.lang.String parentFieldId, com.imagos.axilice.core.common.BaseVO subentityOT)

- *selectTreeItem*
public void selectTreeItem(java.lang.Long id)
 - **Usage**
 - * Select a item in the entity tree. No me gusta esto aqui
 - **Parameters**
 - * **id** - Long. Index of item to select

- *setDialogName*
public void setDialogName(java.lang.String dialogName)

- *setEditGladePath*
public void setEditGladePath(java.lang.String editGladePath)

- *setEditionMode*
public abstract void setEditionMode(boolean enabled)

- *setRegistries*
public void setRegistries(java.util.Collection collection)
 - **Usage**
 - * Setter for lRegistries

- *setWidgetsValues*
 public abstract void **setWidgetsValues**(com.imagos.axilice.core.common.BaseVO
 entity)
 – **Usage**
 * Set the values of the widgets on the edit dialog.
 Must be overwritten.
 – **Parameters**
 * **entity** - BaseVO. Entity that contains the values to be displayed

- *showErrorMessages*
 public void **showErrorMessages**(com.imagos.axilice.core.util.OperationResult result
)

- *updateEntityData*
 public void **updateEntityData**()

- *updateRegistry*
 public void **updateRegistry**(com.imagos.axilice.core.common.BaseVO ent)
 – **Usage**
 * Add or update an entity (to the controller records)
 – **Parameters**
 * **ent** - Entity to add or update

- *updateViews*
 public void **updateViews**(boolean updateFromParent)
 – **Usage**
 * Force the update of the views associated
 – **Parameters**
 * **updateFromParent** - Indicates whether the view must be updated by reading the values
 from the parent entity. takes effect only when the entity is detail (is embedded in another)

- *validateEntity*
 public abstract OperationResult **validateEntity**()
 – **Usage**
 * Validate the values of the widgets in the edit dialog.
 Must be overwritten.
 – **Returns** - OperationResult. Result of the validation.

- *viewEntity*
 public OperationResult **viewEntity**()

METHODS INHERITED FROM CLASS com.imagos.axilice.core.installer.EntityLauncher

(in 24.2.2, page 532)

- *addEntity*
 public abstract ArrayList **addEntity**()
 – **Usage**
 * Add entity button clicked handler. Show the entity content screen ready to start inserting
 data for a new instance of the entity (new registry).
 – **Returns** - ArrayList<ResultMessage>, list of errors if any has taken place, null in any other
 case.

-
- *buildEditEntityGUI*
`public void buildEditEntityGUI()`
 - **Usage**
 - * Builds the edit entity dialog window from the corresponding glade interface file. Stores the entity dialog instance for posterior accesses.
-
- *buildEntity*
`public AxiliceEntityView buildEntity()`
 - **Usage**
 - * Builds the entity GUI and show it on screen to the user.
-
- *buildEntity*
`public AxiliceDetailEntityView buildEntity(SwtWidget parent,
com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData,
java.lang.String rootName)`
 - **Usage**
 - * Builds the entity GUI and show it on screen to the user. The entity GUI will be embedded into the parent composite widget.
 - **Parameters**
 - * **parent** - Composite, parent holder for embedding the entity view.
 - * **subentityData** - SubentityOT, data of the subentity to build if the entity launcher will be used to handle a multi-selection detail entity view.
 - * **rootName** - String, name of the root widget that will be embedded into the master entity edition panel.
 - **Returns** - AxiliceDetailEntityView, the entity detail view built.
-
- *cancelEntityEdition*
`public void cancelEntityEdition()`
-
- *clearEntityData*
`public abstract void clearEntityData()`
 - **Usage**
 - * Clears the entity content panel by resetting the content widgets.
-
- *delEntity*
`public abstract OperationResult delEntity()`
 - **Usage**
 - * Delete entity button clicked handler. Deletes the selected instance of the entity (selected registry) from the database.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.
-
- *delEntity*
`public abstract OperationResult delEntity(java.util.ArrayList lSelRegistries)`
 - **Usage**
 - * Delete entity button clicked handler. Deletes the selected instances of the entity (selected registries) from the database. Its a multiple delete method.
 - **Parameters**
 - * **lSelRegistries** - ArrayList<Long>, list of selected instances of the entity (selected registries). The list will contain only the internal id of the selected registries.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.
-

- *deselectSubentity*

```
public void deselectSubentity( java.lang.String  parentFieldId,
com.imagos.axilice.core.common.BaseVO  subentityOT )
```

- **Usage**

- * Updates an entity instance by removing the selected value for a subentity field.

- **Parameters**

- * **parentFieldId** - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * **subentityOT** - BaseVO, subentity's deselected instance (registry) value.

- *editEntity*

```
public abstract OperationResult editEntity( )
```

- **Usage**

- * Edit entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), and allows the user to modify any content of the entity registry.

- **Returns** - **OperationResult**, result of the delete operation and list of messages generated during the operation.

- *fillEntityData*

```
public abstract void fillEntityData( )
```

- **Usage**

- * Fills the entity content panel with the data of the selected instance (registry) of the entity.

- *fireExceptionAndExit*

```
public void fireExceptionAndExit( java.lang.String  msg )
```

- **Usage**

- * Fire a exception and stop the program

- **Parameters**

- * **msg** -

- *formatRegistries*

```
public ArrayList formatRegistries( )
```

- *formatRegistries*

```
public ArrayList formatRegistries( java.util.ArrayList  IRegistries )
```

- *getDialog*

```
public SwtDialog getDialog( java.lang.String  dialogName )
```

- *getEntity*

```
public EntityOT getEntity( )
```

- **Usage**

- * Gets the entity data.

- **Returns** - **EntityOT**, entity data.

- *getEntityData*

```
public abstract BaseVO getEntityData( )
```

- **Usage**

- * Gets the content panel's data typed by the user formatted as an entity VO object.

- *getEntityEditViewName*

```
public String getEntityEditViewName( java.lang.String  parentGuiName )
```

- *getEntityFields*

```
public abstract ArrayList getEntityFields( )
```

- **Usage**
 - * Gets the list of defined fields for an entity
 - **Returns** - ArrayList, list of entity fields.
- - *getEntityViewName*
public String getEntityViewName(java.lang.String parentGuiName)
 - *getLRegistries*
public ArrayList getLRegistries()
 - **Usage**
 - * Getter for lRegistries
- - *getRegistry*
public BaseVO getRegistry(long id)
 - *getRevision*
public static String getRevision()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
- - *getSelectedEntity*
public BaseVO getSelectedEntity()
 - **Usage**
 - * Gets the selected instance (registry) of the entity.
 - **Returns** - BaseVO, selected instance (registry) of the entity.
- - *getSelectedId*
public Long getSelectedId()
 - **Usage**
 - * Gets the internal id of the selected entity registry.
 - **Returns** - Long, internal id of the selected entity registry, null in case no entity registry has been selected yet.
- - *getSwtWidget*
public SwtWidget getSwtWidget(java.lang.String widgetName)
 - *getWidget*
public Widget getWidget(java.lang.String widgetName)
 - *initEntityLauncher*
public void initEntityLauncher(Logger logger, Gui parentGui, SwtWidget parentPanel)
 - **Usage**
 - * Inits the entity launcher context and prepares the launcher to be started.
 - **Parameters**
 - * **logger** - Logger, logger instance for printing logging messages.
 - * **parentGuiName** - String, name of the JavaGlade instance that built the parent where the entity will be embedded.
 - * **parentPanel** - Composite, parent panel where the entity contents will be embedded.
- - *isDetail*
public boolean isDetail()
 - **Usage**
 - * Indicates whether is a main entity or is embedded in another
- - *on_entitySelectionButton_clicked*
public void on_entitySelectionButton_clicked(Event e)

- **Usage**
 - * Handler for the subentity selection buttons (magnifying glass buttons) clicked. Shows the selection dialog for selecting a valid instance of a subentity.
 - **Parameters**
 - * e - Event, magnifying glass button clicked event data.
-
- *printEntity*
 public abstract OperationResult **printEntity**()
 - **Usage**
 - * Print entity button clicked handler. Builds the report with the data of the selected instance of the entity (selected registry), and allows the user to print the elaborated report.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.
-
- *printWidgets*
 public void **printWidgets**()
-
- *queryEntity*
 public abstract ArrayList **queryEntity**()
 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) available.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.
-
- *queryEntity*
 public abstract ArrayList **queryEntity**(java.lang.String fieldId, java.lang.String value)
 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria.
 - **Parameters**
 - * fieldId - String, name of the entity field that will be queried.
 - * value - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.
-
- *queryEntityWithFormat*
 public abstract ArrayList **queryEntityWithFormat**(java.lang.String fieldId, java.lang.String value)
 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria and show them (formatted as a list) to the user in the entity's screen.
 - **Parameters**
 - * fieldId - String, name of the entity field that will be queried.
 - * value - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList<RegistryOT>, list of entity instances (registries) taken as the result of the query operation.
-
- *querySubentities*
 public abstract ArrayList **querySubentities**(com.imagos.axilice.core.installer.SubentityOT entData)
 - **Usage**

- * Queries the selected entity for all the detail subentities associated with it.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to query for.
 - **Returns** - ArrayList, list of detail subentity registries assigned to the master detail registry.

- *saveEntity*

```
public abstract OperationResult saveEntity( )
```

 - **Usage**
 - * Save entity button clicked handler. Update the system with the data of the edited entity registry. It's called for saving new entity registries (add a new registry) or for updating the contents of existing entity registries.
 - **Returns** - OperationResult, Result of the operation and list of messages generated through the operation.

- *saveEntityRegistry*

```
public void saveEntityRegistry( )
```
- *selectEntity*

```
public abstract void selectEntity( java.lang.Long idEntity )
```

 - **Usage**
 - * Marks an instance (registry) of the entity as the selected.
 - **Parameters**
 - * **idEntity** - Long, internal id of the selected instance (registry).

- *selectSubentity*

```
public void selectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

 - **Usage**
 - * Updates an entity instance by setting up the selected value for a subentity field.
 - **Parameters**
 - * **parentFieldId** - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * **subentityOT** - BaseVO, subentity's selected instance (registry) value.

- *setEntity*

```
public void setEntity( com.imagos.axilice.core.installer.EntityOT entity )
```

 - **Usage**
 - * Sets up the entity name.
 - **Parameters**
 - * **name** - String, entity name.

- *setRegistries*

```
public void setRegistries( java.util.ArrayList lRegistries )
```

 - **Usage**
 - * Sets the list of registries that are showed in the entity view controlled by the Entity Launcher instance. It's used for detail entities where queries are made in the master entity launcher.
 - **Parameters**
 - * **lRegistries** - ArrayList<BaseVO>, list of shown registries of the entity.

- *showSelectionDialog*

```
public void showSelectionDialog( com.imagos.axilice.core.installer.SubentityOT
entData )
```

 - **Usage**

* Opens the selection dialog that enables the user to select an instance of a subentity.

– **Parameters**

* **entData** - SubentityOT, data of the detail subentity to select.

• *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT
entData, long selEntity, boolean readonly )
```

– **Usage**

* Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

– **Parameters**

* **entData** - SubentityOT, data of the detail subentity to select.

* **selEntity** - long, id of the selected instance of the subentity.

* **readonly** - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

• *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT
entData, java.lang.String parentGuiName, long selEntity, boolean readonly )
```

– **Usage**

* Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

– **Parameters**

* **entData** - SubentityOT, data of the detail subentity to select.

* **parentGuiName** - String,

* **selEntity** - long, id of the selected instance of the subentity.

* **readonly** - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

• *showSubentitySelectionDialog*

```
public void showSubentitySelectionDialog(
com.imagos.axilice.core.installer.SubentityOT entData )
```

– **Usage**

* Opens the selection dialog that enables the user to select an instance of a multiple selection detail subentity.

– **Parameters**

* **entData** - SubentityOT, data of the detail subentity to select.

• *undoEntityEdition*

```
public void undoEntityEdition( )
```

• *updateEntityData*

```
public abstract void updateEntityData( )
```

– **Usage**

* Updates the selected instance (registry) data with the content panel's data typed by the user.

• *viewEntity*

```
public abstract OperationResult viewEntity( )
```

– **Usage**

* View entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), in read-only mode.

– **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

17.1.7 CLASS VatTypeClient

DECLARATION

```
public class VatTypeClient
extends com.imagos.axilice.core.installer.EntityLauncher
```

FIELDS

- public static final String AXILICE_LOG_EDITVATTYPE_GLADEPATH
– Message for logging the entity glade configuration file path.

CONSTRUCTORS

- *VatTypeClient*
public **VatTypeClient**()

METHODS

- *addEntity*
public ArrayList **addEntity**()
- *clearEntityData*
public void **clearEntityData**()
- *delEntity*
public OperationResult **delEntity**()
- *delEntity*
public OperationResult **delEntity**(java.util.ArrayList lSelRegistries)
- *editEntity*
public OperationResult **editEntity**()
- *fillEntityData*
public void **fillEntityData**()
- *getEntityData*
public BaseVO **getEntityData**()
- *getEntityFields*
public ArrayList **getEntityFields**()
- *getSelectedEntity*
public BaseVO **getSelectedEntity**()
- *getSelectedId*
public Long **getSelectedId**()

- *printEntity*
public OperationResult **printEntity**()
- *queryEntity*
public ArrayList **queryEntity**()
- *queryEntity*
public ArrayList **queryEntity**(java.lang.String fieldId, java.lang.String value)
- *queryEntityWithFormat*
public ArrayList **queryEntityWithFormat**(java.lang.String fieldId, java.lang.String value)
- *querySubentities*
public ArrayList **querySubentities**(com.imagos.axilice.core.installer.SubentityOT entData)
 - **Usage**
 - * Queries the selected entity for all the detail subentities associated with it.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to query for.
 - **Returns** - ArrayList, list of detail subentity registries assigned to the master detail registry.
- *saveEntity*
public OperationResult **saveEntity**()
- *selectEntity*
public void **selectEntity**(java.lang.Long idEntity)
- *selectSubentity*
public void **selectSubentity**(java.lang.String parentFieldId, com.imagos.axilice.core.common.BaseVO subentityOT)
- *updateEntityData*
public void **updateEntityData**()
- *viewEntity*
public OperationResult **viewEntity**()

METHODS INHERITED FROM CLASS com.imagos.axilice.core.installer.EntityLauncher

(in 24.2.2, page 532)

- *addEntity*
public abstract ArrayList **addEntity**()
 - **Usage**
 - * Add entity button clicked handler. Show the entity content screen ready to start inserting data for a new instance of the entity (new registry).
 - **Returns** - ArrayList<ResultMessage>, list of errors if any has taken place, null in any other case.
- *buildEditEntityGUI*
public void **buildEditEntityGUI**()

- **Usage**
 - * Builds the edit entity dialog window from the corresponding glade interface file. Stores the entity dialog instance for posterior accesses.

- *buildEntity*
public AxiliceEntityView buildEntity()
 - **Usage**
 - * Builds the entity GUI and show it on screen to the user.

- *buildEntity*
**public AxiliceDetailEntityView buildEntity(SwtWidget parent,
com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData,
java.lang.String rootName)**
 - **Usage**
 - * Builds the entity GUI and show it on screen to the user. The entity GUI will be embedded into the parent composite widget.
 - **Parameters**
 - * **parent** - Composite, parent holder for embedding the entity view.
 - * **subentityData** - SubentityOT, data of the subentity to build if the entity launcher will be used to handle a multi-selection detail entity view.
 - * **rootName** - String, name of the root widget that will be embedded into the master entity edition panel.
 - **Returns** - AxiliceDetailEntityView, the entity detail view built.

- *cancelEntityEdition*
public void cancelEntityEdition()

- *clearEntityData*
public abstract void clearEntityData()
 - **Usage**
 - * Clears the entity content panel by resetting the content widgets.

- *delEntity*
public abstract OperationResult delEntity()
 - **Usage**
 - * Delete entity button clicked handler. Deletes the selected instance of the entity (selected registry) from the database.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *delEntity*
public abstract OperationResult delEntity(java.util.ArrayList lSelRegistries)
 - **Usage**
 - * Delete entity button clicked handler. Deletes the selected instances of the entity (selected registries) from the database. Its a multiple delete method.
 - **Parameters**
 - * **lSelRegistries** - ArrayList<Long>, list of selected instances of the entity (selected registries). The list will contain only the internal id of the selected registries.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *deselectSubentity*
**public void deselectSubentity(java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseV0 subentityOT)**

- **Usage**
 - * Updates an entity instance by removing the selected value for a subentity field.
- **Parameters**
 - * **parentFieldId** - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * **subentityOT** - BaseVO, subentity's deselected instance (registry) value.

- *editEntity*
public abstract OperationResult editEntity()
 - **Usage**
 - * Edit entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), and allows the user to modify any content of the entity registry.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *fillEntityData*
public abstract void fillEntityData()
 - **Usage**
 - * Fills the entity content panel with the data of the selected instance (registry) of the entity.

- *fireExceptionAndExit*
public void fireExceptionAndExit(java.lang.String msg)
 - **Usage**
 - * Fire a exception and stop the program
 - **Parameters**
 - * **msg** -

- *formatRegistries*
public ArrayList formatRegistries()

- *formatRegistries*
public ArrayList formatRegistries(java.util.ArrayList IRegistries)

- *getDialog*
public SwtDialog getDialog(java.lang.String dialogName)

- *getEntity*
public EntityOT getEntity()
 - **Usage**
 - * Gets the entity data.
 - **Returns** - EntityOT, entity data.

- *getEntityData*
public abstract BaseVO getEntityData()
 - **Usage**
 - * Gets the content panel's data typed by the user formatted as an entity VO object.

- *getEntityEditViewName*
public String getEntityEditViewName(java.lang.String parentGuiName)

- *getEntityFields*
public abstract ArrayList getEntityFields()
 - **Usage**
 - * Gets the list of defined fields for an entity
 - **Returns** - ArrayList, list of entity fields.

- *getEntityViewName*
public String **getEntityViewName**(java.lang.String parentGuiName)
- *getLRegistries*
public ArrayList **getLRegistries**()
 - **Usage**
 - * Getter for IRegistries
- *getRegistry*
public BaseVO **getRegistry**(long id)
- *getRevision*
public static String **getRevision**()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
- *getSelectedEntity*
public BaseVO **getSelectedEntity**()
 - **Usage**
 - * Gets the selected instance (registry) of the entity.
 - **Returns** - BaseVO, selected instance (registry) of the entity.
- *getSelectedId*
public Long **getSelectedId**()
 - **Usage**
 - * Gets the internal id of the selected entity registry.
 - **Returns** - Long, internal id of the selected entity registry, null in case no entity registry has been selected yet.
- *getSwtWidget*
public SwtWidget **getSwtWidget**(java.lang.String widgetName)
- *getWidget*
public Widget **getWidget**(java.lang.String widgetName)
- *initEntityLauncher*
public void **initEntityLauncher**(Logger logger, Gui parentGui, SwtWidget parentPanel)
 - **Usage**
 - * Inits the entity launcher context and prepares the launcher to be started.
 - **Parameters**
 - * **logger** - Logger, logger instance for printing logging messages.
 - * **parentGuiName** - String, name of the JavaGlade instance that built the parent where the entity will be embedded.
 - * **parentPanel** - Composite, parent panel where the entity contents will be embedded.
- *isDetail*
public boolean **isDetail**()
 - **Usage**
 - * Indicates whether is a main entity or is embedded in another
- *on_entitySelectionButton_clicked*
public void **on_entitySelectionButton_clicked**(Event e)
 - **Usage**
 - * Handler for the subentity selection buttons (magnifying glass buttons) clicked. Shows the selection dialog for selecting a valid instance of a subentity.

- **Parameters**
 - * *e* - Event, magnifying glass button clicked event data.

- *printEntity*

```
public abstract OperationResult printEntity( )
```

 - **Usage**
 - * Print entity button clicked handler. Builds the report with the data of the selected instance of the entity (selected registry), and allows the user to print the elaborated report.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *printWidgets*

```
public void printWidgets( )
```

- *queryEntity*

```
public abstract ArrayList queryEntity( )
```

 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) available.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.

- *queryEntity*

```
public abstract ArrayList queryEntity( java.lang.String fieldId, java.lang.String value )
```

 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria.
 - **Parameters**
 - * *fieldId* - String, name of the entity field that will be queried.
 - * *value* - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.

- *queryEntityWithFormat*

```
public abstract ArrayList queryEntityWithFormat( java.lang.String fieldId, java.lang.String value )
```

 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria and show them (formatted as a list) to the user in the entity's screen.
 - **Parameters**
 - * *fieldId* - String, name of the entity field that will be queried.
 - * *value* - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList<RegistryOT>, list of entity instances (registries) taken as the result of the query operation.

- *querySubentities*

```
public abstract ArrayList querySubentities( com.imagos.axilice.core.installer.SubentityOT entData )
```

 - **Usage**
 - * Queries the selected entity for all the detail subentities associated with it.
 - **Parameters**
 - * *entData* - SubentityOT, data of the detail subentity to query for.

- **Returns** - ArrayList, list of detail subentity registries assigned to the master detail registry.

- *saveEntity*

```
public abstract OperationResult saveEntity( )
```

 - **Usage**
 - * Save entity button clicked handler. Update the system with the data of the edited entity registry. It's called for saving new entity registries (add a new registry) or for updating the contents of existing entity registries.
 - **Returns** - OperationResult, Result of the operation and list of messages generated through the operation.

- *saveEntityRegistry*

```
public void saveEntityRegistry( )
```
- *selectEntity*

```
public abstract void selectEntity( java.lang.Long idEntity )
```

 - **Usage**
 - * Marks an instance (registry) of the entity as the selected.
 - **Parameters**
 - * idEntity - Long, internal id of the selected instance (registry).

- *selectSubentity*

```
public void selectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

 - **Usage**
 - * Updates an entity instance by setting up the selected value for a subentity field.
 - **Parameters**
 - * parentFieldId - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * subentityOT - BaseVO, subentity's selected instance (registry) value.

- *setEntity*

```
public void setEntity( com.imagos.axilice.core.installer.EntityOT entity )
```

 - **Usage**
 - * Sets up the entity name.
 - **Parameters**
 - * name - String, entity name.

- *setRegistries*

```
public void setRegistries( java.util.ArrayList lRegistries )
```

 - **Usage**
 - * Sets the list of registries that are showed in the entity view controlled by the Entity Launcher instance. It's used for detail entities where queries are made in the master entity launcher.
 - **Parameters**
 - * lRegistries - ArrayList<BaseVO>, list of shown registries of the entity.

- *showSelectionDialog*

```
public void showSelectionDialog( com.imagos.axilice.core.installer.SubentityOT
entData )
```

 - **Usage**
 - * Opens the selection dialog that enables the user to select an instance of a subentity.
 - **Parameters**
 - * entData - SubentityOT, data of the detail subentity to select.

- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT
entData, long selEntity, boolean readonly )
```

- Usage

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

- Parameters

- * **entData** - SubentityOT, data of the detail subentity to select.
 - * **selEntity** - long, id of the selected instance of the subentity.
 - * **readonly** - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT
entData, java.lang.String parentGuiName, long selEntity, boolean readonly )
```

- Usage

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

- Parameters

- * **entData** - SubentityOT, data of the detail subentity to select.
 - * **parentGuiName** - String,
 - * **selEntity** - long, id of the selected instance of the subentity.
 - * **readonly** - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

- *showSubentitySelectionDialog*

```
public void showSubentitySelectionDialog(
com.imagos.axilice.core.installer.SubentityOT entData )
```

- Usage

- * Opens the selection dialog that enables the user to select an instance of a multiple selection detail subentity.

- Parameters

- * **entData** - SubentityOT, data of the detail subentity to select.

- *undoEntityEdition*

```
public void undoEntityEdition( )
```

- *updateEntityData*

```
public abstract void updateEntityData( )
```

- Usage

- * Updates the selected instance (registry) data with the content panel's data typed by the user.

- *viewEntity*

```
public abstract OperationResult viewEntity( )
```

- Usage

- * View entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), in read-only mode.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

Chapter 18

Package com.imagos.axilice.core.ssl

<i>Package Contents</i>	<i>Page</i>
<hr/>	
Classes	
RMISSLClientSocketFactory 464	
<i>...no description...</i>	
RMISSLServerSocketFactory 464	
<i>...no description...</i>	
<hr/>	

18.1 Classes

18.1.1 CLASS *RMISSLClientSocketFactory*

DECLARATION

```
public class RMISSLClientSocketFactory
extends java.lang.Object
implements java.rmi.server.RMIClientSocketFactory, java.io.Serializable
```

CONSTRUCTORS

- *RMISSLClientSocketFactory*
public **RMISSLClientSocketFactory**()

METHODS

- *createSocket*
public Socket **createSocket**(java.lang.String host, int port)

18.1.2 CLASS *RMISSLServerSocketFactory*

DECLARATION

```
public class RMISSLServerSocketFactory
extends java.lang.Object
implements java.rmi.server.RMIServerSocketFactory, java.io.Serializable
```

CONSTRUCTORS

- *RMISSLServerSocketFactory*
public **RMISSLServerSocketFactory**()

METHODS

- *createServerSocket*
public ServerSocket **createServerSocket**(int port)

Chapter 19

Package com.imagos.axilice.server.common.bl

<i>Package Contents</i>	<i>Page</i>
<hr/>	
Classes	
CommonBLL 466	
<i>...no description...</i>	
<hr/>	

19.1 Classes

19.1.1 CLASS CommonBLL

DECLARATION

```
public class CommonBLL
extends java.lang.Object
```

METHODS

- *delBarcodeType*
 public **OperationResult** **delBarcodeType**(java.lang.Long id)
- *delBarcodeTypes*
 public **OperationResult** **delBarcodeTypes**(java.util.ArrayList **ISelBarcodeTypes**)
- *delEntities*
 public **OperationResult** **delEntities**(java.util.ArrayList **ISelected**,
 java.lang.Class myClass)
- *delEntity*
 public **OperationResult** **delEntity**(com.imagos.axilice.core.common.BaseVO b)
- *delVatType*
 public **OperationResult** **delVatType**(java.lang.Long id)
- *delVatTypes*
 public **OperationResult** **delVatTypes**(java.util.ArrayList **ISelVatTypes**)
- *getBarcodeType*
 public **BarcodeTypeOT** **getBarcodeType**(java.lang.Long id)
- *getEntity*
 public **BaseVO** **getEntity**(java.lang.Long id, java.lang.Class myClass)
- *getInstance*
 public static final **CommonBLL** **getInstance**()
 - **Usage**
 - * Returns the unique instance of the ManagementBLL class. Implements the singleton design pattern.
 - **Returns** - ManagementBLL, unique instance of ManagementBLL.
- *getVatType*
 public **VatTypeOT** **getVatType**(java.lang.Long id)
- *queryBarcodeTypes*
 public **ArrayList** **queryBarcodeTypes**(java.util.ArrayList **ICriteria**)

- *queryEntities*
public ArrayList queryEntities(java.util.ArrayList ICriteria,
java.lang.Class myClass)
- *queryEntities*
public ArrayList queryEntities(java.lang.String hql)
- *queryVatTypes*
public ArrayList queryVatTypes(java.util.ArrayList ICriteria)
- *saveBarcodeType*
public OperationResult saveBarcodeType(
com.imagos.axilice.core.common.BarcodeTypeOT barcodeType)
- *saveEntity*
public OperationResult saveEntity(com.imagos.axilice.core.common.BaseVO
entity)
- *saveVatType*
public OperationResult saveVatType(
com.imagos.axilice.core.common.VatTypeOT vatType)

Chapter 20

Package com.imagos.axilice.server.warehouse.bl

Package Contents

Page

Classes

ProductBLL	469
<i>Business Logic class that implements the Product business logic processes.</i>	

20.1 Classes

20.1.1 CLASS ProductBLL

Business Logic class that implements the Product business logic processes.
Responsibilities:

- Validation of the products.

DECLARATION

```
public class ProductBLL
extends java.lang.Object
```

METHODS

- *delMeasureUnit*
public OperationResult delMeasureUnit(java.lang.Long id)
- *delMeasureUnits*
public OperationResult delMeasureUnits(java.util.ArrayList lSelMeasureUnits)
- *delProduct*
public OperationResult delProduct(java.lang.Long id)
- *delProductGroup*
public OperationResult delProductGroup(java.lang.Long id)
- *delProductGroups*
public OperationResult delProductGroups(java.util.ArrayList lSelGroups)
- *delProducts*
public OperationResult delProducts(java.util.ArrayList lSelProducts)
- *getInstance*
public static final ProductBLL getInstance()
 - **Usage**
 - * Returns the unique instance of the TraceabilityTreeBLL class. Implements the singleton design pattern.
 - **Returns** - TraceabilityTreeBLL, unique instance of TraceabilityTreeBLL.
- *getMeasureUnit*
public MeasureUnitOT getMeasureUnit(java.lang.Long id)
- *getNextProductGroupId*
public long getNextProductGroupId()
- *getProduct*
public ProductOT getProduct(java.lang.Long id)

- *getProductGroup*
public ProductGroupOT **getProductGroup**(java.lang.Long id)
- *getRevision*
public static String **getRevision**()
- **Usage**
 * Returns tag Id assigned to CVS source file.
- *queryMeasureUnits*
public ArrayList **queryMeasureUnits**(java.util.ArrayList ICriteria)
- *queryProductGroups*
public ArrayList **queryProductGroups**(java.util.ArrayList ICriteria)
- *queryProducts*
public ArrayList **queryProducts**(java.util.ArrayList ICriteria)
- *resetNextProductGroupId*
public void **resetNextProductGroupId**(long idGroup)
- *saveMeasureUnit*
public OperationResult **saveMeasureUnit**(
 com.imagos.axilice.core.warehouse.MeasureUnitOT measureUnit)
- *saveProduct*
public OperationResult **saveProduct**(
 com.imagos.axilice.core.warehouse.ProductOT product)
- *saveProductGroup*
public OperationResult **saveProductGroup**(
 com.imagos.axilice.core.warehouse.ProductGroupOT group)

Chapter 21

Package com.imagos.axilice.core.util

Package Contents

Page

Classes

AddressUtils	473
<i>...no description...</i>	
BarcodeGenerator	474
<i>Generates the images representing barcodes with the help of the Barbecue library.</i>	
BaseVObyIdComparator	475
<i>...no description...</i>	
Converter	475
<i>...no description...</i>	
DateTime	477
<i>Utils for conversion between Dates and Strings.</i>	
FieldValue	480
<i>...no description...</i>	
GoogleMap	481
<i>...no description...</i>	
ImageConverter	482
<i>Converts an AWT/Swing image to its correspondent image object in SWT.</i>	
InstanceLocator	483
<i>Utility class for building a valid instance for a class, given the classname.</i>	
LocalizedResultMessage	484
<i>This class encapsulates the messages returned to the end user as the result of any user operation in the system.</i>	
LoggerFactory	489
<i>Factory class for getting the Logger instance.</i>	
MapUtils	491
<i>...no description...</i>	
OperationResult	492
<i>This class encapsulates the final result of any operation in the system, plus the list of messages generated throughout the operation.</i>	
ReflectionUtils	493
<i>...no description...</i>	
ResultMessage	494
<i>This class encapsulates the messages returned to the end user as the result of any user operation in the system.</i>	

com.imagos.axilice.core.util- ProductBLL 472

ResultMessageCatalog	498
<i>Holds the catalog of available result messages.</i>	
SortUtils	504
<i>...no description...</i>	
Validator	505
<i>Utility class with methods for usual format validations.</i>	

21.1 Classes

21.1.1 CLASS AddressUtils

DECLARATION

```
public class AddressUtils
extends java.lang.Object
```

CONSTRUCTORS

- *AddressUtils*

```
public AddressUtils( )
```

METHODS

- *fillCountryCombo*

```
public int fillCountryCombo( Combo combobox,
com.imagos.axilice.core.common.CountryOT selectedCountry )
```
- *fillStatesCombo*

```
public int fillStatesCombo( Combo combobox,
com.imagos.axilice.core.common.CountryOT selectedCountry,
com.imagos.axilice.core.common.StateOT selectedState )
```
- *getCountries*

```
public ArrayList getCountries( )
```
- *getCountryFromCombo*

```
public CountryOT getCountryFromCombo( Combo combobox )
```
- *getInstance*

```
public static AddressUtils getInstance( )
```
- *getStateFromCombo*

```
public StateOT getStateFromCombo( Combo combobox )
```
- *getStates*

```
public ArrayList getStates( )
```
- *getStates*

```
public ArrayList getStates( com.imagos.axilice.core.common.CountryOT
country )
```
- *refillComboStates*

```
public void refillComboStates( Combo comboStates,
com.imagos.axilice.core.common.CountryOT newCountry )
```
- *setCountries*

```
public void setCountries( java.util.ArrayList countries )
```
- *setStates*

```
public void setStates( java.util.ArrayList states )
```

21.1.2 CLASS BarcodeGenerator

Generates the images representing barcodes with the help of the Barbecue library.
Responsibilities:

- Generate barcode images.

DECLARATION

```
public class BarcodeGenerator
extends java.lang.Object
```

CONSTRUCTORS

- *BarcodeGenerator*
public **BarcodeGenerator**()

METHODS

- *createBarcodeImage*
public static Image **createBarcodeImage**(java.lang.String barcode, Display display)
 - **Usage**
* Create a new Barcode image coded from the barcode data.
 - **Parameters**
* **barcode** - String, barcode data with the number the barcode will be generated for.
* **display** - Display, SWT Display instance used to build the GUI.
 - **Returns** - Image, image object with the barcode generated.

- *createBarcodeImage*
public static Image **createBarcodeImage**(java.lang.String barcode, java.lang.String barcodeType, Display display)
 - **Usage**
* Create a new Barcode image coded from the barcode data.
 - **Parameters**
* **barcode** - String, barcode data with the number the barcode will be generated for.
* **barcodeType** - String, barcode type to generate.
* **display** - Display, SWT Display instance used to build the GUI.
 - **Returns** - Image, image object with the barcode generated.

- *getRevision*
public static String **getRevision**()
 - **Usage**
* Returns tag Id assigned to CVS source file.

21.1.3 CLASS BaseVObyIdComparator

DECLARATION

```
public class BaseVObyIdComparator
extends java.lang.Object
implements java.util.Comparator
```

CONSTRUCTORS

- *BaseVObyIdComparator*
`public BaseVObyIdComparator()`

METHODS

- *compare*
`public int compare(com.imagos.axilice.core.common.BaseVO o1,
com.imagos.axilice.core.common.BaseVO o2)`

21.1.4 CLASS Converter

DECLARATION

```
public class Converter
extends java.lang.Object
```

CONSTRUCTORS

- *Converter*
`public Converter()`

METHODS

- *day*
`public int day(java.util.Date d)`
- *getInstance*
`public static Converter getInstance()`
- *letraNIF*
`public char letraNIF(java.lang.String dni)`
 - Usage

* Calcula la letra del nif segn legislacion espaola

– **Parameters**

* dni -

– **Returns** -

• *month*

public int month(java.util.Date d)

• *parseBigDecimal*

public BigDecimal parseBigDecimal(java.lang.String s)

• *parseCoordinates*

public Point2D parseCoordinates(java.lang.String s)

• *parseDate*

public Date parseDate(java.lang.String text)

• *parseDouble*

public Double parseDouble(java.lang.String s)

• *parseFloat*

public float parseFloat(java.lang.String s)

• *parseInteger*

public Integer parseInteger(java.lang.String s)

• *parseLong*

public Long parseLong(java.lang.String s)

• *setDate*

public void setDate(org.eclipse.swt.widgets.DateTime widget,
java.util.Date d)

• *toDate*

public Date toDate(org.eclipse.swt.widgets.DateTime c)

• *toString*

public String toString(java.math.BigDecimal price)

• *toString*

public String toString(java.util.Date d)

• *toString*

public String toString(double f)

• *toString*

public String toString(java.lang.Double f)

• *toString*

public String toString(float f)

• *toString*

public String toString(java.lang.Float f)

• *toString*

public String toString(java.lang.Integer i)

- *toString*
public String toString(long l)
- *toString*
public String toString(java.lang.String string)
- *year*
public int year(java.util.Date d)

21.1.5 CLASS DateTime

Utils for conversion between Dates and Strings.

DECLARATION

```
public class DateTime
extends java.lang.Object
```

CONSTRUCTORS

- *DateTime*
public **DateTime**()
 - **Usage**
* Default constructor
- *DateTime*
public **DateTime**(java.util.Date date, java.lang.String format)
 - **Usage**
* Constructor from a date in normal format (util.Date format)
 - **Parameters**
* **date** - java.util.Date, date in util.Date format.
* **format** - String, date pattern for formatting the string.
- *DateTime*
public **DateTime**(java.sql.Date date, java.lang.String format)
 - **Usage**
* Constructor from a date in SQL format (sql.Date format)
 - **Parameters**
* **date** - java.sql.Date, date in sql.Date format.
* **format** - String, date pattern for formatting the string.
- *DateTime*
public **DateTime**(com.imagos.axilice.core.util.DateTime d)
 - **Usage**
* Copy constructor.

- *DateTime*

public **DateTime**(java.lang.String date, java.lang.String format)

- **Usage**

- * Constructor from a date in String format.

- **Parameters**

- * **date** - String, date in String format.

- * **format** - String, date pattern for formatting the string.

METHODS

- *addDays*

public void **addDays**(int days)

- **Usage**

- * Add days to the datetime value

- **Parameters**

- * **days** - int, number of days to add.

- *addMonths*

public void **addMonths**(int months)

- **Usage**

- * Add months to the datetime value

- **Parameters**

- * **months** - int, number of months to add.

- *addYears*

public void **addYears**(int years)

- **Usage**

- * Add years to the datetime value.

- **Parameters**

- * **years** - int, number of years to add.

- *getAge*

public int **getAge**()

- **Usage**

- * Get the age, in years passed from the datetime value to the current day.

- **Returns** - int, age.

- *getDate*

public Date **getDate**()

- **Usage**

- * Get the datetime value formatted as SQL compatible date.

- **Returns** - java.sql.Date, datetime value in SQL format.

- *getDateAsString*
`public String getDateAsString()`
 - **Usage**
 - * Get the datetime value formatted as String.
 - **Returns** - String, String value of the datetime.

- *getDay*
`public String getDay()`
 - **Usage**
 - * Get the day of the datetime value.
 - **Returns** - String, day of the datetime value.

- *getDayAsInt*
`public int getDayAsInt()`
 - **Usage**
 - * Get the day of the datetime value as an integer.
 - **Returns** - int, day of the datetime value.

- *getMonth*
`public String getMonth()`
 - **Usage**
 - * Get the month of the datetime value.
 - **Returns** - String, month of the datetime value.

- *getMonthAsInt*
`public int getMonthAsInt()`
 - **Usage**
 - * Get the month of the datetime value as an integer.
 - **Returns** - int, month of the datetime value.

- *getRevision*
`public static String getRevision()`
 - **Usage**
 - * Devuelve el tag Id de CVS asociado al fuente del fichero.

- *getWeekDay*
`public String getWeekDay()`
 - **Usage**
 - * Get the weekday of the datetime value.
 - **Returns** - String, weekday of the datetime value.

- *getYear*
`public String getYear()`
 - **Usage**

- * Get the year of the datetime value.
 - **Returns** - String, year of the datetime value.

- *getYearAsInt*
 - public int **getYearAsInt**()
 - **Usage**
 - * Get the year of the datetime value as an integer.
 - **Returns** - int, year of the datetime value.

- *setDate*
 - public void **setDate**(java.util.Date date)
 - **Usage**
 - * Set the dateTime value
 - **Parameters**
 - * **date** - java.util.Date, date in util.Date format.

- *setDate*
 - public void **setDate**(java.sql.Date date)
 - **Usage**
 - * Set the dateTime value
 - **Parameters**
 - * **date** - java.util.Date, date in sql.Date format.

- *setDate*
 - public void **setDate**(java.lang.String date)
 - **Usage**
 - * Set the dateTime value
 - **Parameters**
 - * **date** - String, date in String format.

- *setDateFormat*
 - public void **setDateFormat**(java.lang.String format)
 - **Usage**
 - * Set the pattern for formatting dates that will be used.
 - **Parameters**
 - * **format** - String, date format pattern.

21.1.6 CLASS FieldValue

DECLARATION

```
public final class FieldValue
extends java.lang.Object
```


CONSTRUCTORS

- *FieldValue*
`public FieldValue()`

METHODS

- *getFieldValue*
`public static Object getFieldValue(java.lang.Class c, java.lang.String fieldName, java.lang.String val)`
- *parseFieldValue*
`public static String parseFieldValue(java.lang.Object o, java.lang.String fieldId)`

21.1.7 CLASS GoogleMap

DECLARATION

```
public class GoogleMap
extends java.lang.Object
```

CONSTRUCTORS

- *GoogleMap*
`public GoogleMap()`
- *GoogleMap*
`public GoogleMap(com.imagos.axilice.core.partners.AddressOT address, int zoom, java.awt.Dimension size)`

METHODS

- *build*
`public OperationResult build()`
- *getAddress*
`public AddressOT getAddress()`
- *getCoordinates*
`public Point2D getCoordinates()`
- *getMapAddress*
`public AddressOT getMapAddress()`
- *getMapImage*
`public Image getMapImage()`

- *getMapSize*
public Dimension getMapSize()
- *getMapURL*
public URL getMapURL()
- *getMapZoom*
public int getMapZoom()
- *setAddress*
public void setAddress(com.imagos.axilice.core.partners.AddressOT address)
- *setCoordinates*
public void setCoordinates(java.awt.geom.Point2D coordinates)
- *setMapAddress*
public void setMapAddress(com.imagos.axilice.core.partners.AddressOT mapAddress)
- *setMapImage*
public void setMapImage(Image mapImage)
- *setMapSize*
public void setMapSize(java.awt.Dimension mapSize)
- *setMapURL*
public void setMapURL(java.net.URL mapURL)
- *setMapZoom*
public void setMapZoom(int mapZoom)

21.1.1.8 CLASS ImageConverter

Converts an AWT/Swing image to its correspondent image object in SWT.
Responsibilities:

- Convert images into SWT compliant image objects.

DECLARATION

```
public class ImageConverter
extends java.lang.Object
```

CONSTRUCTORS

- *ImageConverter*
public **ImageConverter**()

METHODS

- *convertToSWT*

```
public static ImageData convertToSWT( java.awt.image.BufferedImage
bufferedImage )
```

- **Usage**

- * Converts an AWT BufferedImage to SWT format. The library Barbecue, used for the barcode generation, only generates images in AWT format. So a mechanism for converting them into the SWT (GUI library used in the application) is needed.

- **Parameters**

- * **bufferedImage** - BufferedImage, AWT BufferedImage to convert.

- **Returns** - ImageData, SWT formatted image.
Copyright (c) 2000, 2004 IBM Corporation and others. All rights reserved. This program and the accompanying materials are made available under the terms of the Eclipse Public License v1.0 which accompanies this distribution, and is available at <http://www.eclipse.org/legal/epl-v10.html>

Contributors: IBM Corporation - initial API and implementation

- *convertToSWT*

```
public static ImageData convertToSWT( byte [] imgData )
```

- *convertToSWT*

```
public static ImageData convertToSWT( Display display, byte [] imgData )
```

- *getRevision*

```
public static String getRevision( )
```

- **Usage**

- * Returns tag Id assigned to CVS source file.

21.1.1.9 CLASS InstanceLocator

Utility class for building a valid instance for a class, given the classname. It uses the Java reflection to get the unique instance (if the selected class implements the singleton pattern), or create a new instance for the selected class.
Responsibilities:

- Providing a valid instance for a class.

DECLARATION

<pre>public class InstanceLocator extends java.lang.Object</pre>
--

FIELDS

- public static final String ERROR_NOCLASS
 - Constant value with the logging message for an error occurred during the loading of the entity's helper class.
- public static final String ERROR_NOINSTANCE
 - Constant value with the logging message for an error occurred during the instantiation of the entity's helper class.

CONSTRUCTORS

- *InstanceLocator*
public **InstanceLocator**(*Logger* logger)

METHODS

- *getClassInstance*
public Object getClassInstance(java.lang.String classname)
- *getRevision*
public static String **getRevision**()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.

21.1.10 CLASS LocalizedResultMessage

This class encapsulates the messages returned to the end user as the result of any user operation in the system.
Responsibilities:

- Encapsulate and pass on the error messages to the end user.

DECLARATION

```
public class LocalizedResultMessage
extends java.lang.Object
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private int resCode
 - Internal result code to support result localization and classification. It may be either an OK code or any error code supported by the system.
- private String resMessage
 - Result message string that is showed to the end user.
- private ArrayList IParams
 - List of parameters used in the construction of the result message. The number of parameters may vary according to the type of result message built.
- private Throwable exc
 - Error exception that generated the result message. Only applies when the result message is an error message.
- private String fieldName
 - Name of the form field linked to the result message (for example for validation result messages). More than one field name can be linked by appending all the fields names using the "—" character as separator. Only applies when the result message is an error message.
- private I18n i18n
 -

CONSTRUCTORS

- *LocalizedResultMessage*

```
public LocalizedResultMessage( int resCode )
```

 - **Usage**
 - * Constructor for Simple ResultMessage class. Builds an appropriate result message for the result code.
 - **Parameters**
 - * **resCode** - int, result message internal code.
-
- *LocalizedResultMessage*

```
public LocalizedResultMessage( int resCode, java.util.ArrayList IParams, I18n i18n )
```

 - **Usage**
 - * Constructor for Parametrized ResultMessage class. Builds an appropriate result message for the result code, parametrizing the message with the right values.
 - **Parameters**
 - * **resCode** - int, result message internal code.
 - * **lParams** - ArrayList, list of the parameter values for the message.
 - * **i18n** - I18n, instance of Internationalization object.

- *LocalizedResultMessage*

```
public LocalizedResultMessage( int   resCode, java.util.ArrayList  lParams,
                               I18n  i18n, java.lang.String  fieldName )
```

- Usage

- * Constructor for Parametrized Error ResultMessage class. Builds an appropriate result message for the result code, parametrizing the message with the right values and identifying the cause of the message. It also adds a form field linked to the result message.

- Parameters

- * **resCode** - int, result message internal code.
 - * **lParams** - ArrayList, list of the parameter values for the message.
 - * **i18n** - I18n, instance of Internationalization object.
 - * **fieldName** - String, form field's name linked to the result message.
-

- *LocalizedResultMessage*

```
public LocalizedResultMessage( int   resCode, I18n  i18n )
```

- Usage

- * Constructor for Simple ResultMessage class. Builds an appropriate result message for the result code.

- Parameters

- * **resCode** - int, result message internal code.
 - * **i18n** - I18n, instance of Internationalization object.
-

- *LocalizedResultMessage*

```
public LocalizedResultMessage( int   resCode, java.lang.Throwable  cause,
                               java.util.ArrayList  lParams, I18n  i18n )
```

- Usage

- * Constructor for Parametrized Error ResultMessage class. Builds an appropriate result message for the result code, parametrizing the message with the right values and identifying the cause of the message.

- Parameters

- * **resCode** - int, result message internal code.
 - * **cause** - Throwable, exception that raised the result message. Only applies with error messages.
 - * **lParams** - ArrayList, list of the parameter values for the message.
 - * **i18n** - I18n, instance of Internationalization object.
-

- *LocalizedResultMessage*

```
public LocalizedResultMessage( int   resCode, java.lang.Throwable  cause,
                               java.util.ArrayList  lParams, I18n  i18n, java.lang.String  fieldName )
```

- Usage

- * Constructor for Parametrized Error ResultMessage class. Builds an appropriate result message for the result code, parametrizing the message with the right values and identifying the cause of the message. It also adds a form field linked to the result message.

- Parameters

- * **resCode** - int, result message internal code.
- * **cause** - Throwable, exception that raised the result message. Only applies with error messages.
- * **lParams** - ArrayList, list of the parameter values for the message.
- * **i18n** - I18n, instance of Internationalization object.
- * **fieldName** - String, form field's name linked to the result message.

- *LocalizedResultMessage*

```
public LocalizedResultMessage( int resCode, java.lang.Throwable cause,
java.util.ArrayList lParams, java.lang.String fieldName )
```

- **Usage**

- * Constructor for Parametrized Error ResultMessage class. Builds an appropriate result message for the result code, parametrizing the message with the right values and identifying the cause of the message. It also adds a form field linked to the result message.

- **Parameters**

- * **resCode** - int, result message internal code.
- * **cause** - Throwable, exception that raised the result message. Only applies with error messages.
- * **lParams** - ArrayList, list of the parameter values for the message.
- * **fieldName** - String, form field's name linked to the result message.

- *LocalizedResultMessage*

```
public LocalizedResultMessage( int resCode, java.lang.Throwable cause,
I18n i18n )
```

- **Usage**

- * Constructor for Simple Error ResultMessage class. Builds an appropriate result message for the result code, identifying the cause of the message.

- **Parameters**

- * **resCode** - int, result message internal code.
- * **cause** - Throwable, exception that raised the result message. Only applies with error messages.
- * **i18n** - I18n, instance of Internationalization object.

METHODS

- *getExc*

```
public Throwable getExc( )
```

- **Usage**

- * Gets the exception that raised the result message. It comes normally in error or warning messages.

- **Returns** - Throwable, exception linked to the result message.

- *getFieldName*

```
public String getFieldName( )
```

- **Usage**

- * Gets the name of the field that generated the error message.

– **Returns** - String, form field's name.

- *getLParams*

`public ArrayList getLParams()`

– **Usage**

* Gets the list of parameters and their values, used in the construction of the result message.

– **Returns** - ArrayList, list of result message parameters.

- *getMessage*

`public String getMessage()`

– **Usage**

* Constructs the formatted result message text.

– **Returns** - String, formatted result message text.

- *getResCode*

`public int getResCode()`

– **Usage**

* Gets the result code for the result message. This code helps the coder to identify and locate the error source.

– **Returns** - int, result code.

- *getResMessage*

`public String getResMessage()`

– **Usage**

* Gets the message label that will be used for constructing the formatted error message. This label will be the base of the message, constructed also with the parameters specified and using the internationalization mechanism to translate the message to the user's language if it's supported.

– **Returns** - String, message label.

- *getRevision*

`public static String getRevision()`

– **Usage**

* Returns tag Id assigned to CVS source file.

- *setExc*

`public void setExc(java.lang.Throwable exc)`

– **Usage**

* Sets up the exception that raised the result message. It comes normally in error or warning messages.

– **Parameters**

* **exc** - Throwable, exception linked to the result message.

- *setFieldName*

`public void setFieldName(java.lang.String fieldName)`

– **Usage**

* Sets up the name of the field that generated the error message.

– **Parameters**

* **fieldName** - String, form field's name.

- *setLParams*

```
public void setLParams( java.util.ArrayList params )
```

– **Usage**

* Sets up the list of parameters and their values, used in the construction of the result message.

– **Parameters**

* **params** - ArrayList, list of result message parameters.

- *setResCode*

```
public void setResCode( int resCode )
```

– **Usage**

* Sets up the result code for the result message. This code helps the coder to identify and locate the error source.

– **Parameters**

* **resCode** - int, result code.

- *setResMessage*

```
public void setResMessage( java.lang.String resMessage )
```

– **Usage**

* Sets up the message label that will be used for constructing the formatted error message. This label will be the base of the message, constructed also with the parameters specified and using the internationalization mechanism to translate the message to the user's language if it's supported.

– **Parameters**

* **resMessage** - String, message label.

- *toString*

```
public String toString( )
```

– **Usage**

* Format the result message to show it to the end user

– **Returns** - String, the formatted result message.

21.1.11 CLASS **LoggerFactory**

Factory class for getting the Logger instance. It initializes and configures the logging system.

Responsibilities:

- Configuration and initialization of the logging system.

DECLARATION

```
public class LoggerFactory
extends java.lang.Object
```

CONSTRUCTORS

- *LoggerFactory*
public **LoggerFactory**()

METHODS

- *getLogger*
public static final **Logger** **getLogger**(java.lang.String category)
 - **Usage**
 - * Returns the unique instance of the **Logger** class.
 - **Returns** - **Logger**, unique instance of **Logger**.

- *getRevision*
public static **String** **getRevision**()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.

- *initLogger*
public static final void **initLogger**()
 - **Usage**
 - * Configures and initializes the logging system with the default values. The chosen logging system is Apache Logging4Java library.

- *initLogger*
public static final void **initLogger**(java.lang.String logDir,
java.io.InputStream isConfigFile)
 - **Usage**
 - * Configures and initializes the logging system. The chosen logging system is Apache Logging4Java library.
 - **Parameters**
 - * **logDir** - **String**, path to the directory where logs are stored.
 - * **isConfigFile** - **InputStream**, input stream opened to the logging configuration file.

- *initLogger*
public static final void **initLogger**(java.lang.String logDir,
java.lang.String configFile)
 - **Usage**

- * Configures and initializes the logging system. The chosen logging system is Apache Logging4Java library.

– **Parameters**

- * `logDir` - String, path to the directory where logs are stored.
- * `configFile` - String, path to the logging configuration file.

• *initLogger*

```
public static final void initLogger( java.lang.String logDir, java.net.URL
configFile )
```

– **Usage**

- * Configures and initializes the logging system. The chosen logging system is Apache Logging4Java library.

– **Parameters**

- * `logDir` - String, path to the directory where logs are stored.
- * `configFile` - URL, URL that points to the logging configuration file.

21.1.12 CLASS MapUtils

DECLARATION

```
public class MapUtils
extends java.lang.Object
```

CONSTRUCTORS

- *MapUtils*
public **MapUtils**()

METHODS

- *geocode*
public **Point2D** **geocode**(com.imagos.axilice.core.partners.AddressOT address)
- *getGoogleMapImage*
public **Image** **getGoogleMapImage**(com.imagos.axilice.core.partners.AddressOT address, java.awt.Dimension size, int zoom)
- *getGoogleMapImage*
public **Image** **getGoogleMapImage**(java.awt.geom.Point2D c, java.awt.Dimension size, int zoom)
- *getGoogleStaticMapURL*
public **URL** **getGoogleStaticMapURL**(java.awt.geom.Point2D coordinates, java.awt.Dimension size, int zoom)

- *getInstance*
public static MapUtils **getInstance**()
- *getYahooMapImage*
public Image **getYahooMapImage**(com.imagos.axilice.core.partners.AddressOT
address, int imgWidt, int imgHeight, int zoom)
- *getYahooMapImageURL*
public URL **getYahooMapImageURL**(
com.imagos.axilice.core.partners.AddressOT address, int imgWidt, int
imgHeight, int zoom)

21.1.13 CLASS *OperationResult*

This class encapsulates the final result of any operation in the system, plus the list of messages generated throughout the operation.
Responsibilities:

- Encapsulate and pass on the operation result to the end user.
- Keep the list of result messages generated throughtout the operation.

DECLARATION

```
public class OperationResult
extends java.lang.Object
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private ArrayList lResultMessages

—

- private int result

—

FIELDS

- public static final int RESULT_OK

—

- public static final int RESULT_WARNING

—

- public static final int RESULT_ERROR

—

- public static final int RESULT_UNDEFINED

—

CONSTRUCTORS

- *OperationResult*
public **OperationResult**()

METHODS

- *add*
public void **add**(com.imagos.axilice.core.util.OperationResult result)
- *addResultMessage*
public void **addResultMessage**(int result,
com.imagos.axilice.core.util.ResultMessage msg)
- *clearResultMessagesList*
public void **clearResultMessagesList**()
- *getFormattedResult*
public String **getFormattedResult**()
- *getResult*
public int **getResult**()
- *getResultMessagesList*
public ArrayList **getResultMessagesList**()
- *getRevision*
public static String **getRevision**()

– **Usage**
* Returns tag Id assigned to CVS source file.
- *setResult*
public void **setResult**(int result)

21.1.14 CLASS ReflectionUtils

DECLARATION

```
public class ReflectionUtils
extends java.lang.Object
```

CONSTRUCTORS

- *ReflectionUtils*
public **ReflectionUtils**()

METHODS

• *cancelField*

```
public static OperationResult cancelField(
    com.imagos.axilice.core.common.BaseVO b, java.lang.String fieldName )
```

– Usage

* Put null value in a field (Using Reflection)

– Parameters

* **b** - BaseVO

* **fieldName** - Name of the field to cancel

– Returns - Result of the operation

• *getClassShortName*

```
public static String getClassShortName( java.lang.Class myClass )
```

• *getter*

```
public static Method getter( java.lang.Class myClass, java.lang.String
    fieldName )
```

• *setter*

```
public static Method setter( java.lang.Class myClass, java.lang.String
    fieldName )
```

21.1.15 CLASS ResultMessage

This class encapsulates the messages returned to the end user as the result of any user operation in the system.
Responsibilities:

- Encapsulate and pass on the error messages to the end user.

DECLARATION

```
public class ResultMessage
extends java.lang.Object
implements java.io.Serializable
```

SERIALIZABLE FIELDS

• private int resCode

– Internal result code to support result localization and classification. It may be either an OK code or any error code supported by the system.

• private String resMessage

– Result message string that is showed to the end user.

• private ArrayList lParams

- List of parameters used in the construction of the result message. The number of parameters may vary according to the type of result message built.
- private Throwable exc
 - Error exception that generated the result message. Only applies when the result message is an error message.
- private String fieldName
 - Name of the form field linked to the result message (for example for validation result messages). More than one field name can be linked by appending all the fields names using the “—” character as separator. Only applies when the result message is an error message.

CONSTRUCTORS

- *ResultMessage*
public ResultMessage(int resCode)
 - **Usage**
 - * Constructor for Simple ResultMessage class. Builds an appropriate result message for the result code.
 - **Parameters**
 - * **resCode** - int, result message internal code.

- *ResultMessage*
public ResultMessage(int resCode, java.util.ArrayList lParams)
 - **Usage**
 - * Constructor for Parametrized ResultMessage class. Builds an appropriate result message for the result code, parametrizing the message with the right values.
 - **Parameters**
 - * **resCode** - int, result message internal code.
 - * **lParams** - ArrayList, list of the parameter values for the message.

- *ResultMessage*
public ResultMessage(int resCode, java.lang.Throwable cause)
 - **Usage**
 - * Constructor for Simple Error ResultMessage class. Builds an appropriate result message for the result code, identifying the cause of the message.
 - **Parameters**
 - * **resCode** - int, result message internal code.
 - * **cause** - Throwable, exception that raised the result message. Only applies with error messages.

- *ResultMessage*
public ResultMessage(int resCode, java.lang.Throwable cause, java.util.ArrayList lParams)
 - **Usage**

- * Constructor for Parametrized Error ResultMessage class. Builds an appropriate result message for the result code, parametrizing the message with the right values and identifying the cause of the message.

– **Parameters**

- * **resCode** - int, result message internal code.
- * **cause** - Throwable, exception that raised the result message. Only applies with error messages.
- * **lParams** - ArrayList, list of the parameter values for the message.

• *ResultMessage*

```
public ResultMessage( int resCode, java.lang.Throwable cause,
java.util.ArrayList lParams, java.lang.String fieldName )
```

– **Usage**

- * Constructor for Parametrized Error ResultMessage class. Builds an appropriate result message for the result code, parametrizing the message with the right values and identifying the cause of the message. It also adds a form field linked to the result message.

– **Parameters**

- * **resCode** - int, result message internal code.
- * **cause** - Throwable, exception that raised the result message. Only applies with error messages.
- * **lParams** - ArrayList, list of the parameter values for the message.
- * **fieldName** - String, form field's name linked to the result message.

METHODS

• *getExc*

```
public Throwable getExc( )
```

– **Usage**

- * Gets the exception that raised the result message. It comes normally in error or warning messages.

– **Returns** - Throwable, exception linked to the result message.

• *getFieldName*

```
public String getFieldName( )
```

– **Usage**

- * Gets the name of the field that generated the error message.

– **Returns** - String, form field's name.

• *getLParams*

```
public ArrayList getLParams( )
```

– **Usage**

- * Gets the list of parameters and their values, used in the construction of the result message.

– **Returns** - ArrayList, list of result message parameters.

- *getMessage*
`public String getMessage()`
 - **Usage**
 - * Constructs the formatted result message text.
 - **Returns** - String, formatted result message text.

- *getResCode*
`public int getResCode()`
 - **Usage**
 - * Gets the result code for the result message. This code helps the coder to identify and locate the error source.
 - **Returns** - int, result code.

- *getResMessage*
`public String getResMessage()`
 - **Usage**
 - * Gets the message label that will be used for constructing the formatted error message. This label will be the base of the message, constructed also with the parameters specified and using the internationalization mechanism to translate the message to the user's language if it's supported.
 - **Returns** - String, message label.

- *getRevision*
`public static String getRevision()`
 - **Usage**
 - * Returns tag Id assigned to CVS source file.

- *setExc*
`public void setExc(java.lang.Throwable exc)`
 - **Usage**
 - * Sets up the exception that raised the result message. It comes normally in error or warning messages.
 - **Parameters**
 - * `exc` - Throwable, exception linked to the result message.

- *setFieldName*
`public void setFieldName(java.lang.String fieldName)`
 - **Usage**
 - * Sets up the name of the field that generated the error message.
 - **Parameters**
 - * `fieldName` - String, form field's name.

- *setLParams*
`public void setLParams(java.util.ArrayList params)`
 - **Usage**

- * Sets up the list of parameters and their values, used in the construction of the result message.

– **Parameters**

- * **params** - ArrayList, list of result message parameters.

• *setResCode*

```
public void setResCode( int resCode )
```

– **Usage**

- * Sets up the result code for the result message. This code helps the coder to identify and locate the error source.

– **Parameters**

- * **resCode** - int, result code.

• *setResMessage*

```
public void setResMessage( java.lang.String resMessage )
```

– **Usage**

- * Sets up the message label that will be used for constructing the formatted error message. This label will be the base of the message, constructed also with the parameters specified and using the internationalization mechanism to translate the message to the user's language if it's supported.

– **Parameters**

- * **resMessage** - String, message label.

• *toString*

```
public String toString( )
```

– **Usage**

- * Format the result message to show it to the end user

– **Returns** - String, the formatted result message.

21.1.16 CLASS ResultMessageCatalog

Holds the catalog of available result messages.
Responsibilities:

- Store all the messages available.
- Get messages from the messages resource file.

DECLARATION

```
public class ResultMessageCatalog
extends java.lang.Object
```

CONSTRUCTORS

• *ResultMessageCatalog*

```
public ResultMessageCatalog( java.io.InputStream  isResourcesFile )
```

– Usage

- * Constructor for ResultMessageCatalog. Inits and builds an appropriate result message catalog for the module.

– Parameters

- * **isResourcesFile** - InputStream, input stream opened for reading the messages resources file.
-

• *ResultMessageCatalog*

```
public ResultMessageCatalog( java.io.InputStream  isResourcesFile, I18n  
i18n )
```

– Usage

- * Constructor for ResultMessageCatalog. Inits and builds an appropriate result message catalog for the module.

– Parameters

- * **isResourcesFile** - InputStream, input stream opened for reading the messages resources file.
 - * **i18n** - I18n, instance of internationalization used for localizing messages.
-

• *ResultMessageCatalog*

```
public ResultMessageCatalog( java.io.InputStream  isResourcesFile, I18n  
i18n, Logger  logger )
```

– Usage

- * Constructor for ResultMessageCatalog. Inits and builds an appropriate result message catalog for the module.

– Parameters

- * **isResourcesFile** - InputStream, input stream opened for reading the messages resources file.
 - * **i18n** - I18n, instance of internationalization used for localizing messages.
 - * **logger** - Logger, logger instance used for the logging of the module.
-

• *ResultMessageCatalog*

```
public ResultMessageCatalog( java.lang.String  resourcesFile )
```

– Usage

- * Default constructor for ResultMessageCatalog. Inits and builds an appropriate result message catalog for the module.

– Parameters

- * **resourcesFile** - String, path to the messages resources file.
-

• *ResultMessageCatalog*

```
public ResultMessageCatalog( java.lang.String  resourcesFile, I18n  i18n )
```

– Usage

- * Constructor for ResultMessageCatalog. Inits and builds an appropriate result message catalog for the module.

– **Parameters**

- * **resourcesFile** - String, path to the messages resources file.
- * **i18n** - I18n, instance of internationalization used for localizing messages.

• *ResultMessageCatalog*

```
public ResultMessageCatalog( java.lang.String resourcesFile, I18n i18n,
    Logger logger )
```

– **Usage**

- * Constructor for ResultMessageCatalog. Inits and builds an appropriate result message catalog for the module.

– **Parameters**

- * **resourcesFile** - String, path to the messages resources file.
- * **i18n** - I18n, instance of internationalization used for localizing messages.
- * **logger** - Logger, logger instance used for the logging of the module.

METHODS

• *addResultMessages*

```
public void addResultMessages( java.io.InputStream isResourcesFile )
```

– **Usage**

- * Add a list of result messages to the catalog. The new result messages will be loaded from a different messages properties file

– **Parameters**

- * **isResourcesFile** - InputStream, stream opened to the new messages properties file.

• *getI18n*

```
public I18n getI18n( )
```

– **Usage**

- * Gets the internationalization object used for localizing messages.

– **Returns** - I18n, internationalization object.

• *getMessage*

```
public String getMessage( int resultCode )
```

– **Usage**

- * Returns the message string defined for the result code. If the message code cannot be found in this catalog the code is searched in the parent catalogs.

– **Parameters**

- * **resultCode** - int, result code.

– **Returns** - String, message literal defined for the result code.

• *getParent*

```
public ResultMessageCatalog getParent( )
```

- **Usage**
 - * Gets the parent messages catalog.
 - **Returns** - ResultMessageCatalog, parent messages catalog.
-
- *getResourcesFile*
 public String getResourcesFile()
 - **Usage**
 - * Gets the message resource file.
 - **Returns** - String, path to the messages resource file.
-
- *getRevision*
 public static String getRevision()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
-
- *newResultMessage*
 public ResultMessage newResultMessage(int resCode)
 - **Usage**
 - * Creates a new ResultMessage appropriate for the result code.
 - **Parameters**
 - * resCode - int, result message internal code.
 - **Returns** - ResultMessage, the result message created.
-
- *newResultMessage*
 public ResultMessage newResultMessage(int resCode, java.util.ArrayList lParams)
 - **Usage**
 - * Creates a new ResultMessage appropriate for the result code, parametrizing the message with the right values.
 - **Parameters**
 - * resCode - int, result message internal code.
 - * lParams - ArrayList, list of the parameter values for the message.
 - **Returns** - ResultMessage, the result message created.
-
- *newResultMessage*
 public ResultMessage newResultMessage(int resCode, java.util.ArrayList lParams, java.lang.String fieldName)
 - **Usage**
 - * Creates a new ResultMessage appropriate for the result code, parametrizing the message with the right values and identifying the cause of the message. It also adds a form field linked to the result message.
 - **Parameters**
 - * resCode - int, result message internal code.
 - * lParams - ArrayList, list of the parameter values for the message.
 - * fieldName - String, form field's name linked to the result message.
 - **Returns** - ResultMessage, the result message created.

- *newResultMessage*

```
public ResultMessage newResultMessage( int  resCode, java.lang.Object
param1, java.lang.Object  param2 )
```

- **Usage**

- * Creates a new ResultMessage appropriate for the result code, parametrizing the message with the right values. It's a shortcut for using only one or two parameters, without building a parameter list.

- **Parameters**

- * **resCode** - int, result message internal code.
 - * **param1** - Object, first parameter values for the message.
 - * **param2** - Object, second parameter values for the message.

- **Returns** - ResultMessage, the result message created.

- *newResultMessage*

```
public ResultMessage newResultMessage( int  resCode, java.lang.Object
param1, java.lang.Object  param2, java.lang.String  fieldName )
```

- **Usage**

- * Creates a new ResultMessage appropriate for the result code, parametrizing the message with the right values. It also adds a form field linked to the result message. It's a shortcut for using only one or two parameters, without building a parameter list.

- **Parameters**

- * **resCode** - int, result message internal code.
 - * **param1** - Object, first parameter values for the message.
 - * **param2** - Object, second parameter values for the message.
 - * **fieldName** - String, form field's name linked to the result message.

- **Returns** - ResultMessage, the result message created.

- *newResultMessage*

```
public ResultMessage newResultMessage( int  resCode, java.lang.Throwable
cause )
```

- **Usage**

- * Creates a new ResultMessage appropriate for the result code, identifying the cause of the message.

- **Parameters**

- * **resCode** - int, result message internal code.
 - * **cause** - Throwable, exception that raised the result message. Only applies with error messages.

- **Returns** - ResultMessage, the result message created.

- *newResultMessage*

```
public ResultMessage newResultMessage( int  resCode, java.lang.Throwable
cause, java.util.ArrayList  lParams )
```

- **Usage**

- * Creates a new ResultMessage appropriate for the result code, parametrizing the message with the right values and identifying the cause of the message.

– **Parameters**

- * **resCode** - int, result message internal code.
- * **cause** - Throwable, exception that raised the result message. Only applies with error messages.
- * **lParams** - ArrayList, list of the parameter values for the message.

– **Returns** - ResultMessage, the result message created.

• *newResultMessage*

```
public ResultMessage newResultMessage( int resCode, java.lang.Throwable
cause, java.util.ArrayList lParams, java.lang.String fieldName )
```

– **Usage**

- * Creates a new ResultMessage appropriate for the result code, parametrizing the message with the right values and identifying the cause of the message. It also adds a form field linked to the result message.

– **Parameters**

- * **resCode** - int, result message internal code.
- * **cause** - Throwable, exception that raised the result message. Only applies with error messages.
- * **lParams** - ArrayList, list of the parameter values for the message.
- * **fieldName** - String, form field's name linked to the result message.

– **Returns** - ResultMessage, the result message created.

• *newResultMessage*

```
public ResultMessage newResultMessage( int resCode, java.lang.Throwable
cause, java.lang.Object param1, java.lang.Object param2 )
```

– **Usage**

- * Creates a new ResultMessage appropriate for the result code, parametrizing the message with the right values and identifying the cause of the message. It's a shortcut for using only one or two parameters, without building a parameter list.

– **Parameters**

- * **resCode** - int, result message internal code.
- * **cause** - Throwable, exception that raised the result message. Only applies with error messages.
- * **param1** - Object, first parameter values for the message.
- * **param2** - Object, second parameter values for the message.

– **Returns** - ResultMessage, the result message created.

• *newResultMessage*

```
public ResultMessage newResultMessage( int resCode, java.lang.Throwable
cause, java.lang.Object param1, java.lang.Object param2,
java.lang.String fieldName )
```

– **Usage**

- * Creates a new ResultMessage appropriate for the result code, parametrizing the message with the right values and identifying the cause of the message. It also adds a form field linked to the result message. It's a shortcut for using only one or two parameters, without building a parameter list.

– **Parameters**

- * **resCode** - int, result message internal code.

- * **cause** - Throwable, exception that raised the result message. Only applies with error messages.
 - * **param1** - Object, first parameter values for the message.
 - * **param2** - Object, second parameter values for the message.
 - * **fieldName** - String, form field's name linked to the result message.
 - **Returns** - ResultMessage, the result message created.
-
- *printMessages*
 public void **printMessages**()
 – **Usage**
 * Print the list of loaded messages available in the catalog to the standard console.
-
- *printMessages*
 public void **printMessages**(java.io.PrintStream **out**)
 – **Usage**
 * Print the list of loaded messages available in the catalog to the indicated output stream.
-
- *setI18n*
 public void **setI18n**(I18n **i18n**)
 – **Usage**
 * Sets up the internationalization object used for localizing messages.
 – **Parameters**
 * **I18n** - i18n, internationalization object.
-
- *setParent*
 public void **setParent**(com.imagos.axilice.core.util.ResultMessageCatalog **parent**)
 – **Usage**
 * Sets up the parent messages catalog.
 – **Parameters**
 * **parent** - ResultMessageCatalog, parent messages catalog.
-
- *setResourcesFile*
 public void **setResourcesFile**(java.lang.String **resourcesFile**)
 – **Usage**
 * Sets up the message resource file.
 – **Parameters**
 * **resourcesFile** - String, path to the messages resource file.
-
- *translateMessage*
 public ResultMessage **translateMessage**(
 com.imagos.axilice.core.util.ResultMessage **resMsg**)

DECLARATION

```
public class SortUtils
extends java.lang.Object
```

CONSTRUCTORS

- *SortUtils*
public **SortUtils**()

METHODS

- *sortLBaseVO*
public static void **sortLBaseVO**(java.util.List c)
 - **Usage**
* Utility class

Sort list of entities by id
 - **Parameters**
* c - List of entities to be sorted

21.1.18 CLASS Validator

Utility class with methods for usual format validations.

DECLARATION

```
public class Validator
extends java.lang.Object
```

METHODS

- *getInstance*
public static Validator **getInstance**()
 - **Usage**
* Return the unique instance of the class Validator (Singleton pattern)
 - **Returns** - Validator, static and unique instance of the class Validator.

- *getRevision*
public static String **getRevision**()
 - **Usage**

* Return the CVS tag id of the file.

- *isDateAfter*

```
public boolean isDateAfter( java.lang.String  date1, java.lang.String  date2
)
```

- **Usage**

- * Check if the first date comes after the second date.

- **Parameters**

- * **date1** - String, first date

- * **date2** - String, second date

- **Returns** - boolean, True if date1 >date2; False otherwise.

- *isDateInRange*

```
public boolean isDateInRange( java.sql.Date  date, java.sql.Date  dateMax,
java.sql.Date  dateMin )
```

- **Usage**

- * Check if a date is in the range [dateMin .. dateMax].

- **Parameters**

- * **date** - Date, date value

- * **dateMax** - Date, top date of the range.

- * **dateMin** - Date, bottom date of the range.

- **Returns** - boolean, True if dateMin <= date <= dateMax; False otherwise

- *isDateInRange*

```
public boolean isDateInRange( java.sql.Date  date, java.lang.String
dateMax, java.lang.String  dateMin )
```

- **Usage**

- * Check if a date is in the range [dateMin .. dateMax].

- **Parameters**

- * **date** - Date, date value

- * **dateMax** - Date, top date of the range.

- * **dateMin** - Date, bottom date of the range.

- **Returns** - boolean, True if dateMin <= date <= dateMax; False otherwise

- *isDateInRange*

```
public boolean isDateInRange( com.imagos.axilice.core.util.DateTime  date,
com.imagos.axilice.core.util.DateTime  dateMin,
com.imagos.axilice.core.util.DateTime  dateMax )
```

- **Usage**

- * Comprueba si una dateha est comprendida en el rango especificado.

- **Parameters**

- * **date** - DateTime, con la dateha a validar.

- * **dateMax** - DateTime, con la dateha superior del rango.

- * **dateMin** - DateTime, con la dateha inferior del rango.

- **Returns** - boolean, True si dateMin <= date <= dateMax; False en caso contrario

- *isDateInRange*

```
public boolean isDateInRange( java.lang.String  date, java.lang.String
dateMax, java.lang.String  dateMin )
```

- **Usage**

- * Check if a date is in the range [dateMin .. dateMax].

- **Parameters**

- * **date** - String, date value
 - * **dateMax** - String, top date of the range.
 - * **dateMin** - String, bottom date of the range.

- **Returns** - boolean, True if dateMin <= date <= dateMax; False otherwise

- *isEmpty*

```
public boolean isEmpty( java.lang.String  s )
```

- **Usage**

- * Check if the String is empty.

- **Parameters**

- * **s** - String to check.

- **Returns** - boolean, True if the string is empty; False otherwise

- *isNumberInRange*

```
public boolean isNumberInRange( java.lang.String  num, double  max,
double  min )
```

- **Usage**

- * Check if a float number is in the range [min .. max].

- **Parameters**

- * **num** - String, float numeric value
 - * **max** - double, top number of the range.
 - * **min** - double, bottom number of the range.

- **Returns** - boolean, True if min <= num <= max; False otherwise

- *isNumberInRange*

```
public boolean isNumberInRange( java.lang.String  num, int  max, int
min )
```

- **Usage**

- * Check if an integer number is in the range [min .. max].

- **Parameters**

- * **num** - String, numeric value
 - * **max** - int, top of the range.
 - * **min** - int, bottom of the range.

- **Returns** - boolean, True if min <= num <= max; False otherwise

- *isValidAlpha*

```
public boolean isValidAlpha( java.lang.String  s )
```

- **Usage**

- * Check if the string is a alphabetic string (formed by only letters).

- **Parameters**

- * **s** - String to check.
 - **Returns** - boolean, True if the string is a valid alphabetic string; False otherwise

- *isValidAlphaNum*
 public boolean isValidAlphaNum(java.lang.String s)
 - **Usage**
 - * Check if the string is a alphanumeric string (formed by only letters or digits).
 - **Parameters**
 - * **s** - String to check.
 - **Returns** - boolean, True if the string is a valid alphanumeric string; False otherwise

- *isValidBankAccount*
 public boolean isValidBankAccount(long idBank, long idBranch, long idControlNumber, long idAccountNumber)

- *isValidBarcode*
 public boolean isValidBarcode(java.lang.String s)
 - **Usage**
 - * Check if the data is a barcode
 - **Parameters**
 - * **s** - String to check
 - **Returns** - boolean, True if the string is a valid barcode; False otherwise

- *isValidBoolean*
 public boolean isValidBoolean(java.lang.String s)
 - **Usage**
 - * Check if the string is a boolean.
 - **Parameters**
 - * **s** - String to check.
 - **Returns** - boolean, True if the string is a valid boolean; False otherwise

- *isValidCharacterAlpha*
 public boolean isValidCharacterAlpha(java.lang.String s)
 - **Usage**
 - * Check if the string is a unique character of type letter.
 - **Parameters**
 - * **s** - String to check.
 - **Returns** - boolean, True if the string is a valid letter; False otherwise

- *isValidCcc*
 public boolean isValidCcc(java.lang.String cccEnt, java.lang.String cccSuc, java.lang.String cccDc, java.lang.String cccNum)
 - **Usage**
 - * Check if the strings are a valid current account.
 - **Parameters**

- * **cccEnt** - String, Bank entity of the current account.
- * **cccSuc** - String, Bank branch of the current account.
- * **cccDc** - String, Control digits of the current account.
- * **cccNum** - String, Bank account number.

– **Returns** - boolean, True if the strings form a valid current account number; False otherwise.

- *isValidCharacter*

public boolean isValidCharacter(java.lang.String s)

– **Usage**

- * Check if the string is a unique character (letter, number or special character).

– **Parameters**

- * **s** - String to check.

– **Returns** - boolean, True if the string is a valid character; False otherwise

- *isValidDate*

public boolean isValidDate(java.lang.String s)

– **Usage**

- * Check if the string is a date.

– **Parameters**

- * **s** - String to check.

– **Returns** - boolean, True if the string is a valid date; False otherwise

- *isValidDateTime*

public boolean isValidDateTime(java.lang.String s)

– **Usage**

- * Check if the string is a date with time.

– **Parameters**

- * **s** - String to check.

– **Returns** - boolean, True if the string is a valid date time; False otherwise

- *isValidDigit*

public boolean isValidDigit(java.lang.String s)

– **Usage**

- * Check if the string is a unique character of type digit.

– **Parameters**

- * **s** - String to check.

– **Returns** - boolean, True if the string is a valid digit; False otherwise

- *isValidDNI*

public boolean isValidDNI(java.lang.String dni)

– **Usage**

- * Asses where is a valid identity card (only numbers)

– **Parameters**

- * **dni** - to validate

– **Returns** - true if a valid dni

- *isValidDni*

```
public boolean isValidDni( java.lang.String docType, java.lang.String  
docNum, java.lang.String docLet )
```

– **Usage**

* Check if the string is a valid DNI.

– **Parameters**

* **docType** - String, Document type.

* **docNum** - String, Document number.

* **docLet** - String, Document letter.

– **Returns** - boolean, True if the strings form a valid DNI; False otherwise

- *isValidEmail*

```
public boolean isValidEmail( java.lang.String s )
```

– **Usage**

* Check if the string is an e-mail address.

– **Parameters**

* **s** - String to check.

– **Returns** - boolean, True if the string is a valid e-mail; False otherwise

- *isValidFile*

```
public boolean isValidFile( java.lang.String s )
```

– **Usage**

* Check if the data is a file path

– **Parameters**

* **s** - String to check

– **Returns** - boolean, True if the string is a valid file path; False otherwise

- *isValidFloat*

```
public boolean isValidFloat( java.lang.String s )
```

– **Usage**

* Check if the string is a float numeric.

– **Parameters**

* **s** - String to check.

– **Returns** - boolean, True if the string is a valid float; False otherwise

- *isValidHora*

```
public boolean isValidHora( java.lang.String s )
```

– **Usage**

* Check if the string is a time.

– **Parameters**

* **s** - String to check.

– **Returns** - boolean, True if the string is a valid time; False otherwise

- *isValidNumeric*

```
public boolean isValidNumeric( java.lang.String s )
```

- **Usage**

- * Check if the data is numeric (unsigned integer)

- **Parameters**

- * **s** - String to check

- **Returns** - boolean, True if the string is numeric; False otherwise

- *isValidPattern*

```
public boolean isValidPattern( java.lang.String s, java.lang.String pattern
)
```

- **Usage**

- * Check if the string matches the specified pattern.

- **Parameters**

- * **s** - String to check.

- **Returns** - boolean, True if the string matches the pattern; False otherwise

- *isValidPhone*

```
public boolean isValidPhone( java.lang.String s )
```

- **Usage**

- * Check if the string is a phone number.

- **Parameters**

- * **s** - String to check.

- **Returns** - boolean, True if the string is a valid phone number; False otherwise

- *isValidPostalCode*

```
public boolean isValidPostalCode( java.lang.String s )
```

- **Usage**

- * Check if the string is a postal code.

- **Parameters**

- * **s** - String to check.

- **Returns** - boolean, True if the string is a valid postal code; False otherwise

- *isValidSpanishCif*

```
public boolean isValidSpanishCif( java.lang.String cif )
```

- *isValidSpanishNif*

```
public boolean isValidSpanishNif( java.lang.String nif )
```

- *isValidValueInCollection*

```
public boolean isValidValueInCollection( java.lang.Object val,
java.lang.Object [] col )
```

- **Usage**

- * Check if an object is included in the range of valid objects (Collection of valid objects)

- **Parameters**

- * **val** - Object, object value.
- * **col** - Object[], collection of allowed values.
- **Returns** - boolean, True if val in col; False otherwise

Chapter 22

Package com.imagos.axilice.client

<i>Package Contents</i>	<i>Page</i>
<hr/>	
Classes	
AutoUpdateInitProgressBarOperation	514
<i>...no description...</i>	
BuildEntityViewOperation	514
<i>...no description...</i>	
BuildGuiOperation	515
<i>...no description...</i>	
BuildGUIThread	515
<i>...no description...</i>	
ClientContext	517
<i>This class lays on the client side of the Axilice application and works as an interface for initializing the client generic instances used by the client objects.</i>	
ClientManager	522
<i>Main client class that starts the client side of the application.</i>	
CloseInitializingWindowOperation	523
<i>...no description...</i>	
GetInitProgressBarOperation	523
<i>...no description...</i>	
InitClientOperation	524
<i>...no description...</i>	
UpdateInitProgressBarOperation	526
<i>...no description...</i>	

22.1 Classes

22.1.1 CLASS AutoUpdateInitProgressBarOperation

DECLARATION

```
public class AutoUpdateInitProgressBarOperation
extends java.lang.Object
implements java.lang.Runnable
```

CONSTRUCTORS

- *AutoUpdateInitProgressBarOperation*
`public AutoUpdateInitProgressBarOperation(Display display, ProgressBar progressBar, StatusBar statusBar, java.lang.String statusText)`

METHODS

- *run*
`public void run()`
- *stop*
`public void stop()`

22.1.2 CLASS BuildEntityViewOperation

DECLARATION

```
public class BuildEntityViewOperation
extends java.lang.Object
implements java.lang.Runnable
```

CONSTRUCTORS

- *BuildEntityViewOperation*
`public BuildEntityViewOperation(Display display)`

METHODS

- *run*
`public void run()`

22.1.3 CLASS BuildGuiOperation

DECLARATION

```
public class BuildGuiOperation
extends java.lang.Object
implements java.lang.Runnable
```

CONSTRUCTORS

- *BuildGuiOperation*
public **BuildGuiOperation**(Display display, ProgressBar progressBar, Shell
initWindow, StatusBar statusBar, java.lang.Object builderInstance,
java.lang.String builderMethodName)

METHODS

- *run*
public void **run**()

22.1.4 CLASS BuildGUIThread

DECLARATION

```
public class BuildGUIThread
extends java.lang.Thread
```

CONSTRUCTORS

- *BuildGUIThread*
public **BuildGUIThread**(Display display, ProgressBar progressBar, Shell
initWindow, StatusBar statusBar)

METHODS

- *run*
public void **run**()

METHODS INHERITED FROM CLASS java.lang.Thread

- *activeCount*
public static int activeCount()
- *checkAccess*
public final void checkAccess()
- *countStackFrames*
public native int countStackFrames()
- *currentThread*
public static native Thread currentThread()
- *destroy*
public void destroy()
- *dumpStack*
public static void dumpStack()
- *enumerate*
public static int enumerate(java.lang.Thread [] arg0)
- *getAllStackTraces*
public static Map getAllStackTraces()
- *getContextClassLoader*
public ClassLoader getContextClassLoader()
- *getDefaultUncaughtExceptionHandler*
public static Thread.UncaughtExceptionHandler getDefaultUncaughtExceptionHandler()
- *getId*
public long getId()
- *getName*
public final String getName()
- *getPriority*
public final int getPriority()
- *getStackTrace*
public StackTraceElement getStackTrace()
- *getState*
public Thread.State getState()
- *getThreadGroup*
public final ThreadGroup getThreadGroup()
- *getUncaughtExceptionHandler*
public Thread.UncaughtExceptionHandler getUncaughtExceptionHandler()
- *holdsLock*
public static native boolean holdsLock(java.lang.Object arg0)
- *interrupt*
public void interrupt()
- *interrupted*
public static boolean interrupted()
- *isAlive*
public final native boolean isAlive()
- *isDaemon*
public final boolean isDaemon()
- *isInterrupted*
public boolean isInterrupted()

- *join*
public final void join()
- *join*
public final synchronized void join(long arg0)
- *join*
public final synchronized void join(long arg0, int arg1)
- *resume*
public final void resume()
- *run*
public void run()
- *setContextClassLoader*
public void setContextClassLoader(java.lang.ClassLoader arg0)
- *setDaemon*
public final void setDaemon(boolean arg0)
- *setDefaultUncaughtExceptionHandler*
public static void setDefaultUncaughtExceptionHandler(java.lang.Thread.UncaughtExceptionHandler arg0)
- *setName*
public final void setName(java.lang.String arg0)
- *setPriority*
public final void setPriority(int arg0)
- *setUncaughtExceptionHandler*
public void setUncaughtExceptionHandler(java.lang.Thread.UncaughtExceptionHandler arg0)
- *sleep*
public static native void sleep(long arg0)
- *sleep*
public static void sleep(long arg0, int arg1)
- *start*
public synchronized void start()
- *stop*
public final void stop()
- *stop*
public final synchronized void stop(java.lang.Throwable arg0)
- *suspend*
public final void suspend()
- *toString*
public String toString()
- *yield*
public static native void yield()

22.1.5 CLASS ClientContext

This class lays on the client side of the Axilice application and works as an interface for initializing the client generic instances used by the client objects. It also gives access to the generic properties and systems, as for example, logging, internationalization, etc.
Responsibilities:

Initialize client logging system.

Initialize client internationalization system.

Build the catalog of messages used by the system.

Give accessors for all these general objects.

DECLARATION

```
public class ClientContext
extends java.lang.Object
```

METHODS

-
- *addEntity*

```
public void addEntity( com.imagos.axilice.core.installer.EntityOT  entData )
```

 - **Usage**
 - * Adds an entity to the available entities catalog.
 - **Parameters**
 - * **entData** - EntityOT, data of the entity.

 - *addModuleEntities*

```
public void addModuleEntities( com.imagos.axilice.core.installer.ModuleOT modData )
```

 - **Usage**
 - * Adds the module's entities list to the available entities catalog.
 - **Parameters**
 - * **modData** - ModuleOT, data of the module.

 - *addModuleResultMessages*

```
public void addModuleResultMessages( java.io.InputStream  isResourcesFile )
```

 - **Usage**
 - * Adds a list of result messages defined for an application module to the catalog of result messages of the client application.
 - **Parameters**
 - * **isResourcesFile** - InputStream, stream to the properties file of the module result messages.

 - *buildEntitySelectionViews*

```
public void buildEntitySelectionViews( )
```

 - **Usage**
 - * Builds and initializes the stack of available entity view components.

 - *buildEntityViews*

```
public void buildEntityViews( )
```

 - **Usage**
 - * Builds and initializes the stack of available entity view components.

- *getAreasCatalog*

```
public ArrayList getAreasCatalog( )
```
- *getAvailableEntitySelectionView*

```
public AxiliceEntitySelectionView getAvailableEntitySelectionView( )
```

 - **Usage**
 - * Returns an available instance of an entity view component from the stack.
 - **Returns** - AxiliceEntityView, available instance of entity view component.

- *getAvailableEntityView*

```
public AxiliceEntityView getAvailableEntityView( )
```

 - **Usage**
 - * Returns an available instance of an entity view component from the stack.
 - **Returns** - AxiliceEntityView, available instance of entity view component.

- *getEntities*

```
public HashMap getEntities( )
```

 - **Usage**
 - * Gets the complete catalog of available (installed) entities.
 - **Returns** - HashMap<String, EntityOT>, catalog of available entities.

- *getEntity*

```
public EntityOT getEntity( java.lang.String entId )
```

 - **Usage**
 - * Gets an entity from the entity catalog. It seeks for an entity name among the available (installed) entities list.
 - **Parameters**
 - * entId - String, name of the searched entity.
 - **Returns** - EntityOT, data of the entity.

- *getEntityLauncherFactories*

```
public HashMap getEntityLauncherFactories( )
```

 - **Usage**
 - * Gets the complete catalog of available entity launcher factories.
 - **Returns** - HashMap<String, EntityLauncherFactory>, catalog of available entity launcher factories.

- *getEntityLauncherFactory*

```
public EntityLauncherFactory getEntityLauncherFactory( java.lang.String entId )
```

 - **Usage**
 - * Gets an entity launcher factory from the catalog. It seeks for an entity id among the available entity launcher factories' catalog.
 - **Parameters**
 - * entId - String, key of the searched entity.

- **Returns** - EntityLauncherFactory, factory for building entity launchers.

- *getFakeParent*

public Composite **getFakeParent**()

- **Usage**

- * Returns the fake hidden container for building the entity view components before embedding them in a valid GUI container.

- **Returns** - Composite, fake hidden container.

- *getFakeWindow*

public SwtTopLevelWindow **getFakeWindow**()

- **Usage**

- * Returns the fake hidden window for building the entity selection view components before linking them to a valid window.

- **Returns** - SwtTopLevelWindow, fake hidden window.

- *getI18n*

public I18n **getI18n**()

- **Usage**

- * Gets the internationalization object used by the client objects for translating messages.

- **Returns** - I18n, Axilice client internationalization instance.

- *getInstance*

public static final ClientContext **getInstance**()

- **Usage**

- * Returns the unique instance of the ClientContext class. Implements the singleton design pattern.

- **Returns** - ClientContext, unique instance of ClientContext.

- *getLogger*

public Logger **getLogger**()

- **Usage**

- * Gets the logger instance used by the client objects for system logging messages.

- **Returns** - Logger, Axilice client logger instance.

- *getLogger*

public Logger **getLogger**(java.lang.String category)

- **Usage**

- * Gets the logger instance corresponding to the specified category.

- **Returns** - Logger, category logger instance.

- *getMsgCatalog*

public ResultMessageCatalog **getMsgCatalog**()

- **Usage**

- * Gets the catalog of result and error messages showed to the user as the result of the operations made by the client objects.
 - **Returns** - ResultMessageCatalog, Axilice client messages catalog.

- *getResourceDir*
`public String getResourceDir()`
 - **Usage**
 - * Gets the path to the local resources directory.
 - **Returns** - String, path of the resources directory.

 - *getRevision*
`public static String getRevision()`
 - **Usage**
 - * Returns tag Id assigned to CVS source file.

 - *initClientContext*
`public void initClientContext()`
 - **Usage**
 - * Inits the client context and builds all the generic objects.

 - *initEntityLauncherFactory*
`public void initEntityLauncherFactory(
com.imagos.axilice.core.installer.EntityOT entData, Logger logger)`
 - **Usage**
 - * Creates a new entity launcher factory and insert it into the catalog.
 - **Parameters**
 - * **entData** - EntityOT, data of the entity whose launcher factory will be created.
 - * **logger** - Logger, logger instance used for the entities build by the factory.

 - *isReady*
`public boolean isReady()`
 - **Usage**
 - * Gets the value of the flag isReady.
 - **Returns** - boolean, True if the client context was correctly configured; False in case any error took place during the module configuration.

 - *localizeOperationResult*
`public OperationResult localizeOperationResult(
com.imagos.axilice.core.util.OperationResult res)`
 - **Usage**
 - * Translates a result message contents and localize it into the client user locale.
 - **Parameters**
 - * **resMsg** - ResultMessage, message to translate.
 - **Returns** - ResultMessage, translated message.

- *printEntityLaunchers*
public void **printEntityLaunchers**()
 - *printEntityLaunchers*
public void **printEntityLaunchers**(Logger logger)
 - *releaseEntitySelectionView*
public void **releaseEntitySelectionView**(
com.imagos.axilice.client.gui.AxiliceEntitySelectionView entSelView)
 - *releaseEntityView*
public void **releaseEntityView**(
com.imagos.axilice.client.gui.AxiliceEntityView entView)
 - *setReady*
public void **setReady**(boolean isReady)
- **Usage**
 - * Sets up the value of the flag isReady.
 - **Parameters**
 - * **ready** - boolean, True if the client context was correctly configured; False in case any error took place during the module configuration.

22.1.6 CLASS ClientManager

Main client class that starts the client side of the application.
Responsibilities:

- Configure the client context of the application.
- Init the resources installation process.
- Init the main screen building process.

DECLARATION

```
public class ClientManager
extends java.lang.Object
```

CONSTRUCTORS

- *ClientManager*
public **ClientManager**()

METHODS

- *getRevision*
public static String **getRevision**()
- **Usage**
 - * Returns tag Id assigned to CVS source file.

-
- *main*

```
public static void main( java.lang.String [] args )
```

 - **Parameters**
 * args -

22.1.7 CLASS CloseInitializingWindowOperation

DECLARATION

```
public class CloseInitializingWindowOperation
extends java.lang.Object
implements java.lang.Runnable
```

CONSTRUCTORS

- *CloseInitializingWindowOperation*

```
public CloseInitializingWindowOperation( Shell  initWindow )
```

METHODS

- *run*

```
public void run( )
```

22.1.8 CLASS GetInitProgressBarOperation

DECLARATION

```
public class GetInitProgressBarOperation
extends java.lang.Object
implements java.lang.Runnable
```

CONSTRUCTORS

- *GetInitProgressBarOperation*

```
public GetInitProgressBarOperation( ProgressBar  progressBar )
```

METHODS

- *getCurrentProgress*
public int **getCurrentProgress**()
- *run*
public void **run**()

22.1.9 CLASS InitClientOperation

DECLARATION

```
public class InitClientOperation
extends java.lang.Thread
```

CONSTRUCTORS

- *InitClientOperation*
public InitClientOperation(Display **display**, ProgressBar **progressBar**,
Shell **initWindow**, StatusBar **statusBar**)

METHODS

- *run*
public void **run**()
- *updateProgress*
public void **updateProgress**(int **progressInc**, java.lang.String **statusText**)

METHODS INHERITED FROM CLASS java.lang.Thread

- *activeCount*
public static int **activeCount**()
- *checkAccess*
public final void **checkAccess**()
- *countStackFrames*
public native int **countStackFrames**()
- *currentThread*
public static native Thread **currentThread**()
- *destroy*
public void **destroy**()
- *dumpStack*
public static void **dumpStack**()

- *enumerate*
public static int enumerate(java.lang.Thread [] arg0)
- *getAllStackTraces*
public static Map getAllStackTraces()
- *getContextClassLoader*
public ClassLoader getContextClassLoader()
- *getDefaultUncaughtExceptionHandler*
public static Thread.UncaughtExceptionHandler getDefaultUncaughtExceptionHandler()
- *getId*
public long getId()
- *getName*
public final String getName()
- *getPriority*
public final int getPriority()
- *getStackTrace*
public StackTraceElement getStackTrace()
- *getState*
public Thread.State getState()
- *getThreadGroup*
public final ThreadGroup getThreadGroup()
- *getUncaughtExceptionHandler*
public Thread.UncaughtExceptionHandler getUncaughtExceptionHandler()
- *holdsLock*
public static native boolean holdsLock(java.lang.Object arg0)
- *interrupt*
public void interrupt()
- *interrupted*
public static boolean interrupted()
- *isAlive*
public final native boolean isAlive()
- *isDaemon*
public final boolean isDaemon()
- *isInterrupted*
public boolean isInterrupted()
- *join*
public final void join()
- *join*
public final synchronized void join(long arg0)
- *join*
public final synchronized void join(long arg0, int arg1)
- *resume*
public final void resume()
- *run*
public void run()
- *setContextClassLoader*
public void setContextClassLoader(java.lang.ClassLoader arg0)
- *setDaemon*
public final void setDaemon(boolean arg0)

- *setDefaultUncaughtExceptionHandler*
public static void setDefaultUncaughtExceptionHandler(
java.lang.Thread.UncaughtExceptionHandler arg0)
- *setName*
public final void setName(java.lang.String arg0)
- *setPriority*
public final void setPriority(int arg0)
- *setUncaughtExceptionHandler*
public void setUncaughtExceptionHandler(
java.lang.Thread.UncaughtExceptionHandler arg0)
- *sleep*
public static native void sleep(long arg0)
- *sleep*
public static void sleep(long arg0, int arg1)
- *start*
public synchronized void start()
- *stop*
public final void stop()
- *stop*
public final synchronized void stop(java.lang.Throwable arg0)
- *suspend*
public final void suspend()
- *toString*
public String toString()
- *yield*
public static native void yield()

22.1.10 CLASS UpdateInitProgressBarOperation

DECLARATION

```
public class UpdateInitProgressBarOperation
extends java.lang.Object
implements java.lang.Runnable
```

CONSTRUCTORS

- *UpdateInitProgressBarOperation*
public **UpdateInitProgressBarOperation**(ProgressBar progressBar, int progressInc, StatusBar statusBar, java.lang.String statusText)
- *UpdateInitProgressBarOperation*
public **UpdateInitProgressBarOperation**(ProgressBar progressBar, StatusBar statusBar, java.lang.String statusText)

METHODS

- *run*
public void run()

Chapter 23

Package

com.imagos.axilice.server.management.dao

<i>Package Contents</i>	<i>Page</i>
<hr/>	
Classes	
ManagementDAOno description...	528
<hr/>	

23.1 Classes

23.1.1 CLASS *ManagementDAO*

DECLARATION

```
public class ManagementDAO  
extends java.lang.Object
```

CONSTRUCTORS

- *ManagementDAO*
 public ManagementDAO()

Chapter 24

Package

com.imagos.axilice.core.installer

Package Contents

Page

Interfaces

Installer	531
<i>...no description...</i>	

Classes

AreaOT	531
<i>...no description...</i>	
EntityLauncher	532
<i>Generic class that defines the common operations of the main class of each entity.</i>	
EntityLauncherFactory	541
<i>...no description...</i>	
EntityOT	541
<i>ValueObject with the data of an Entity item.</i>	
FieldOT	545
<i>ValueObject with the data of a Field item.</i>	
ModuleEntityResolver	548
<i>This class implements a SAX Entity Resolver to update the DTD used in the Module Config XML file Parsing to the local version.</i>	
ModuleLauncher	549
<i>Generic class that defines the common operations of the main class of each module.</i>	
ModuleOT	551
<i>ValueObject with the data of a Module item.</i>	
ModuleParser	555
<i>This class implements a SAX Parser to parse an XML module configuration file.</i>	
ResourceException	559
<i>This class encapsulates and holds all the errors that might have taken place during any of the application resources operations.</i>	
ResourceOT	561
<i>...no description...</i>	
SubentityOT	563
<i>ValueObject with the data of a subentity item.</i>	



24.1 Interfaces

24.1.1 INTERFACE Installer

DECLARATION

```
public interface Installer
implements java.rmi.Remote
```

METHODS

- *getAreasList*
public List **getAreasList**()
- *getLatestVersion*
public Integer **getLatestVersion**(java.lang.String codResource)
- *getResource*
public byte **getResource**(java.lang.String codResource)
- *getResourcesList*
public List **getResourcesList**()
- *isUpdated*
public boolean **isUpdated**(java.lang.String codResource, java.lang.Integer currentVersion)

24.2 Classes

24.2.1 CLASS AreaOT

DECLARATION

```
public class AreaOT
extends java.lang.Object
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private String idArea
—
- private String description
—

- private ArrayList lModules

—

- private Integer version

—

CONSTRUCTORS

- *AreaOT*
public AreaOT()

METHODS

- *equals*
public boolean equals(java.lang.Object other)
- *getDescription*
public String getDescription()
- *getIdArea*
public String getIdArea()
- *getLModules*
public ArrayList getLModules()
- *getVersion*
public Integer getVersion()
- *hashCode*
public int hashCode()
- *setDescription*
public void setDescription(java.lang.String description)
- *setIdArea*
public void setIdArea(java.lang.String idArea)
- *setLModules*
public void setLModules(java.util.ArrayList modules)
- *setVersion*
public void setVersion(java.lang.Integer version)

24.2.2 CLASS EntityLauncher

Generic class that defines the common operations of the main class of each entity.
Responsibilities:

- Provide a method for initializing and configuring an entity.
- Provide a method for building the GUI of an entity.
- Provide a method for handling the add entity event.
- Provide a method for handling the delete entity event.
- Provide a method for handling the edit entity event.

- Provide a method for handling the print entity event.
- Provide a method for handling the search entity event.
- Provide a method for handling the select entity event.
- Keep an instance of the logger for the entity.

DECLARATION

```
public abstract class EntityLauncher
extends java.lang.Object
```

FIELDS

- public static final String ERROR_START_ENTITY
 - Constant value with the logging message for an error occurred during the initialization of the entity (the flag entityReady is unset).
- public static final String ERROR_BUILD_ENTITY_PANEL
 - Constant value with the logging message for an error occurred during the building of the entity panel.
- public static final String ENTITY_START
 - Constant value with the logging message for starting a new entity.
- public static final String ERROR_ENTITY_NOHELPERINSTANCE
 - Constant value with the logging message for an error occurred during the instantiation of the entity's helper class.

METHODS

- *addEntity*
 public abstract ArrayList **addEntity**()
 - **Usage**
 - * Add entity button clicked handler. Show the entity content screen ready to start inserting data for a new instance of the entity (new registry).
 - **Returns** - ArrayList<ResultMessage>, list of errors if any has taken place, null in any other case.
- *buildEditEntityGUI*
 public void **buildEditEntityGUI**()
 - **Usage**
 - * Builds the edit entity dialog window from the corresponding glade interface file. Stores the entity dialog instance for posterior accesses.

- *buildEntity*

```
public AxiliceEntityView buildEntity( )
```

- Usage

- * Builds the entity GUI and show it on screen to the user.

- *buildEntity*

```
public AxiliceDetailEntityView buildEntity( SwtWidget parent,
com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData,
java.lang.String rootName )
```

- Usage

- * Builds the entity GUI and show it on screen to the user. The entity GUI will be embedded into the parent composite widget.

- Parameters

- * **parent** - Composite, parent holder for embedding the entity view.
 - * **subentityData** - SubentityOT, data of the subentity to build if the entity launcher will be used to handle a multi-selection detail entity view.
 - * **rootName** - String, name of the root widget that will be embedded into the master entity edition panel.

- **Returns** - AxiliceDetailEntityView, the entity detail view built.

- *cancelEntityEdition*

```
public void cancelEntityEdition( )
```

- *clearEntityData*

```
public abstract void clearEntityData( )
```

- Usage

- * Clears the entity content panel by resetting the content widgets.

- *delEntity*

```
public abstract OperationResult delEntity( )
```

- Usage

- * Delete entity button clicked handler. Deletes the selected instance of the entity (selected registry) from the database.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *delEntity*

```
public abstract OperationResult delEntity( java.util.ArrayList lSelRegistries
)
```

- Usage

- * Delete entity button clicked handler. Deletes the selected instances of the entity (selected registries) from the database. Its a multiple delete method.

- Parameters

- * **lSelRegistries** - ArrayList<Long>, list of selected instances of the entity (selected registries). The list will contain only the internal id of the selected registries.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *deselectSubentity*

```
public void deselectSubentity( java.lang.String  parentFieldId,
com.imagos.axilice.core.common.BaseVO  subentityOT )
```

- **Usage**

- * Updates an entity instance by removing the selected value for a subentity field.

- **Parameters**

- * **parentFieldId** - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
- * **subentityOT** - BaseVO, subentity's deselected instance (registry) value.

- *editEntity*

```
public abstract OperationResult editEntity( )
```

- **Usage**

- * Edit entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), and allows the user to modify any content of the entity registry.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *fillEntityData*

```
public abstract void fillEntityData( )
```

- **Usage**

- * Fills the entity content panel with the data of the selected instance (registry) of the entity.

- *fireExceptionAndExit*

```
public void fireExceptionAndExit( java.lang.String  msg )
```

- **Usage**

- * Fire a exception and stop the program

- **Parameters**

- * **msg** -

- *formatRegistries*

```
public ArrayList formatRegistries( )
```

- *formatRegistries*

```
public ArrayList formatRegistries( java.util.ArrayList  lRegistries )
```

- *getDialog*

```
public SwtDialog getDialog( java.lang.String  dialogName )
```

- *getEntity*

```
public EntityOT getEntity( )
```

- **Usage**

- * Gets the entity data.

– **Returns** - EntityOT, entity data.

• *getEntityData*

public abstract BaseVO getEntityData()

– **Usage**

* Gets the content panel's data typed by the user formatted as an entity VO object.

• *getEntityEditViewName*

public String getEntityEditViewName(java.lang.String parentGuiName)

• *getEntityFields*

public abstract ArrayList getEntityFields()

– **Usage**

* Gets the list of defined fields for an entity

– **Returns** - ArrayList, list of entity fields.

• *getEntityViewName*

public String getEntityViewName(java.lang.String parentGuiName)

• *getLRegistries*

public ArrayList getLRegistries()

– **Usage**

* Getter for LRegistries

• *getRegistry*

public BaseVO getRegistry(long id)

• *getRevision*

public static String getRevision()

– **Usage**

* Returns tag Id assigned to CVS source file.

• *getSelectedEntity*

public BaseVO getSelectedEntity()

– **Usage**

* Gets the selected instance (registry) of the entity.

– **Returns** - BaseVO, selected instance (registry) of the entity.

• *getSelectedId*

public Long getSelectedId()

– **Usage**

* Gets the internal id of the selected entity registry.

– **Returns** - Long, internal id of the selected entity registry, null in case no entity registry has been selected yet.

• *getSwtWidget*

public SwtWidget getSwtWidget(java.lang.String widgetName)

- *getWidget*

```
public Widget getWidget( java.lang.String  widgetName )
```
- *initEntityLauncher*

```
public void initEntityLauncher( Logger  logger, Gui  parentGui, SwtWidget  
parentPanel )
```

 - **Usage**
 - * Inits the entity launcher context and prepares the launcher to be started.
 - **Parameters**
 - * **logger** - Logger, logger instance for printing logging messages.
 - * **parentGuiName** - String, name of the JavaGlade instance that built the parent where the entity will be embedded.
 - * **parentPanel** - Composite, parent panel where the entity contents will be embedded.

- *isDetail*

```
public boolean isDetail( )
```

 - **Usage**
 - * Indicates whether is a main entity or is embedded in another

- *on_entitySelectionButton_clicked*

```
public void on_entitySelectionButton_clicked( Event  e )
```

 - **Usage**
 - * Handler for the subentity selection buttons (magnifying glass buttons) clicked. Shows the selection dialog for selecting a valid instance of a subentity.
 - **Parameters**
 - * **e** - Event, magnifying glass button clicked event data.

- *printEntity*

```
public abstract OperationResult printEntity( )
```

 - **Usage**
 - * Print entity button clicked handler. Builds the report with the data of the selected instance of the entity (selected registry), and allows the user to print the elaborated report.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *printWidgets*

```
public void printWidgets( )
```
- *queryEntity*

```
public abstract ArrayList queryEntity( )
```

 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) available.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.

- *queryEntity*

```
public abstract ArrayList queryEntity( java.lang.String fieldId,
java.lang.String value )
```

- **Usage**

- * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria.

- **Parameters**

- * **fieldId** - String, name of the entity field that will be queried.
 - * **value** - String, value of the entity field that will be searched for.

- **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.

- *queryEntityWithFormat*

```
public abstract ArrayList queryEntityWithFormat( java.lang.String fieldId,
java.lang.String value )
```

- **Usage**

- * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria and show them (formatted as a list) to the user in the entity's screen.

- **Parameters**

- * **fieldId** - String, name of the entity field that will be queried.
 - * **value** - String, value of the entity field that will be searched for.

- **Returns** - ArrayList<RegistryOT>, list of entity instances (registries) taken as the result of the query operation.

- *querySubentities*

```
public abstract ArrayList querySubentities(
com.imagos.axilice.core.installer.SubentityOT entData )
```

- **Usage**

- * Queries the selected entity for all the detail subentities associated with it.

- **Parameters**

- * **entData** - SubentityOT, data of the detail subentity to query for.

- **Returns** - ArrayList, list of detail subentity registries assigned to the master detail registry.

- *saveEntity*

```
public abstract OperationResult saveEntity( )
```

- **Usage**

- * Save entity button clicked handler. Update the system with the data of the edited entity registry. It's called for saving new entity registries (add a new registry) or for updating the contents of existing entity registries.

- **Returns** - OperationResult, Result of the operation and list of messages generated through the operation.

- *saveEntityRegistry*

```
public void saveEntityRegistry( )
```

- *selectEntity*

```
public abstract void selectEntity( java.lang.Long  idEntity )
```

- **Usage**

- * Marks an instance (registry) of the entity as the selected.

- **Parameters**

- * **idEntity** - Long, internal id of the selected instance (registry).

- *selectSubentity*

```
public void selectSubentity( java.lang.String  parentFieldId,  
com.imagos.axilice.core.common.BaseVO  subentityOT )
```

- **Usage**

- * Updates an entity instance by setting up the selected value for a subentity field.

- **Parameters**

- * **parentFieldId** - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.

- * **subentityOT** - BaseVO, subentity's selected instance (registry) value.

- *setEntity*

```
public void setEntity( com.imagos.axilice.core.installer.EntityOT  entity )
```

- **Usage**

- * Sets up the entity name.

- **Parameters**

- * **name** - String, entity name.

- *setRegistries*

```
public void setRegistries( java.util.ArrayList  lRegistries )
```

- **Usage**

- * Sets the list of registries that are showed in the entity view controlled by the Entity Launcher instance. It's used for detail entities where queries are made in the master entity launcher.

- **Parameters**

- * **lRegistries** - ArrayList<BaseVO>, list of shown registries of the entity.

- *showSelectionDialog*

```
public void showSelectionDialog(  
com.imagos.axilice.core.installer.SubentityOT  entData )
```

- **Usage**

- * Opens the selection dialog that enables the user to select an instance of a subentity.

- **Parameters**

- * **entData** - SubentityOT, data of the detail subentity to select.

- *showSubentityDialog*

```
public void showSubentityDialog(
    com.imagos.axilice.core.installer.SubentityOT entData, long selEntity,
    boolean readonly )
```

- **Usage**

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

- **Parameters**

- * **entData** - SubentityOT, data of the detail subentity to select.
 - * **selEntity** - long, id of the selected instance of the subentity.
 - * **readonly** - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.
-

- *showSubentityDialog*

```
public void showSubentityDialog(
    com.imagos.axilice.core.installer.SubentityOT entData, java.lang.String
    parentGuiName, long selEntity, boolean readonly )
```

- **Usage**

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

- **Parameters**

- * **entData** - SubentityOT, data of the detail subentity to select.
 - * **parentGuiName** - String,
 - * **selEntity** - long, id of the selected instance of the subentity.
 - * **readonly** - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.
-

- *showSubentitySelectionDialog*

```
public void showSubentitySelectionDialog(
    com.imagos.axilice.core.installer.SubentityOT entData )
```

- **Usage**

- * Opens the selection dialog that enables the user to select an instance of a multiple selection detail subentity.

- **Parameters**

- * **entData** - SubentityOT, data of the detail subentity to select.
-

- *undoEntityEdition*

```
public void undoEntityEdition( )
```

- *updateEntityData*

```
public abstract void updateEntityData( )
```

- **Usage**

- * Updates the selected instance (registry) data with the content panel's data typed by the user.
-

- *viewEntity*

```
public abstract OperationResult viewEntity( )
```

- **Usage**
 - * View entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), in read-only mode.
- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

24.2.3 CLASS EntityLauncherFactory

DECLARATION

```
public class EntityLauncherFactory
extends java.lang.Object
```

CONSTRUCTORS

- *EntityLauncherFactory*
 public **EntityLauncherFactory**(com.imagos.axilice.core.installer.EntityOT
 entData, Logger logger)

METHODS

- *addEntityLauncher*
 public void **addEntityLauncher**(
 com.imagos.axilice.core.installer.EntityLauncher entLauncher, SwtWidget
 parentWidget)
- *getEntityLauncher*
 public EntityLauncher **getEntityLauncher**(SwtWidget parentWidget)
- *toString*
 public String **toString**()

24.2.4 CLASS EntityOT

ValueObject with the data of an Entity item.
Responsibilities:

- Store the entity data.

DECLARATION

```
public class EntityOT
extends java.lang.Object
```

CONSTRUCTORS

- *EntityOT*

```
public EntityOT( java.lang.String key, java.lang.String name,
java.lang.String icon, java.lang.String idField, java.lang.String
entityHelperClass )
```

- **Usage**

- * EntityOT constructor. It initializes all the entity values.

- **Parameters**

- * **key** - String, entity key.
 - * **name** - String, entity name.
 - * **icon** - String, entity icon.
 - * **idField** - String, entity's identifier field name.
 - * **entityHelperClass** - String, entity helper classname.

- *EntityOT*

```
public EntityOT( java.lang.String key, java.lang.String name,
java.lang.String icon, java.lang.String idField, java.lang.String
entityHelperClass, java.lang.String visible )
```

- **Usage**

- * EntityOT constructor. It initializes all the entity values.

- **Parameters**

- * **key** - String, entity key.
 - * **name** - String, entity name.
 - * **icon** - String, entity icon.
 - * **idField** - String, entity's identifier field name.
 - * **entityHelperClass** - String, entity helper classname.
 - * **visible** - String, entity visibility.

METHODS

- *addField*

```
public void addField( com.imagos.axilice.core.installer.FieldOT field )
```

- **Usage**

- * Adds a new field to the entity fields' list.

- **Parameters**

- * **field** - FieldOT, data of the entity field.

- *getEntityHelperClass*

```
public String getEntityHelperClass( )
```

- **Usage**

- * Gets the entity helper classname.

- **Returns** - String, entity helper classname.

- *getField*

```
public FieldOT getField( java.lang.String fieldId )
```

- **Usage**
 - * Returns the field of the entity.
 - **Parameters**
 - * `fieldId` - String, key of the selected field.
 - **Returns** - FieldOT, field of the entity.

- *getIcon*
`public String getIcon()`
 - **Usage**
 - * Gets the entity icon.
 - **Returns** - String, entity icon.

- *getIdField*
`public String getIdField()`
 - **Usage**
 - * Gets the entity id field.
 - **Returns** - String, entity's identifier field name.

- *getKey*
`public String getKey()`
 - **Usage**
 - * Gets the entity key.
 - **Returns** - String, entity key.

- *getLFields*
`public ArrayList getLFields()`
 - **Usage**
 - * Gets the entity fields' list.
 - **Returns** - ArrayList<FieldOT>, entity list of fields.

- *getName*
`public String getName()`
 - **Usage**
 - * Gets the entity name.
 - **Returns** - String, entity name.

- *getRevision*
`public static String getRevision()`
 - **Usage**
 - * Returns tag Id assigned to CVS source file.

- *isVisible*
`public boolean isVisible()`
 - **Usage**

- * Gets the entity visibility
 - **Returns** - boolean, True if the field is visible; False otherwise.

- *setEntityHelperClass*
 - public void **setEntityHelperClass**(java.lang.String **entityHelperClass**)
 - **Usage**
 - * Sets up the entity helper classname.
 - **Parameters**
 - * **entityHelperClass** - String, entity helper classname.

- *setIcon*
 - public void **setIcon**(java.lang.String **icon**)
 - **Usage**
 - * Sets up the entity icon.
 - **Parameters**
 - * **icon** - String, entity icon.

- *setIdField*
 - public void **setIdField**(java.lang.String **idField**)
 - **Usage**
 - * Sets up the entity id field.
 - **Parameters**
 - * **name** - String, entity's identifier field name.

- *setKey*
 - public void **setKey**(java.lang.String **key**)
 - **Usage**
 - * Sets up the entity key.
 - **Parameters**
 - * **key** - String, entity key.

- *setLFields*
 - public void **setLFields**(java.util.ArrayList **fields**)
 - **Usage**
 - * Sets up the entity fields' list.
 - **Parameters**
 - * **lFields** - ArrayList<FieldOT>, entity list of fields.

- *setName*
 - public void **setName**(java.lang.String **name**)
 - **Usage**
 - * Sets up the entity name.
 - **Parameters**
 - * **name** - String, entity name.

- *setVisible*

```
public void setVisible( boolean visible )
```

- **Usage**

- * Sets up the entity visibility

- **Parameters**

- * **visible** - boolean, True if the field is visible; False otherwise.

24.2.5 CLASS FieldOT

ValueObject with the data of a Field item.
Responsibilities:

- Store the field data.

DECLARATION

```
public class FieldOT
extends java.lang.Object
```

CONSTRUCTORS

- *FieldOT*

```
public FieldOT( java.lang.String id, java.lang.String name, boolean
searchable )
```

- **Usage**

- * FieldOT constructor. It initializes all the field values.

- **Parameters**

- * **id** - String, field id.
 - * **name** - String, field name.
 - * **searchable** - boolean, shows if queries by the field are allowed.

- *FieldOT*

```
public FieldOT( java.lang.String id, java.lang.String name,
java.lang.String searchable )
```

- **Usage**

- * FieldOT constructor. It initializes all the field values.

- **Parameters**

- * **id** - String, field id.
 - * **name** - String, field name.
 - * **searchable** - String, shows if queries by the field are allowed.

- *FieldOT*

```
public FieldOT( java.lang.String id, java.lang.String name,
java.lang.String searchable, java.lang.String visible, java.lang.String
detailEntity, java.lang.String detailField )
```

- **Usage**

- * FieldOT constructor. It initializes all the field values.

- **Parameters**

- * **id** - String, field id.
 - * **name** - String, field name.
 - * **searchable** - String, shows if queries by the field are allowed.
 - * **visible** - String, true if the field is shown in the entity table or false otherwise.
 - * **detailEntity** - String, name of the detail entity for the query.
 - * **detailField** - String, name of the detail entity field for the query.

METHODS

- *getDetailEntity*

```
public String getDetailEntity( )
```

- **Usage**

- * Gets the entity detail entity of the search.

- **Returns** - String, Name of the detail entity to enable the query by; null in case no detail search enabled.

- *getDetailField*

```
public String getDetailField( )
```

- **Usage**

- * Gets the entity detail field of the search.

- **Returns** - String, Name of the detail entity field to enable the query by; null in case no detail search enabled.

- *getId*

```
public String getId( )
```

- **Usage**

- * Gets the field id.

- **Returns** - String, field id.

- *getName*

```
public String getName( )
```

- **Usage**

- * Gets the field name.

- **Returns** - String, field name.

- *getRevision*

```
public static String getRevision( )
```

- **Usage**

- * Returns tag Id assigned to CVS source file.

- *isSearchable*

```
public boolean isSearchable( )
```

- **Usage**
 - * Gets the value of the field searchable flag.
 - **Returns** - boolean, True if queries by the field are allowed; False in any other case.

- *isVisible*
 public boolean isVisible()
 - **Usage**
 - * Gets the entity visibility
 - **Returns** - boolean, True if the field is visible; False otherwise.

- *setDetailEntity*
 public void setDetailEntity(java.lang.String detailEntity)
 - **Usage**
 - * Sets up the entity detail of the search.
 - **Parameters**
 - * detailEntity - String, Name of the detail entity to enable the query by; null in case no detail search enabled.

- *setDetailField*
 public void setDetailField(java.lang.String detailField)
 - **Usage**
 - * Sets up the entity detail field of the search.
 - **Parameters**
 - * detailField - String, Name of the detail entity field to enable the query by; null in case no detail search enabled.

- *setId*
 public void setId(java.lang.String id)
 - **Usage**
 - * Sets up the field id.
 - **Parameters**
 - * id - String, field id.

- *setName*
 public void setName(java.lang.String name)
 - **Usage**
 - * Sets up the field id.
 - **Parameters**
 - * name - String, field name.

- *setSearchable*
 public void setSearchable(boolean searchable)
 - **Usage**
 - * Sets up the value of the field searchable flag.
 - **Parameters**

* **searchable** - boolean, True to allow queries by the field; False in any other case.

- *setVisible*

```
public void setVisible( boolean visible )
```

- **Usage**

- * Sets up the entity visibility

- **Parameters**

- * **visible** - boolean, True if the field is visible; False otherwise.

24.2.6 CLASS ModuleEntityResolver

This class implements a SAX Entity Resolver to update the DTD used in the Module Config XML file Parsing to the local version. That way the application can work even with no Internet connection and the parsing is faster.
Responsibilities:

- Translate the web URL of the DTD to a local URI.

DECLARATION

```
public class ModuleEntityResolver
extends java.lang.Object
implements org.xml.sax.EntityResolver
```

CONSTRUCTORS

- *ModuleEntityResolver*

```
public ModuleEntityResolver( )
```

METHODS

- *getRevision*

```
public static String getRevision( )
```

- **Usage**

- * Returns tag Id assigned to CVS source file.

- *resolveEntity*

```
public InputSource resolveEntity( java.lang.String publicId,
java.lang.String systemId )
```

- **Usage**

- * Allow the application to resolve external entities. Its the implementation of the resolveEntity method of the EntityResolver interface. In this method the DTD external entity will be redirected to its local version.

- **Parameters**

- * **publicId** - String, The public identifier of the external entity being referenced, or null if none was supplied.
- * **systemId** - String, The system identifier of the external entity being referenced.
- **Returns** - InputSource An InputSource object describing the new input source, or null to request that the parser open a regular URI connection to the system identifier.

24.2.7 CLASS ModuleLauncher

Generic class that defines the common operations of the main class of each module.
Responsibilities:

- Provide a method for initializing and configuring a module.
- Provide a method for starting a module.
- Store the catalog of messages resources for each module.
- Keep an instance of the logger for the module.
- Keep an instance of the internationalization object for the module.

DECLARATION

```
public class ModuleLauncher
extends java.lang.Object
```

FIELDS

- public static final String ERROR_CONFIG_MOD
 - Constant value with the logging message for an error occurred during the loading of the configuration file of the module.
- public static final String ERROR_CONFIG_LOG
 - Constant value with the logging message for an error occurred during the loading of the logging configuration file of the module.
- public static final String ERROR_CONFIG_MSG
 - Constant value with the logging message for an error occurred during the loading of the result messages properties file of the module.
- public static final String ERROR_START_MOD
 - Constant value with the logging message for an error occurred during the initialization of the module (the flag moduleReady is unset).
- public static final String ERROR_BUILD_MOD_PANEL
 - Constant value with the logging message for an error occurred during the building of the module panel.
- public static final String ERROR_START_NOMAINCLASS
 - Constant value with the logging message for an error occurred during the loading of the module's main class.

- public static final String ERROR_START_NOMAINMETHOD
 - Constant value with the logging message for an error occurred during the location of the module's starting method.
- public static final String ERROR_START_NOMAININSTANCE
 - Constant value with the logging message for an error occurred during the instantiation of the module's main class.
- public static final String MODULE_START
 - Constant value with the logging message for starting a new module.
- public static final String ENTITY_UNDEFINED
 - Constant value with the logging message for reporting an error searching for an undefined entity.
- public static final String ERROR_NO_ENTITIES_DEFINED
 - Constant value with the logging message for reporting an error building the catalog of entities for the module. No entities have been defined.

CONSTRUCTORS

- *ModuleLauncher*
 public **ModuleLauncher**(java.io.InputStream is)
 - **Usage**
 - * Constructor of ModuleLauncher class.
 - **Parameters**
 - * is - InputStream, reading stream to the module configuration file.

METHODS

- *getLogger*
 public **Logger** **getLogger**()
 - **Usage**
 - * Gets the logger for the module.
 - **Returns** - Logger, the logger for the module.
- *getModuleData*
 public **ModuleOT** **getModuleData**()
 - **Usage**
 - * Gets the configuration of the module.
 - **Returns** - ModuleOT, module configuration data.
- *getRevision*
 public static **String** **getRevision**()
 - **Usage**

* Returns tag Id assigned to CVS source file.

- *initModule*

public void initModule()

– **Usage**

* Initialize and configure the module. As the result of this operation the module launcher will be ready to start the module when it's requested.

- *isModuleReady*

public boolean isModuleReady()

– **Usage**

* Gets the value of the flag moduleReady.

– **Returns** - boolean, True if the module was correctly configured and is ready to be started; False in case any error took place during the module configuration.

- *setModuleReady*

public void setModuleReady(boolean moduleReady)

– **Usage**

* Sets up the value of the flag moduleReady.

– **Parameters**

* **moduleReady** - boolean, True if the module was correctly configured and is ready to be started; False in case any error took place during the module configuration.

- *startModule*

public void startModule()

– **Usage**

* Inits the execution of the module. Shows the main screen of the module.

24.2.8 CLASS ModuleOT

ValueObject with the data of a Module item.
Responsibilities:

- Store the module data.

DECLARATION

<pre>public class ModuleOT extends java.lang.Object</pre>

CONSTRUCTORS

• *ModuleOT*

```
public ModuleOT( java.lang.String name, java.lang.String mainClass,
java.lang.String mainMethod, java.lang.String icon, java.lang.String
logCategory, java.lang.String logConfigFile, java.lang.String msgPropFile
)
```

– **Usage**

* ModuleOT constructor. It initializes all the module values.

– **Parameters**

* **name** - String, module name.
 * **mainClass** - String, module main class.
 * **mainMethod** - String, module main method.
 * **icon** - String, module icon.
 * **logCategory** - String, module log category.
 * **logConfigFile** - String, module log configuration filename.
 * **msgPropFile** - String, module message properties filename.

METHODS

• *addEntity*

```
public void addEntity( com.imagos.axilice.core.installer.EntityOT entity )
```

– **Usage**

* Adds a new entity to the module entities' list.

– **Parameters**

* **entity** - EntityOT, data of the module entity.

• *getIcon*

```
public String getIcon( )
```

– **Usage**

* Gets the code of the icon resource for the module.

– **Returns** - String, the icon of the module.• *getLEntities*

```
public ArrayList getLEntities( )
```

– **Usage**

* Gets the module entities' list.

– **Returns** - ArrayList<EntityOT>, module list of entities.• *getLogCategory*

```
public String getLogCategory( )
```

– **Usage**

* Gets the log category used for login the processes for the module.

– **Returns** - String, log category of the module.

- *getLogConfigFile*

public String getLogConfigFile()

- **Usage**

- * Gets the path of the log configuration path for the module.

- **Returns** - String, path of the log configuration file of the module.

- *getMainClass*

public String getMainClass()

- **Usage**

- * Gets the module's main classname, that is, the class that implements the method that starts the execution of the module.

- **Returns** - String, the main classname of the module.

- *getMainMethod*

public String getMainMethod()

- **Usage**

- * Gets the module's main method, that is, the name of the method that starts the execution of the module.

- **Returns** - String, the main method's name of the module.

- *getMsgPropFile*

public String getMsgPropFile()

- **Usage**

- * Gets the messages properties file name of the module.

- **Returns** - String, messages properties file name.

- *getName*

public String getName()

- **Usage**

- * Gets the name of the module.

- **Returns** - String, the name of the module.

- *getRevision*

public static String getRevision()

- **Usage**

- * Returns tag Id assigned to CVS source file.

- *setIcon*

public void setIcon(java.lang.String moduleIcon)

- **Usage**

- * Sets up the code of the icon resource for the module.

- **Parameters**

- * moduleIcon - String, the icon of the module.

- *setLEntities*

```
public void setLEntities( java.util.ArrayList entities )
```

- **Usage**

- * Sets up the module entities' list.

- **Parameters**

- * lEntities - ArrayList<EntityOT>, module list of entities.

- *setLogCategory*

```
public void setLogCategory( java.lang.String logCategory )
```

- **Usage**

- * Sets up the log category used for login the processes for the module.

- **Parameters**

- * logCategory - String, log category of the module.

- *setLogConfigFile*

```
public void setLogConfigFile( java.lang.String logConfigFile )
```

- **Usage**

- * Sets up the path of the log configuration path for the module.

- **Parameters**

- * logConfigFile - String, path of the log configuration file of the module.

- *setMainClass*

```
public void setMainClass( java.lang.String moduleMainClass )
```

- **Usage**

- * Sets up the module's main classname, that is, the class that implements the method that starts the execution of the module.

- **Parameters**

- * moduleMainClass - String, the main classname of the module.

- *setMainMethod*

```
public void setMainMethod( java.lang.String moduleMainMethod )
```

- **Usage**

- * Sets up the module's main method, that is, the name of the method that starts the execution of the module.

- **Parameters**

- * moduleMainMethod - String, the main method's name of the module.

- *setMsgPropFile*

```
public void setMsgPropFile( java.lang.String msgPropertiesFile )
```

- **Usage**

- * Sets up the messages properties file name of the module.

- **Parameters**

- * msgPropertiesFile - String, messages properties file name.

- *setName*
`public void setName(java.lang.String moduleName)`
 - **Usage**
 - * Sets up the name of the module.
 - **Parameters**
 - * `moduleName` - String, the name of the module.

- *toString*
`public String toString()`
 - **Usage**
 - * Implementation of the toString method to format the data of the module.
 - **Returns** - String, formatted module data for printing them out.

24.2.9 CLASS ModuleParser

This class implements a SAX Parser to parse an XML module configuration file.
Responsibilities:

- Parse XML module configuration files.
- Validate XML module configuration files.

DECLARATION

```
public class ModuleParser
extends org.xml.sax.helpers.DefaultHandler
```

CONSTRUCTORS

- *ModuleParser*
`public ModuleParser(java.io.InputStream is)`
 - **Usage**
 - * Parser constructor. Creates a new instance of the parser and parses the file specified.
 - **Parameters**
 - * `is` - InputStream, input stream of the module configuration file that will be parsed.

- *ModuleParser*
`public ModuleParser(java.lang.String aPathFile)`
 - **Usage**
 - * Parser constructor. Creates a new instance of the parser and parses the file specified.
 - **Parameters**
 - * `aPathFile` - String, path of the module configuration file that will be parsed.

METHODS

• *appendContent*

```
public void appendContent( char [] ch, int start, int length )
```

– **Usage**

* Updates the content of a PCDATA XML element with the parsed characters.

– **Parameters**

* **ch** - char[], set of characters that is being parsed.
 * **start** - int, starting index of the first character parsed.
 * **length** - int, number of characters parsed.

• *characters*

```
public void characters( char [] ch, int start, int length )
```

– **Usage**

* Receive notification of character data inside an element.

– **Parameters**

* **ch** - char[], The characters.
 * **start** - int, The start position in the character array.
 * **length** - int, The number of characters to use from the character array.

– **Exceptions**

* **SAXException**, - if there is a problem processing the glade file (such as incorrect file format, format not compliant with the schema, unknown tag appeared, etc.).

• *endElement*

```
public void endElement( java.lang.String uri, java.lang.String  
localName, java.lang.String qName )
```

– **Usage**

* Receive notification of the end of an element.

– **Parameters**

* **uri** - String,
 * **localName** - String, the local name (without prefix), or the empty string if Namespace processing is not being performed.
 * **qName** - String, the qualified name (with prefix), or the empty string if qualified names are not available.

– **Exceptions**

* **SAXException**, - if there is a problem processing the glade file (such as incorrect file format, format not compliant with the schema, unknown tag appeared, etc.).

• *error*

```
public void error( org.xml.sax.SAXParseException e )
```

– **Usage**

* Receive notification of a recoverable parser error.

– **Parameters**

* **e** - SAXParseException, The warning information encoded as an exception.

• *fatalError*

```
public void fatalError( org.xml.sax.SAXParseException e )
```

- **Usage**
 - * Report a fatal XML parsing error.
 - **Parameters**
 - * **e** - SAXParseException, The warning information encoded as an exception.
-
- *getContent*
`public String getContent()`
 - **Usage**
 - * Gets the PCDATA XML element content.
 - **Returns** - String, parsed content.
-
- *getModule*
`public ModuleOT getModule()`
 - **Usage**
 - * Gets the final result of the module configuration XML parsed file.
 - **Returns** - ModuleOT, all the configuration info loaded for the module.
-
- *getRevision*
`public static String getRevision()`
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
-
- *setContent*
`public void setContent(java.lang.String content)`
 - **Usage**
 - * Sets up the PCDATA XML element content.
 - **Parameters**
 - * **content** - String, parsed content.
-
- *startDocument*
`public void startDocument()`
 - **Usage**
 - * Receive notification of the start of the module configuration file to parse.
 - **Exceptions**
 - * **SAXException**, - if there is a problem processing the configuration file (such as incorrect file format, format not compliant with the schema, unknown tag appeared, etc.).
-
- *startElement*
`public void startElement(java.lang.String uri, java.lang.String localName, java.lang.String qName, org.xml.sax.Attributes attributes)`
 - **Usage**
 - * Receive notification of the start of an element.
 - **Parameters**
 - * **uri** - String,

- * **localName** - String, the local name (without prefix), or the empty string if Namespace processing is not being performed.
 - * **qName** - String, the qualified name (with prefix), or the empty string if qualified names are not available.
 - * **attributes** - Attributes, the specified or defaulted attributes.
 - **Exceptions**
 - * **SAXException**, - if there is a problem processing the glade file (such as incorrect file format, format not compliant with the schema, unknown tag appeared, etc.).
-
- *warning*
 public void **warning**(org.xml.sax.SAXParseException e)
 - **Usage**
 - * Receive notification of a parser warning.
 - **Parameters**
 - * **e** - SAXParseException, The warning information encoded as an exception.

METHODS INHERITED FROM CLASS org.xml.sax.helpers.DefaultHandler

- *characters*
 public void **characters**(char [] arg0, int arg1, int arg2)
- *endDocument*
 public void **endDocument**()
- *endElement*
 public void **endElement**(java.lang.String arg0, java.lang.String arg1, java.lang.String arg2)
- *endPrefixMapping*
 public void **endPrefixMapping**(java.lang.String arg0)
- *error*
 public void **error**(org.xml.sax.SAXParseException arg0)
- *fatalError*
 public void **fatalError**(org.xml.sax.SAXParseException arg0)
- *ignorableWhitespace*
 public void **ignorableWhitespace**(char [] arg0, int arg1, int arg2)
- *notationDecl*
 public void **notationDecl**(java.lang.String arg0, java.lang.String arg1, java.lang.String arg2)
- *processingInstruction*
 public void **processingInstruction**(java.lang.String arg0, java.lang.String arg1)
- *resolveEntity*
 public InputSource **resolveEntity**(java.lang.String arg0, java.lang.String arg1)
- *setDocumentLocator*
 public void **setDocumentLocator**(org.xml.sax.Locator arg0)
- *skippedEntity*
 public void **skippedEntity**(java.lang.String arg0)
- *startDocument*
 public void **startDocument**()
- *startElement*
 public void **startElement**(java.lang.String arg0, java.lang.String arg1, java.lang.String arg2, org.xml.sax.Attributes arg3)

- *startPrefixMapping*
public void startPrefixMapping(java.lang.String arg0, java.lang.String arg1)
- *unparsedEntityDecl*
public void unparsedEntityDecl(java.lang.String arg0, java.lang.String arg1, java.lang.String arg2, java.lang.String arg3)
- *warning*
public void warning(org.xml.sax.SAXParseException arg0)

24.2.10 CLASS ResourceException

This class encapsulates and holds all the errors that might have taken place during any of the application resources operations.
Responsibilities:

Encapsulate and pass on the error messages to the upper layers.

DECLARATION

```
public class ResourceException
extends java.lang.Exception
```

SERIALIZABLE FIELDS

- private String codResource
 - Code of the resource involved in the operation that originated the error.
- private String pathResource
 - Path of the resource involved in the operation that originated the error.
- private int errCode
 - Internal error code to support error localization and classification.
- private String errorMessage
 - Error message string that is showed to the end user.

CONSTRUCTORS

- *ResourceException*
public ResourceException(int errCode)
- *ResourceException*
public ResourceException(int errCode, java.lang.Throwable cause)
- *ResourceException*
public ResourceException(java.lang.String codResource, int errCode)

- *ResourceException*
public ResourceException(java.lang.String codResource, int errCode, java.lang.Throwable cause)
- *ResourceException*
public ResourceException(java.lang.String codResource, java.lang.String pathResource, int errCode)
- *ResourceException*
public ResourceException(java.lang.String codResource, java.lang.String pathResource, int errCode, java.lang.Throwable cause)

METHODS

- *getCodResource*
public String getCodResource()
- *getErrCode*
public int getErrCode()
- *getErrMsg*
public String getErrMsg()
- *getMessage*
public String getMessage()
- *getPathResource*
public String getPathResource()
- *getRevision*
public static String getRevision()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
- *setCodResource*
public void setCodResource(java.lang.String codResource)
- *setErrCode*
public void setErrCode(int errCode)
- *setErrMsg*
public void setErrMsg(java.lang.String errMsg)
- *setPathResource*
public void setPathResource(java.lang.String pathResource)
- *toString*
public String toString()

METHODS INHERITED FROM CLASS java.lang.Exception

METHODS INHERITED FROM CLASS java.lang.Throwable

- *fillInStackTrace*
public synchronized native Throwable fillInStackTrace()
- *getCause*
public Throwable getCause()
- *getLocalizedMessage*
public String getLocalizedMessage()
- *getMessage*
public String getMessage()
- *getStackTrace*
public StackTraceElement getStackTrace()
- *initCause*
public synchronized Throwable initCause(java.lang.Throwable arg0)
- *printStackTrace*
public void printStackTrace()
- *printStackTrace*
public void printStackTrace(java.io.PrintStream arg0)
- *printStackTrace*
public void printStackTrace(java.io.PrintWriter arg0)
- *setStackTrace*
public void setStackTrace(java.lang.StackTraceElement [] arg0)
- *toString*
public String toString()

24.2.11 CLASS ResourceOT

DECLARATION

```
public class ResourceOT
extends java.lang.Object
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idResource
—
- private String codResource
—
- private String description
—
- private String path
—

- private String idArea
—
- private Integer version
—

CONSTRUCTORS

- *ResourceOT*
public **ResourceOT**()

METHODS

- *equals*
public boolean equals(java.lang.Object other)
- *getCodResource*
public String getCodResource()
- *getDescription*
public String getDescription()
- *getIdArea*
public String getIdArea()
- *getIdResource*
public long getIdResource()
- *getPath*
public String getPath()
- *getVersion*
public Integer getVersion()
- *hashCode*
public int hashCode()
- *setCodResource*
public void setCodResource(java.lang.String codResource)
- *setDescription*
public void setDescription(java.lang.String description)
- *setIdArea*
public void setIdArea(java.lang.String idArea)
- *setIdResource*
public void setIdResource(long idResource)
- *setPath*
public void setPath(java.lang.String path)
- *setVersion*
public void setVersion(java.lang.Integer version)

24.2.12 CLASS SubentityOT

ValueObject with the data of a subentity item. It holds the subentity information for building the selection dialog (EntityOT instance), plus the data of the parent entity for returning the selected data.

Responsibilities:

- Store the subentity data.

DECLARATION

```
public class SubentityOT
extends java.lang.Object
```

CONSTRUCTORS

- *SubentityOT*
public SubentityOT()
- *SubentityOT*
**public SubentityOT(com.imagos.axilice.core.installer.EntityOT
subentityData, java.lang.String parentFieldId,
com.imagos.axilice.core.installer.EntityLauncher parentEntity)**
 - **Usage**
 - * Default constructor.
 - **Parameters**
 - * **subentityData** - EntityOT, subentity data.
 - * **parentFieldId** - String, parent entity field name linked to the subentity.
 - * **parentEntity** - EntityLauncher, parent entity launcher.

METHODS

- *getParentEntity*
public EntityLauncher getParentEntity()
 - **Usage**
 - * Gets the parent entity launcher that will manage the events of the subentity selection process.
 - **Returns** - EntityLauncher, parent entity launcher.
 - *getParentFieldId*
public String getParentFieldId()
 - **Usage**
 - * Gets the parent entity field whose value is the selected subentity.
 - **Returns** - String, parent entity field id.
-

- *getRevision*

```
public static String getRevision( )
```

- **Usage**

- * Returns tag Id assigned to CVS source file.

- *getSubentityData*

```
public EntityOT getSubentityData( )
```

- **Usage**

- * Gets the subentity data.

- **Returns** - EntityOT, subentity data.

- *setParentEntity*

```
public void setParentEntity(  
com.imagos.axilice.core.installer.EntityLauncher  parentEntity )
```

- **Usage**

- * Sets up the parent entity launcher that will manage the events of the subentity selection process.

- **Parameters**

- * parentEntity - EntityLauncher, parent entity launcher.

- *setParentFieldId*

```
public void setParentFieldId( java.lang.String  parentFieldId )
```

- **Usage**

- * Sets up the parent entity field whose value is the selected subentity.

- **Parameters**

- * parentFieldId - String, parent entity field id.

- *setSubentityData*

```
public void setSubentityData( com.imagos.axilice.core.installer.EntityOT  
subentityData )
```

- **Usage**

- * Sets up the subentity data.

- **Parameters**

- * subentityData - EntityOT, subentity data.

Chapter 25

Package com.imagos.axilice.server.traceability

<i>Package Contents</i>	<i>Page</i>
<hr/>	
Classes	
TracerImpl.....	566
...no description...	
<hr/>	

25.1 Classes

25.1.1 CLASS TracerImpl

DECLARATION

```
public class TracerImpl
extends java.rmi.server.UnicastRemoteObject
implements com.imagos.axilice.core.traceability.Tracer
```

CONSTRUCTORS

- *TracerImpl*
public **TracerImpl**()

METHODS

- *buildTraceabilityTree*
public TraceabilityTree **buildTraceabilityTree**(byte [] xmlContent)

METHODS INHERITED FROM CLASS java.rmi.server.UnicastRemoteObject

- *clone*
public Object clone()
- *exportObject*
public static RemoteStub exportObject(java.rmi.Remote arg0)
- *exportObject*
public static Remote exportObject(java.rmi.Remote arg0, int arg1)
- *exportObject*
public static Remote exportObject(java.rmi.Remote arg0, int arg1, java.rmi.server.RMIClientSocketFactory arg2, java.rmi.server.RMIServerSocketFactory arg3)
- *unexportObject*
public static boolean **unexportObject**(java.rmi.Remote arg0, boolean arg1)

METHODS INHERITED FROM CLASS java.rmi.server.RemoteServer

- *getClientHost*
public static String getClientHost()
- *getLog*
public static PrintStream getLog()
- *setLog*
public static void **setLog**(java.io.OutputStream arg0)

METHODS INHERITED FROM CLASS java.rmi.server.RemoteObject

- *equals*
public boolean equals(java.lang.Object arg0)
- *getRef*
public RemoteRef getRef()
- *hashCode*
public int hashCode()
- *toString*
public String toString()
- *toStub*
public static Remote toStub(java.rmi.Remote arg0)

Chapter 26

Package com.imagos.axilice.server.installer

<i>Package Contents</i>	<i>Page</i>
<hr/>	
Classes	
InstallerImpl	569
<i>...no description...</i>	
ResourceDAO	570
<i>This class implements the application resources data access object.</i>	
ResourceLN	572
<i>This class implements the resources catalog model.</i>	
UpdateManager	574
<i>...no description...</i>	
<hr/>	

26.1 Classes

26.1.1 CLASS *InstallerImpl*

DECLARATION

```
public class InstallerImpl
extends java.rmi.server.UnicastRemoteObject
implements com.imagos.axilice.core.installer.Installer
```

CONSTRUCTORS

- *InstallerImpl*
public **InstallerImpl**()

METHODS

- *getAreasList*
public List **getAreasList**()
- *getLatestVersion*
public Integer **getLatestVersion**(java.lang.String **codResource**)
- *getResource*
public byte **getResource**(java.lang.String **codResource**)
- *getResourcesList*
public List **getResourcesList**()
- *isUpdated*
public boolean **isUpdated**(java.lang.String **codResource**, java.lang.Integer **currentVersion**)

METHODS INHERITED FROM CLASS java.rmi.server.UnicastRemoteObject

- *clone*
public Object **clone**()
- *exportObject*
public static RemoteStub **exportObject**(java.rmi.Remote **arg0**)
- *exportObject*
public static Remote **exportObject**(java.rmi.Remote **arg0**, int **arg1**)
- *exportObject*
public static Remote **exportObject**(java.rmi.Remote **arg0**, int **arg1**, java.rmi.server.RMIClientSocketFactory **arg2**, java.rmi.server.RMIServerSocketFactory **arg3**)
- *unexportObject*
public static boolean **unexportObject**(java.rmi.Remote **arg0**, boolean **arg1**)

METHODS INHERITED FROM CLASS java.rmi.server.RemoteServer

- *getClientHost*
public static String getClientHost()
- *getLog*
public static PrintStream getLog()
- *setLog*
public static void setLog(java.io.OutputStream arg0)

METHODS INHERITED FROM CLASS java.rmi.server.RemoteObject

- *equals*
public boolean equals(java.lang.Object arg0)
- *getRef*
public RemoteRef getRef()
- *hashCode*
public int hashCode()
- *toString*
public String toString()
- *toStub*
public static Remote toStub(java.rmi.Remote arg0)

26.1.2 CLASS ResourceDAO

This class implements the application resources data access object. It's a part of the data access layer and gives access to the database data. It knows the database structure and implementation.
This class is implemented to access a PostgreSQL database implementation.
Responsibilities:

Provide statements for inserting new application resources into the database.

Provide statements for updating application resources into the database.

Provide statements for deleting application resources from the database.

Provide the database queries needed for application resources.

DECLARATION

```
public class ResourceDAO
extends java.lang.Object
```

CONSTRUCTORS

- *ResourceDAO*
public ResourceDAO()

METHODS

• *getAllAreas*

```
public List getAllAreas( )
```

– **Usage**

* Return the areas' list including the list of all the modules belonging to each area.

– **Returns** - List, list of areas.– **Exceptions**

* com.imagos.axilice.core.installer.ResourceException - in case any error take place during the query operation.

• *getAllResources*

```
public List getAllResources( )
```

– **Usage**

* Query the database to get all the application resources available.

– **Returns** - ArrayList, list of all the application resources.– **Exceptions**

* ResourceException, - in case any error take place during the query operation.

• *getAreaModules*

```
public List getAreaModules( java.lang.String idArea )
```

• *getResource*

```
public ResourceOT getResource( long idResource )
```

– **Usage**

* Query (by primary key) the database to get an application resource data.

– **Parameters**

* idResource - long, Internal id of the application resource.

– **Returns** - ResourceOT, application resource if it exists, null in any other case.– **Exceptions**

* ResourceException, - in case any error take place during the query operation.

• *getResource*

```
public ResourceOT getResource( java.lang.String codResource )
```

– **Usage**

* Query the database to get an application resource data.

– **Parameters**

* codResource - String, External code of the application resource.

– **Returns** - ResourceOT, application resource if it exists, null in any other case.– **Exceptions**

* ResourceException, - in case any error take place during the query operation.

• *getRevision*

```
public static String getRevision( )
```

– **Usage**

* Returns tag Id assigned to CVS source file.

26.1.3 CLASS ResourceLN

This class implements the resources catalog model. It's a part of the business logic layer and encapsulates the access to the resources tables, providing methods to query, update, insert or delete resources.

Responsibilities:

Provide access to resources metadata (path, version).

Query the database by the primary key or by any other query criteria for resources.

Insert new resources into the catalog.

Update resources catalog (paths and versions).

Delete resources from the catalog.

DECLARATION

```
public class ResourceLN
extends java.lang.Object
```

METHODS

- *getAllAreas*

```
public List getAllAreas( )
```

 - **Usage**
 - * Return the areas' list including the list of all the modules belonging to each area.
 - **Returns** - List, list of areas.
 - **Exceptions**
 - * com.imagos.axilice.core.installer.ResourceException - in case any error take place during the query operation.

- *getAllResources*

```
public List getAllResources( )
```

 - **Usage**
 - * Return the resources' list including the info of all the resources stored in the catalog.
 - **Returns** - List, list of resources.
 - **Exceptions**
 - * com.imagos.axilice.core.installer.ResourceException - in case any error take place during the query operation.

- *getInstance*

```
public static ResourceLN getInstance( )
```

- **Usage**
 - * Returns the Singleton instance of class ResourceLN.
 - **Returns** - ResourceLN, the Singleton instance of ResourceLN.
-
- *getLatestVersion*
 public Integer **getLatestVersion**(java.lang.String codResource)
 - **Usage**
 - * Get the latest version available of the resource.
 - **Parameters**
 - * codResource - String, external identification code of the resource.
 - **Returns** - Integer, resource's latest version available.
 - **Exceptions**
 - * com.imagos.axilice.core.installer.ResourceException - in case any error take place during the query operation.
-
- *getResource*
 public ResourceOT **getResource**(long idResource)
 - **Usage**
 - * Return the resource metadata.
 - **Parameters**
 - * idResource - long, internal id of the resource.
 - **Returns** - ResourceOT, resource metadata.
 - **Exceptions**
 - * com.imagos.axilice.core.installer.ResourceException - in case any error take place during the query operation.
-
- *getResource*
 public ResourceOT **getResource**(java.lang.String codResource)
 - **Usage**
 - * Return the resource metadata.
 - **Parameters**
 - * codResource - String, external identification code of the resource.
 - **Returns** - ResourceOT, resource metadata.
 - **Exceptions**
 - * com.imagos.axilice.core.installer.ResourceException - in case any error take place during the query operation.
-
- *getRevision*
 public static String **getRevision**()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
-
- *isResourceUpdated*
 public boolean **isResourceUpdated**(java.lang.String codResource, java.lang.Integer currentVersion)

- **Usage**

- * Checks if the resource available at the client is updated to the last version.

- **Parameters**

- * **codResource** - String, external identification code of the resource.
 - * **Integer** - currentVersion, current version of the resource that is available at the client.

- **Returns** - boolean, True in case the resource is up to date; False in any other case.

- **Exceptions**

- * **com.imagos.axilice.core.installer.ResourceException** - in case any error take place during the query operation.

- *loadResource*

```
public byte loadResource( java.lang.String  codResource )
```

- **Usage**

- * Load the resource stored in the file pointed by resource's path. It returns the bytes that the resource is made up of.

- **Parameters**

- * **codResource** - String, external identification code of the resource.

- **Returns** - byte[], resource data.

- **Exceptions**

- * **com.imagos.axilice.core.installer.ResourceException** - in case any error take place during the query operation.

26.1.4 CLASS UpdateManager

DECLARATION

```
public class UpdateManager
extends java.lang.Object
```

METHODS

- *getInstance*

```
public static UpdateManager getInstance( )
```

- **Usage**

- * Gets an instance of the UpdateManager class.

- **Returns** - UpdateManager, the unique instance of the UpdateManager class (Singleton pattern).

- *initInstaller*

```
public void initInstaller( )
```

- **Usage**

- * Initializes the installer remote object. Makes the registration and binding of the remote Installer service.

Chapter 27

Package com.imagos.axilice.client.partners

<i>Package Contents</i>	<i>Page</i>
<hr/>	
Classes	
AddressClient <i>...no description...</i>	576
AddressListener <i>...no description...</i>	589
BankAccountClient <i>...no description...</i>	590
ContactClient <i>...no description...</i>	602
DepartmentClient <i>...no description...</i>	615
DiscountClient <i>...no description...</i>	628
PartnerClient <i>...no description...</i>	641
PartnerGroupClient <i>...no description...</i>	654
PaymentMethodClient <i>...no description...</i>	666
RappelIntervalClient <i>...no description...</i>	679
RappelTypeClient <i>...no description...</i>	692
RappelTypeListener <i>...no description...</i>	704
VendorClient <i>...no description...</i>	705

27.1 Classes

27.1.1 CLASS AddressClient

DECLARATION

```
public class AddressClient
extends com.imagos.axilice.client.gui.common.CommonClient
```

CONSTRUCTORS

- *AddressClient*
public **AddressClient**()

METHODS

- *build_extension_fields_view*
public void **build_extension_fields_view**()
- *buildSelectionSets*
public void **buildSelectionSets**()
- *fillEntityData*
public void **fillEntityData**()
- *getComboAddressCountry*
public Combo **getComboAddressCountry**()
- *getComboAddressState*
public Combo **getComboAddressState**()
- *getCountries*
public ArrayList **getCountries**()
- *getEntityData*
public BaseVO **getEntityData**()
- *getStates*
public ArrayList **getStates**()
- *getWidgets*
public void **getWidgets**()
- *init_specific_components*
public void **init_specific_components**()
- *loadCountries*
public void **loadCountries**()
- *loadStates*
public void **loadStates**(com.imagos.axilice.core.common.CountryOT country)

- *printEntity*
public OperationResult **printEntity**()
- *setComboAddressCountry*
public void **setComboAddressCountry**(Combo comboAddressCountry)
- *setComboAddressState*
public void **setComboAddressState**(Combo comboAddressState)
- *setCountries*
public void **setCountries**(java.util.ArrayList countries)
- *setEditionMode*
public void **setEditionMode**(boolean enabled)
- *setStates*
public void **setStates**(java.util.ArrayList states)
- *setWidgetsValues*
public void **setWidgetsValues**(com.imagos.axilice.core.common.BaseVO entity)
- *validateEntity*
public OperationResult **validateEntity**()

METHODS INHERITED FROM CLASS com.imagos.axilice.client.gui.common.CommonClient

(in 17.1.2, page 402)

- *addAxiliceEventListener*
public void **addAxiliceEventListener**(
com.imagos.axilice.client.gui.AxiliceEventListener l)
- *addEntity*
public ArrayList **addEntity**()
- *addExistingSubentity*
public void **addExistingSubentity**(com.imagos.axilice.core.installer.SubentityOT ent)
 - **Usage**
 - * Shows a dialog for selection of a new subentity
 - **Parameters**
 - * ent - SubentityOT. Encapsulate data type of subentity
- *addSubentity*
public void **addSubentity**(com.imagos.axilice.core.common.BaseVO subentity,
com.imagos.axilice.core.common.MultiSelectionSubentityOT subE)
 - **Usage**
 - * Adds a existent entity to a multiselection list
 - **Parameters**
 - * subentity - Subentity to be added
 - * subE - Object that encapsulates information about the subentity
- *addTrazabilityTree*
public void **addTrazabilityTree**(TabFolder tabFolder)
- *build_extension_fields_view*
public abstract void **build_extension_fields_view**()

– **Usage**

- * Build the view of the extension fields.

It must implement only those clients who require it

- *buildEntity*
public AxiliceEntityView buildEntity()
- *buildEntityDetail*
public AxiliceEntityView buildEntityDetail(
com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData)
- *buildEntitySelectionView*
public AxiliceEntitySelectionView buildEntitySelectionView()

– **Usage**

- * Build view of selection entities

– **Returns** - AxilicePartnerSelectionView. SelectionView created

- *buildSelectionSets*
public abstract void buildSelectionSets()

– **Usage**

- * Build 2 sets of SubentityOT: lSelectionSubentities and lMultiSelectionSubentities

These sets are used to connect the widgets with the handlers for the selection buttons and the multiselection panels

It must implement only those clients who require it

- *buildSubentityMultiSelection*
public void buildSubentityMultiSelection()
- *clearEntityData*
public void clearEntityData()
- *clearParentField*
public OperationResult clearParentField(java.util.ArrayList selRegistries,
java.lang.String parentField)

– **Usage**

- * Clears the value of the field indicated

– **Parameters**

- * selRegistries - ArrayList. Index list of records to process
- * parentField - Name of the field to clear

– **Returns** - OperationResult. Result of operation

- *delEntity*
public OperationResult delEntity()
- *delEntity*
public OperationResult delEntity(java.util.ArrayList selRegistries)
- *delEntity*
public OperationResult delEntity(com.imagos.axilice.core.common.BaseVO b)

– **Usage**

- * Delete a entity from database

– **Parameters**

- * b - BaseVO. Entity to delete

– **Returns** - OperationResult. Result of the operation

- *delSubentities*
public OperationResult delSubentities(java.util.ArrayList selRegistries)
 - **Usage**
 - * Delete a set of subentities from record list of controller, not from database
 - Set the parent field of subentity to null
 - **Parameters**
 - * selRegistries - List of index to delete
 - **Returns** - OperationResult. Result of the operation

- *editEntity*
public OperationResult editEntity()
- *fillEntityData*
public void fillEntityData()
- *fillEntityExtensionTable*
public void fillEntityExtensionTable(java.lang.Long id)
 - **Usage**
 - * Retrieve extension field values from the server for a given entity and displays them in the view
 - **Parameters**
 - * id - Long. Key of the entity whose values are to be displayed

- *fillEntityTable*
public void fillEntityTable()
 - **Usage**
 - * Load records from database and fill the entity view table and controller record list

- *fillEntityTable*
public void fillEntityTable(java.util.ArrayList lR)
 - **Usage**
 - * Fill the entity view table with the records given
 - **Parameters**
 - * lR - ArrayList. Records to be displayed

- *getClassEntity*
public Class getClassEntity()
 - **Returns** - Class of the entity for which the controller is built

- *getDialogName*
public String getDialogName()
- *getEditGladePath*
public String getEditGladePath()
- *getEditionGui*
public Gui getEditionGui()
 - **Usage**
 - * Returns the edition Gui instance.
 - **Returns** - Gui, entity edition Gui.

- *getEditWindow*
public SwtTopLevelWindow getEditWindow()

- *getEntityData*
public abstract BaseVO **getEntityData**()
- *getEntityFields*
public ArrayList **getEntityFields**()
- *getLogger*
public Logger **getLogger**()
- *getParentEntityLauncher*
public EntityLauncher **getParentEntityLauncher**()
 – **Returns** - EntityLauncher of parent entity if is a embedded entity or null otherwise
- *getRegistries*
public Collection **getRegistries**()
 – **Returns** - IRegistries
- *getRegistries*
public List **getRegistries**(java.util.ArrayList lIds)
 – **Usage**
 * Get a list of entities stored by the controller
 – **Parameters**
 * lIds - Index list
 – **Returns** - List of entities
- *getView*
public AxiliceEntityView **getView**()
 – **Returns** - AxiliceEntityView. Main view or Detail view if is a embedded entity.
- *getWidgets*
public abstract void **getWidgets**()
 – **Usage**
 * Get the widgets from the glade and stored in local variables.

 Must be overwritten
- *init_specific_components*
public abstract void **init_specific_components**()
 – **Usage**
 * Initializes the specific components of the client
- *printEntity*
public OperationResult **printEntity**()
- *printEntity*
public OperationResult **printEntity**(java.lang.String reportPath, java.util.HashMap parameters)
- *queryEntity*
public ArrayList **queryEntity**()
- *queryEntity*
public ArrayList **queryEntity**(java.lang.String fieldId, java.lang.String value)
- *queryEntityWithFormat*
public ArrayList **queryEntityWithFormat**(java.lang.String fieldId, java.lang.String value)
- *queryFreeEntities*
public ArrayList **queryFreeEntities**(com.imagos.axilice.core.common.MultiSelectionSubentityOT mSubE)

- **Usage**
 - * Search entities not associated with a parent entity (null parent field)
 - **Parameters**
 - * mSubE - SubentityOT for the parent entity that must be null
 - **Returns** - ArrayList. List of entities found
-
- *querySubentities*
 public ArrayList **querySubentities**(com.imagos.axilice.core.installer.SubentityOT
 entData)
-
- *refreshEntity*
 public OperationResult **refreshEntity**()
-
- *refreshEntity*
 public BaseVO **refreshEntity**(com.imagos.axilice.core.common.BaseVO entity)
-
- *refreshParentSelectedEntity*
 public OperationResult **refreshParentSelectedEntity**()
- **Usage**
 - * Reload from database the value of the parent entity of the selected entity
 - **Returns** - OperationResult. The result of the operation.
-
- *removeRegistries*
 public OperationResult **removeRegistries**(java.util.ArrayList selRegistries,
 java.lang.String parentField)
- **Usage**
 - * Deletes a list of the records stored by the controller. Also clean your parent fields and updates the database
 - **Parameters**
 - * selRegistries - Index list to remove
 - * parentField - Field name that links to the parent entity
 - **Returns** - OperationResult. Result of the operation
-
- *saveEntity*
 public OperationResult **saveEntity**()
-
- *saveEntity*
 public OperationResult **saveEntity**(java.util.ArrayList lIds)
-
- *saveEntity*
 public OperationResult **saveEntity**(com.imagos.axilice.core.common.BaseVO b)
- **Usage**
 - * Save or update a entity to database
 - **Parameters**
 - * b - BaseVO. Entity to save or update
 - **Returns** - OperationResult. Result of the operation
-
- *selectEntity*
 public void **selectEntity**(java.lang.Long idEntity)
-
- *selectSubentity*
 public void **selectSubentity**(java.lang.String parentFieldId,
 com.imagos.axilice.core.common.BaseVO subentityOT)
-
- *selectTreeItem*
 public void **selectTreeItem**(java.lang.Long id)
- **Usage**
 - * Select a item in the entity tree. No me gusta esto aqui
 - **Parameters**

- * `id` - Long. Index of item to select
- - *setDialogName*
public void setDialogName(java.lang.String dialogName)
 - *setEditGladePath*
public void setEditGladePath(java.lang.String editGladePath)
 - *setEditionMode*
public abstract void setEditionMode(boolean enabled)
 - *setRegistries*
public void setRegistries(java.util.Collection collection)
 - **Usage**
 - * Setter for IRegistries

- *setWidgetsValues*
public abstract void setWidgetsValues(com.imagos.axilice.core.common.BaseVO entity)
 - **Usage**
 - * Set the values of the widgets on the edit dialog.
 - Must be overwritten.
 - **Parameters**
 - * `entity` - BaseVO. Entity that contains the values to be displayed
- - *showErrorMessages*
public void showErrorMessages(com.imagos.axilice.core.util.OperationResult result)

- *updateEntityData*
public void updateEntityData()
- *updateRegistry*
public void updateRegistry(com.imagos.axilice.core.common.BaseVO ent)
 - **Usage**
 - * Add or update an entity (to the controller records)
 - **Parameters**
 - * `ent` - Entity to add or update
- - *updateViews*
public void updateViews(boolean updateFromParent)
 - **Usage**
 - * Force the update of the views associated
 - **Parameters**
 - * `updateFromParent` - Indicates whether the view must be updated by reading the values from the parent entity. takes effect only when the entity is detail (is embedded in another)

- *validateEntity*
public abstract OperationResult validateEntity()
 - **Usage**
 - * Validate the values of the widgets in the edit dialog.
 - Must be overwritten.
 - **Returns** - OperationResult. Result of the validation.

- *viewEntity*
public OperationResult viewEntity()

METHODS INHERITED FROM CLASS com.imagos.axilice.core.installer.EntityLauncher

(in 24.2.2, page 532)

- *addEntity*
 public abstract ArrayList **addEntity**()
 – **Usage**
 * Add entity button clicked handler. Show the entity content screen ready to start inserting data for a new instance of the entity (new registry).
 – **Returns** - ArrayList<ResultMessage>, list of errors if any has taken place, null in any other case.

- *buildEditEntityGUI*
 public void **buildEditEntityGUI**()
 – **Usage**
 * Builds the edit entity dialog window from the corresponding glade interface file. Stores the entity dialog instance for posterior accesses.

- *buildEntity*
 public AxiliceEntityView **buildEntity**()
 – **Usage**
 * Builds the entity GUI and show it on screen to the user.

- *buildEntity*
 public AxiliceDetailEntityView **buildEntity**(SwtWidget **parent**,
 com.imagos.axilice.core.common.MultiSelectionSubentityOT **subentityData**,
 java.lang.String **rootName**)
 – **Usage**
 * Builds the entity GUI and show it on screen to the user. The entity GUI will be embedded into the parent composite widget.
 – **Parameters**
 * **parent** - Composite, parent holder for embedding the entity view.
 * **subentityData** - SubentityOT, data of the subentity to build if the entity launcher will be used to handle a multi-selection detail entity view.
 * **rootName** - String, name of the root widget that will be embedded into the master entity edition panel.
 – **Returns** - AxiliceDetailEntityView, the entity detail view built.

- *cancelEntityEdition*
 public void **cancelEntityEdition**()

- *clearEntityData*
 public abstract void **clearEntityData**()
 – **Usage**
 * Clears the entity content panel by resetting the content widgets.

- *delEntity*
 public abstract OperationResult **delEntity**()
 – **Usage**
 * Delete entity button clicked handler. Deletes the selected instance of the entity (selected registry) from the database.
 – **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *delEntity*

```
public abstract OperationResult delEntity( java.util.ArrayList lSelRegistries )
```

- **Usage**

- * Delete entity button clicked handler. Deletes the selected instances of the entity (selected registries) from the database. Its a multiple delete method.

- **Parameters**

- * lSelRegistries - ArrayList<Long>, list of selected instances of the entity (selected registries). The list will contain only the internal id of the selected registries.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *deselectSubentity*

```
public void deselectSubentity( java.lang.String parentFieldId,  
com.imagos.axilice.core.common.BaseVO subentityOT )
```

- **Usage**

- * Updates an entity instance by removing the selected value for a subentity field.

- **Parameters**

- * parentFieldId - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.

- * subentityOT - BaseVO, subentity's deselected instance (registry) value.

- *editEntity*

```
public abstract OperationResult editEntity( )
```

- **Usage**

- * Edit entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), and allows the user to modify any content of the entity registry.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *fillEntityData*

```
public abstract void fillEntityData( )
```

- **Usage**

- * Fills the entity content panel with the data of the selected instance (registry) of the entity.

- *fireExceptionAndExit*

```
public void fireExceptionAndExit( java.lang.String msg )
```

- **Usage**

- * Fire a exception and stop the program

- **Parameters**

- * msg -

- *formatRegistries*

```
public ArrayList formatRegistries( )
```

- *formatRegistries*

```
public ArrayList formatRegistries( java.util.ArrayList lRegistries )
```

- *getDialog*

```
public SwtDialog getDialog( java.lang.String dialogName )
```

- *getEntity*

```
public EntityOT getEntity( )
```

- **Usage**

- * Gets the entity data.

- **Returns** - EntityOT, entity data.
- *getEntityData*
public abstract BaseVO **getEntityData**()
 - **Usage**
 - * Gets the content panel's data typed by the user formatted as an entity VO object.
- *getEntityEditViewName*
public String **getEntityEditViewName**(java.lang.String parentGuiName)
- *getEntityFields*
public abstract ArrayList **getEntityFields**()
 - **Usage**
 - * Gets the list of defined fields for an entity
 - **Returns** - ArrayList, list of entity fields.
- *getEntityViewName*
public String **getEntityViewName**(java.lang.String parentGuiName)
- *getLRegistries*
public ArrayList **getLRegistries**()
 - **Usage**
 - * Getter for lRegistries
- *getRegistry*
public BaseVO **getRegistry**(long id)
- *getRevision*
public static String **getRevision**()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
- *getSelectedEntity*
public BaseVO **getSelectedEntity**()
 - **Usage**
 - * Gets the selected instance (registry) of the entity.
 - **Returns** - BaseVO, selected instance (registry) of the entity.
- *getSelectedId*
public Long **getSelectedId**()
 - **Usage**
 - * Gets the internal id of the selected entity registry.
 - **Returns** - Long, internal id of the selected entity registry, null in case no entity registry has been selected yet.
- *getSwtWidget*
public SwtWidget **getSwtWidget**(java.lang.String widgetName)
- *getWidget*
public Widget **getWidget**(java.lang.String widgetName)
- *initEntityLauncher*
public void **initEntityLauncher**(Logger logger, Gui parentGui, SwtWidget parentPanel)
 - **Usage**
 - * Inits the entity launcher context and prepares the launcher to be started.
 - **Parameters**

- * **logger** - Logger, logger instance for printing logging messages.
- * **parentGuiName** - String, name of the JavaGlade instance that built the parent where the entity will be embedded.
- * **parentPanel** - Composite, parent panel where the entity contents will be embedded.

- *isDetail*

public boolean **isDetail**()

– **Usage**

- * Indicates whether is a main entity or is embedded in another

- *on_entitySelectionButton_clicked*

public void **on_entitySelectionButton_clicked**(Event e)

– **Usage**

- * Handler for the subentity selection buttons (magnifying glass buttons) clicked. Shows the selection dialog for selecting a valid instance of a subentity.

– **Parameters**

- * **e** - Event, magnifying glass button clicked event data.

- *printEntity*

public abstract **OperationResult printEntity**()

– **Usage**

- * Print entity button clicked handler. Builds the report with the data of the selected instance of the entity (selected registry), and allows the user to print the elaborated report.

– **Returns** - **OperationResult**, result of the delete operation and list of messages generated during the operation.

- *printWidgets*

public void **printWidgets**()

- *queryEntity*

public abstract **ArrayList queryEntity**()

– **Usage**

- * Search entity button clicked handler. Searches the system for the entity instances (registries) available.

– **Returns** - **ArrayList**, list of entity instances (registries) taken as the result of the query operation.

- *queryEntity*

public abstract **ArrayList queryEntity**(java.lang.String **fieldId**, java.lang.String **value**)

– **Usage**

- * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria.

– **Parameters**

- * **fieldId** - String, name of the entity field that will be queried.
- * **value** - String, value of the entity field that will be searched for.

– **Returns** - **ArrayList**, list of entity instances (registries) taken as the result of the query operation.

- *queryEntityWithFormat*

public abstract **ArrayList queryEntityWithFormat**(java.lang.String **fieldId**, java.lang.String **value**)

– **Usage**

- * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria and show them (formatted as a list) to the user in the entity's screen.

– **Parameters**

- * **fieldId** - String, name of the entity field that will be queried.
- * **value** - String, value of the entity field that will be searched for.

- **Returns** - ArrayList<RegistryOT>, list of entity instances (registries) taken as the result of the query operation.

• *querySubentities*

```
public abstract ArrayList querySubentities(
com.imagos.axilice.core.installer.SubentityOT entData )
```

– **Usage**

- * Queries the selected entity for all the detail subentities associated with it.

– **Parameters**

- * **entData** - SubentityOT, data of the detail subentity to query for.

- **Returns** - ArrayList, list of detail subentity registries assigned to the master detail registry.

• *saveEntity*

```
public abstract OperationResult saveEntity( )
```

– **Usage**

- * Save entity button clicked handler. Update the system with the data of the edited entity registry. It's called for saving new entity registries (add a new registry) or for updating the contents of existing entity registries.

- **Returns** - OperationResult, Result of the operation and list of messages generated through the operation.

• *saveEntityRegistry*

```
public void saveEntityRegistry( )
```

• *selectEntity*

```
public abstract void selectEntity( java.lang.Long idEntity )
```

– **Usage**

- * Marks an instance (registry) of the entity as the selected.

– **Parameters**

- * **idEntity** - Long, internal id of the selected instance (registry).

• *selectSubentity*

```
public void selectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

– **Usage**

- * Updates an entity instance by setting up the selected value for a subentity field.

– **Parameters**

- * **parentFieldId** - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
- * **subentityOT** - BaseVO, subentity's selected instance (registry) value.

• *setEntity*

```
public void setEntity( com.imagos.axilice.core.installer.EntityOT entity )
```

– **Usage**

- * Sets up the entity name.

– **Parameters**

- * **name** - String, entity name.
-

- *setRegistries*

```
public void setRegistries( java.util.ArrayList  lRegistries )
```

- Usage

- * Sets the list of registries that are showed in the entity view controlled by the Entity Launcher instance. It's used for detail entities where queries are made in the master entity launcher.

- Parameters

- * lRegistries - ArrayList<BaseVO>, list of shown registries of the entity.

- *showSelectionDialog*

```
public void showSelectionDialog( com.imagos.axilice.core.installer.SubentityOT
entData )
```

- Usage

- * Opens the selection dialog that enables the user to select an instance of a subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.

- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT
entData, long selEntity, boolean readonly )
```

- Usage

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.
 - * selEntity - long, id of the selected instance of the subentity.
 - * readonly - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT
entData, java.lang.String parentGuiName, long selEntity, boolean readonly )
```

- Usage

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.
 - * parentGuiName - String,
 - * selEntity - long, id of the selected instance of the subentity.
 - * readonly - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

- *showSubentitySelectionDialog*

```
public void showSubentitySelectionDialog(
com.imagos.axilice.core.installer.SubentityOT entData )
```

- Usage

- * Opens the selection dialog that enables the user to select an instance of a multiple selection detail subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.

- *undoEntityEdition*

```
public void undoEntityEdition( )
```

- *updateEntityData*
 public abstract void **updateEntityData**()
 – **Usage**
 * Updates the selected instance (registry) data with the content panel's data typed by the user.

- *viewEntity*
 public abstract **OperationResult** **viewEntity**()
 – **Usage**
 * View entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), in read-only mode.
 – **Returns** - **OperationResult**, result of the delete operation and list of messages generated during the operation.

27.1.2 CLASS AddressListener

DECLARATION

```
public class AddressListener
extends GenericHandler
```

CONSTRUCTORS

- *AddressListener*
 public **AddressListener**()

- *AddressListener*
 public **AddressListener**(com.imagos.axilice.client.partners.AddressClient
 addressClient)

METHODS

- *getAddressClient*
 public **AddressClient** **getAddressClient**()

- *getLogger*
 public **Logger** **getLogger**()

- *on_button_map_view_clicked*
 public void **on_button_map_view_clicked**(**Event** e)

- *on_country_combo_changed*
 public void **on_country_combo_changed**(**Event** e)

- *setAddressClient*
 public void **setAddressClient**(
 com.imagos.axilice.client.partners.AddressClient addressClient)

- *setLogger*
 public void **setLogger**(**Logger** logger)

METHODS INHERITED FROM CLASS GenericHandler

27.1.3 CLASS BankAccountClient

DECLARATION

```
public class BankAccountClient
extends com.imagos.axilice.client.gui.common.CommonClient
```

CONSTRUCTORS

- *BankAccountClient*
public **BankAccountClient**()

METHODS

- *build_extension_fields_view*
public void **build_extension_fields_view**()
- *buildSelectionSets*
public void **buildSelectionSets**()
- *clearEntityData*
public void **clearEntityData**()
- *fillEntityData*
public void **fillEntityData**()
- *getEntityData*
public BaseVO **getEntityData**()
- *getWidgets*
public void **getWidgets**()
- *init_specific_components*
public void **init_specific_components**()
- *setEditionMode*
public void **setEditionMode**(boolean enabled)
- *setWidgetsValues*
public void **setWidgetsValues**(com.imagos.axilice.core.common.BaseVO entity)
- *validateEntity*
public **OperationResult** **validateEntity**()

METHODS INHERITED FROM CLASS com.imagos.axilice.client.gui.common.CommonClient

(in 17.1.2, page 402)

- *addAxiliceEventListener*
 public void **addAxiliceEventListener**(
com.imagos.axilice.client.gui.AxiliceEventListener l)
- *addEntity*
 public ArrayList **addEntity**()
- *addExistingSubentity*
 public void **addExistingSubentity**(com.imagos.axilice.core.installer.SubentityOT
 ent)
 - Usage
 - * Shows a dialog for selection of a new subentity
 - Parameters
 - * ent - SubentityOT. Encapsulate data type of subentity
- *addSubentity*
 public void **addSubentity**(com.imagos.axilice.core.common.BaseVO subentity,
 com.imagos.axilice.core.common.MultiSelectionSubentityOT subE)
 - Usage
 - * Adds a existent entity to a multiselection list
 - Parameters
 - * subentity - Subentity to be added
 - * subE - Object that encapsulates information about the subentity
- *addTrazabilityTree*
 public void **addTrazabilityTree**(TabFolder tabFolder)
- *build_extension_fields_view*
 public abstract void **build_extension_fields_view**()
 - Usage
 - * Build the view of the extension fields.

It must implement only those clients who require it
- *buildEntity*
 public AxiliceEntityView **buildEntity**()
- *buildEntityDetail*
 public AxiliceEntityView **buildEntityDetail**(
com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData)
- *buildEntitySelectionView*
 public AxiliceEntitySelectionView **buildEntitySelectionView**()
 - Usage
 - * Build view of selection entities
 - Returns - AxilicePartnerSelectionView. SelectionView created
- *buildSelectionSets*
 public abstract void **buildSelectionSets**()
 - Usage

* Build 2 sets of SubentityOT: lSelectionSubentities and lMultiSelectionSubentities

These sets are used to connect the widgets with the handlers for the selection buttons and the multiselection panels

It must implement only those clients who require it

- buildSubentityMultiSelection
public void buildSubentityMultiSelection()
- clearEntityData
public void clearEntityData()
- clearParentField
public OperationResult clearParentField(java.util.ArrayList selRegistries,
java.lang.String parentField)
 - Usage
 - * Clears the value of the field indicated
 - Parameters
 - * selRegistries - ArrayList. Index list of records to process
 - * parentField - Name of the field to clear
 - Returns - OperationResult. Result of operation
- delEntity
public OperationResult delEntity()
- delEntity
public OperationResult delEntity(java.util.ArrayList selRegistries)
- delEntity
public OperationResult delEntity(com.imagos.axilice.core.common.BaseVO b)
 - Usage
 - * Delete a entity from database
 - Parameters
 - * b - BaseVO. Entity to delete
 - Returns - OperationResult. Result of the operation
- delSubentities
public OperationResult delSubentities(java.util.ArrayList selRegistries)
 - Usage
 - * Delete a set of subentities from record list of controller, not from database
 - Set the parent field of subentity to null
 - Parameters
 - * selRegistries - List of index to delete
 - Returns - OperationResult. Result of the operation
- editEntity
public OperationResult editEntity()
- fillEntityData
public void fillEntityData()
- fillEntityExtensionTable
public void fillEntityExtensionTable(java.lang.Long id)
 - Usage
 - * Retrieve extension field values from the server for a given entity and displays them in the view

- **Parameters**
 - * id - Long. Key of the entity whose values are to be displayed

- *fillEntityTable*
public void fillEntityTable()
 - **Usage**
 - * Load records from database and fill the entity view table and controller record list

- *fillEntityTable*
public void fillEntityTable(java.util.ArrayList lR)
 - **Usage**
 - * Fill the entity view table with the records given
 - **Parameters**
 - * lR - ArrayList. Records to be displayed

- *getClassEntity*
public Class getClassEntity()
 - **Returns** - Class of the entity for which the controller is built

- *getDialogName*
public String getDialogName()

- *getEditGladePath*
public String getEditGladePath()

- *getEditionGui*
public Gui getEditionGui()
 - **Usage**
 - * Returns the edition Gui instance.
 - **Returns** - Gui, entity edition Gui.

- *getEditWindow*
public SwtTopLevelWindow getEditWindow()

- *getEntityData*
public abstract BaseVO getEntityData()

- *getEntityFields*
public ArrayList getEntityFields()

- *getLogger*
public Logger getLogger()

- *getParentEntityLauncher*
public EntityLauncher getParentEntityLauncher()
 - **Returns** - EntityLauncher of parent entity if is a embedded entity or null otherwise

- *getRegistries*
public Collection getRegistries()
 - **Returns** - lRegistries

- *getRegistries*
public List getRegistries(java.util.ArrayList lIds)
 - **Usage**
 - * Get a list of entities stored by the controller
 - **Parameters**
 - * lIds - Index list
 - **Returns** - List of entities

- *getView*
 public AxiliceEntityView **getView**()
 – **Returns** - AxiliceEntityView. Main view or Detail view if is a embedded entity.

- *getWidgets*
 public abstract void **getWidgets**()
 – **Usage**
 * Get the widgets from the glade and stored in local variables.
 Must be overwritten

- *init_specific_components*
 public abstract void **init_specific_components**()
 – **Usage**
 * Initializes the specific components of the client

- *printEntity*
 public OperationResult **printEntity**()

- *printEntity*
 public OperationResult **printEntity**(java.lang.String **reportPath**, java.util.HashMap **parameters**)

- *queryEntity*
 public ArrayList **queryEntity**()

- *queryEntity*
 public ArrayList **queryEntity**(java.lang.String **fieldId**, java.lang.String **value**)

- *queryEntityWithFormat*
 public ArrayList **queryEntityWithFormat**(java.lang.String **fieldId**, java.lang.String **value**)

- *queryFreeEntities*
 public ArrayList **queryFreeEntities**(com.imagos.axilice.core.common.MultiSelectionSubentityOT **mSubE**)
 – **Usage**
 * Search entities not associated with a parent entity (null parent field)
 – **Parameters**
 * **mSubE** - SubentityOT for the parent entity that must be null
 – **Returns** - ArrayList. List of entities found

- *querySubentities*
 public ArrayList **querySubentities**(com.imagos.axilice.core.installer.SubentityOT **entData**)

- *refreshEntity*
 public OperationResult **refreshEntity**()

- *refreshEntity*
 public BaseVO **refreshEntity**(com.imagos.axilice.core.common.BaseVO **entity**)

- *refreshParentSelectedEntity*
 public OperationResult **refreshParentSelectedEntity**()
 – **Usage**
 * Reload from database the value of the parent entity of the selected entity
 – **Returns** - OperationResult. The result of the operation.

- *removeRegistries*
 public OperationResult **removeRegistries**(java.util.ArrayList **selRegistries**, java.lang.String **parentField**)

- **Usage**
 - * Deletes a list of the records stored by the controller. Also clean your parent fields and updates the database
 - **Parameters**
 - * **selRegistries** - Index list to remove
 - * **parentField** - Field name that links to the parent entity
 - **Returns** - **OperationResult**. Result of the operation
-
- *saveEntity*
public **OperationResult** **saveEntity**()
 - *saveEntity*
public **OperationResult** **saveEntity**(java.util.ArrayList **lIds**)
 - *saveEntity*
public **OperationResult** **saveEntity**(com.imagos.axilice.core.common.BaseVO **b**)
- **Usage**
 - * Save or update a entity to database
 - **Parameters**
 - * **b** - BaseVO. Entity to save or update
 - **Returns** - **OperationResult**. Result of the operation
-
- *selectEntity*
public void **selectEntity**(java.lang.Long **idEntity**)
 - *selectSubentity*
public void **selectSubentity**(java.lang.String **parentFieldId**,
com.imagos.axilice.core.common.BaseVO **subentityOT**)
 - *selectTreeItem*
public void **selectTreeItem**(java.lang.Long **id**)
- **Usage**
 - * Select a item in the entity tree. No me gusta esto aqui
 - **Parameters**
 - * **id** - Long. Index of item to select
-
- *setDialogName*
public void **setDialogName**(java.lang.String **dialogName**)
 - *setEditGladePath*
public void **setEditGladePath**(java.lang.String **editGladePath**)
 - *setEditionMode*
public abstract void **setEditionMode**(boolean **enabled**)
 - *setRegistries*
public void **setRegistries**(java.util.Collection **collection**)
- **Usage**
 - * Setter for lRegistries
-
- *setWidgetsValues*
public abstract void **setWidgetsValues**(com.imagos.axilice.core.common.BaseVO
entity)
- **Usage**
 - * Set the values of the widgets on the edit dialog.
 - **Parameters**
 - * **entity** - BaseVO. Entity that contains the values to be displayed

-
- *showErrorMessage*
 public void **showErrorMessage**(com.imagos.axilice.core.util.OperationResult result)

 - *updateEntityData*
 public void **updateEntityData**()

 - *updateRegistry*
 public void **updateRegistry**(com.imagos.axilice.core.common.BaseVO ent)
 - **Usage**
 - * Add or update an entity (to the controller records)
 - **Parameters**
 - * ent - Entity to add or update

 - *updateViews*
 public void **updateViews**(boolean updateFromParent)
 - **Usage**
 - * Force the update of the views associated
 - **Parameters**
 - * updateFromParent - Indicates whether the view must be updated by reading the values from the parent entity. takes effect only when the entity is detail (is embedded in another)

 - *validateEntity*
 public abstract OperationResult **validateEntity**()
 - **Usage**
 - * Validate the values of the widgets in the edit dialog.
 - **Returns** - OperationResult. Result of the validation.

 - *viewEntity*
 public OperationResult **viewEntity**()

METHODS INHERITED FROM CLASS com.imagos.axilice.core.installer.EntityLauncher

(in 24.2.2, page 532)

- *addEntity*
 public abstract ArrayList **addEntity**()
 - **Usage**
 - * Add entity button clicked handler. Show the entity content screen ready to start inserting data for a new instance of the entity (new registry).
 - **Returns** - ArrayList<ResultMessage>, list of errors if any has taken place, null in any other case.

- *buildEditEntityGUI*
 public void **buildEditEntityGUI**()
 - **Usage**
 - * Builds the edit entity dialog window from the corresponding glade interface file. Stores the entity dialog instance for posterior accesses.

- *buildEntity*
 public AxiliceEntityView **buildEntity**()
 - **Usage**

* Builds the entity GUI and show it on screen to the user.

- *buildEntity*

```
public AxiliceDetailView buildEntity( SwtWidget parent,
com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData,
java.lang.String rootName )
```

- **Usage**

- * Builds the entity GUI and show it on screen to the user. The entity GUI will be embedded into the parent composite widget.

- **Parameters**

- * **parent** - Composite, parent holder for embedding the entity view.
 - * **subentityData** - SubentityOT, data of the subentity to build if the entity launcher will be used to handle a multi-selection detail entity view.
 - * **rootName** - String, name of the root widget that will be embedded into the master entity edition panel.

- **Returns** - AxiliceDetailView, the entity detail view built.

- *cancelEntityEdition*

```
public void cancelEntityEdition( )
```

- *clearEntityData*

```
public abstract void clearEntityData( )
```

- **Usage**

- * Clears the entity content panel by resetting the content widgets.

- *delEntity*

```
public abstract OperationResult delEntity( )
```

- **Usage**

- * Delete entity button clicked handler. Deletes the selected instance of the entity (selected registry) from the database.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *delEntity*

```
public abstract OperationResult delEntity( java.util.ArrayList lSelRegistries )
```

- **Usage**

- * Delete entity button clicked handler. Deletes the selected instances of the entity (selected registries) from the database. Its a multiple delete method.

- **Parameters**

- * **lSelRegistries** - ArrayList<Long>, list of selected instances of the entity (selected registries). The list will contain only the internal id of the selected registries.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *deselectSubentity*

```
public void deselectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

- **Usage**

- * Updates an entity instance by removing the selected value for a subentity field.

- **Parameters**

- * **parentFieldId** - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * **subentityOT** - BaseVO, subentity's deselected instance (registry) value.

- *editEntity*
 public abstract **OperationResult editEntity()**
 - **Usage**
 - * Edit entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), and allows the user to modify any content of the entity registry.
 - **Returns** - **OperationResult**, result of the delete operation and list of messages generated during the operation.

- *fillEntityData*
 public abstract void **fillEntityData()**
 - **Usage**
 - * Fills the entity content panel with the data of the selected instance (registry) of the entity.

- *fireExceptionAndExit*
 public void **fireExceptionAndExit(java.lang.String msg)**
 - **Usage**
 - * Fire a exception and stop the program
 - **Parameters**
 - * msg -

- *formatRegistries*
 public **ArrayList formatRegistries()**
- *formatRegistries*
 public **ArrayList formatRegistries(java.util.ArrayList lRegistries)**

- *getDialog*
 public **SwtDialog getDialog(java.lang.String dialogName)**

- *getEntity*
 public **EntityOT getEntity()**
 - **Usage**
 - * Gets the entity data.
 - **Returns** - **EntityOT**, entity data.

- *getEntityData*
 public abstract **BaseVO getEntityData()**
 - **Usage**
 - * Gets the content panel's data typed by the user formatted as an entity VO object.

- *getEntityEditViewName*
 public **String getEntityEditViewName(java.lang.String parentGuiName)**

- *getEntityFields*
 public abstract **ArrayList getEntityFields()**
 - **Usage**
 - * Gets the list of defined fields for an entity
 - **Returns** - **ArrayList**, list of entity fields.

- *getEntityViewName*
 public **String getEntityViewName(java.lang.String parentGuiName)**

- *getLRegistries*
 public **ArrayList getLRegistries()**
 - **Usage**
 - * Getter for lRegistries

-
- *getRegistry*
`public BaseVO getRegistry(long id)`
 - *getRevision*
`public static String getRevision()`
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
-
- *getSelectedEntity*
`public BaseVO getSelectedEntity()`
 - **Usage**
 - * Gets the selected instance (registry) of the entity.
 - **Returns** - BaseVO, selected instance (registry) of the entity.
-
- *getSelectedId*
`public Long getSelectedId()`
 - **Usage**
 - * Gets the internal id of the selected entity registry.
 - **Returns** - Long, internal id of the selected entity registry, null in case no entity registry has been selected yet.
-
- *getSwtWidget*
`public SwtWidget getSwtWidget(java.lang.String widgetName)`
 - *getWidget*
`public Widget getWidget(java.lang.String widgetName)`
 - *initEntityLauncher*
`public void initEntityLauncher(Logger logger, Gui parentGui, SwtWidget parentPanel)`
 - **Usage**
 - * Inits the entity launcher context and prepares the launcher to be started.
 - **Parameters**
 - * **logger** - Logger, logger instance for printing logging messages.
 - * **parentGuiName** - String, name of the JavaGlade instance that built the parent where the entity will be embedded.
 - * **parentPanel** - Composite, parent panel where the entity contents will be embedded.
-
- *isDetail*
`public boolean isDetail()`
 - **Usage**
 - * Indicates whether is a main entity or is embedded in another
-
- *on_entitySelectionButton_clicked*
`public void on_entitySelectionButton_clicked(Event e)`
 - **Usage**
 - * Handler for the subentity selection buttons (magnifying glass buttons) clicked. Shows the selection dialog for selecting a valid instance of a subentity.
 - **Parameters**
 - * **e** - Event, magnifying glass button clicked event data.
-
- *printEntity*
`public abstract OperationResult printEntity()`
 - **Usage**

- * Print entity button clicked handler. Builds the report with the data of the selected instance of the entity (selected registry), and allows the user to print the elaborated report.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *printWidgets*
public void printWidgets()

- *queryEntity*
public abstract ArrayList queryEntity()
 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) available.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.

- *queryEntity*
public abstract ArrayList queryEntity(java.lang.String fieldId, java.lang.String value)
 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria.
 - **Parameters**
 - * **fieldId** - String, name of the entity field that will be queried.
 - * **value** - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.

- *queryEntityWithFormat*
public abstract ArrayList queryEntityWithFormat(java.lang.String fieldId, java.lang.String value)
 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria and show them (formatted as a list) to the user in the entity's screen.
 - **Parameters**
 - * **fieldId** - String, name of the entity field that will be queried.
 - * **value** - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList<RegistryOT>, list of entity instances (registries) taken as the result of the query operation.

- *querySubentities*
public abstract ArrayList querySubentities(com.imagos.axilice.core.installer.SubentityOT entData)
 - **Usage**
 - * Queries the selected entity for all the detail subentities associated with it.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to query for.
 - **Returns** - ArrayList, list of detail subentity registries assigned to the master detail registry.

- *saveEntity*
public abstract OperationResult saveEntity()
 - **Usage**

- * Save entity button clicked handler. Update the system with the data of the edited entity registry. It's called for saving new entity registries (add a new registry) or for updating the contents of existing entity registries.
 - **Returns** - OperationResult, Result of the operation and list of messages generated through the operation.

- *saveEntityRegistry*
public void saveEntityRegistry()
- *selectEntity*
public abstract void selectEntity(java.lang.Long idEntity)
 - **Usage**
 - * Marks an instance (registry) of the entity as the selected.
 - **Parameters**
 - * idEntity - Long, internal id of the selected instance (registry).

- *selectSubentity*
public void selectSubentity(java.lang.String parentFieldId, com.imagos.axilice.core.common.BaseVO subentityOT)
 - **Usage**
 - * Updates an entity instance by setting up the selected value for a subentity field.
 - **Parameters**
 - * parentFieldId - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * subentityOT - BaseVO, subentity's selected instance (registry) value.

- *setEntity*
public void setEntity(com.imagos.axilice.core.installer.EntityOT entity)
 - **Usage**
 - * Sets up the entity name.
 - **Parameters**
 - * name - String, entity name.

- *setRegistries*
public void setRegistries(java.util.ArrayList lRegistries)
 - **Usage**
 - * Sets the list of registries that are showed in the entity view controlled by the Entity Launcher instance. It's used for detail entities where queries are made in the master entity launcher.
 - **Parameters**
 - * lRegistries - ArrayList<BaseVO>, list of shown registries of the entity.

- *showSelectionDialog*
public void showSelectionDialog(com.imagos.axilice.core.installer.SubentityOT entData)
 - **Usage**
 - * Opens the selection dialog that enables the user to select an instance of a subentity.
 - **Parameters**
 - * entData - SubentityOT, data of the detail subentity to select.

- *showSubentityDialog*
public void showSubentityDialog(com.imagos.axilice.core.installer.SubentityOT entData, long selEntity, boolean readonly)
 - **Usage**

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

– **Parameters**

- * **entData** - SubentityOT, data of the detail subentity to select.
- * **selEntity** - long, id of the selected instance of the subentity.
- * **readonly** - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

• *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT
entData, java.lang.String parentGuiName, long selEntity, boolean readonly )
```

– **Usage**

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

– **Parameters**

- * **entData** - SubentityOT, data of the detail subentity to select.
- * **parentGuiName** - String,
- * **selEntity** - long, id of the selected instance of the subentity.
- * **readonly** - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

• *showSubentitySelectionDialog*

```
public void showSubentitySelectionDialog(
com.imagos.axilice.core.installer.SubentityOT entData )
```

– **Usage**

- * Opens the selection dialog that enables the user to select an instance of a multiple selection detail subentity.

– **Parameters**

- * **entData** - SubentityOT, data of the detail subentity to select.

• *undoEntityEdition*

```
public void undoEntityEdition( )
```

• *updateEntityData*

```
public abstract void updateEntityData( )
```

– **Usage**

- * Updates the selected instance (registry) data with the content panel's data typed by the user.

• *viewEntity*

```
public abstract OperationResult viewEntity( )
```

– **Usage**

- * View entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), in read-only mode.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

27.1.4 CLASS ContactClient

DECLARATION

```
public class ContactClient
extends com.imagos.axilice.client.gui.common.CommonClient
```

CONSTRUCTORS

- *ContactClient*
`public ContactClient()`

METHODS

- *build_extension_fields_view*
`public void build_extension_fields_view()`
- *buildSelectionSets*
`public void buildSelectionSets()`
- *clearEntityData*
`public void clearEntityData()`
- *fillEntityData*
`public void fillEntityData()`
- *getEntityData*
`public BaseVO getEntityData()`
- *getWidgets*
`public void getWidgets()`
- *init_specific_components*
`public void init_specific_components()`
- *setEditionMode*
`public void setEditionMode(boolean enabled)`
- *setWidgetsValues*
`public void setWidgetsValues(com.imagos.axilice.core.common.BaseVO entity)`
- *validateEntity*
`public OperationResult validateEntity()`

METHODS INHERITED FROM CLASS com.imagos.axilice.client.gui.common.CommonClient

(in 17.1.2, page 402)

- *addAxiliceEventListener*
`public void addAxiliceEventListener(com.imagos.axilice.client.gui.AxiliceEventListener l)`
- *addEntity*
`public ArrayList addEntity()`
- *addExistingSubentity*
`public void addExistingSubentity(com.imagos.axilice.core.installer.SubentityOT ent)`

– Usage

* Shows a dialog for selection of a new subentity

– Parameters

- * **ent** - SubentityOT. Encapsulate data type of subentity
- - *addSubentity*
 public void **addSubentity**(com.imagos.axilice.core.common.BaseVO **subentity**,
 com.imagos.axilice.core.common.MultiSelectionSubentityOT **subE**)
 - **Usage**
 - * Adds a existent entity to a multiselection list
 - **Parameters**
 - * **subentity** - Subentity to be added
 - * **subE** - Object that encapsulates information about the subentity
- - *addTrazabilityTree*
 public void **addTrazabilityTree**(TabFolder **tabFolder**)
- *build_extension_fields_view*
 public abstract void **build_extension_fields_view**()
 - **Usage**
 - * Build the view of the extension fields.

It must implement only those clients who require it
- - *buildEntity*
 public AxiliceEntityView **buildEntity**()
- *buildEntityDetail*
 public AxiliceEntityView **buildEntityDetail**(
 com.imagos.axilice.core.common.MultiSelectionSubentityOT **subentityData**)
- - *buildEntitySelectionView*
 public AxiliceEntitySelectionView **buildEntitySelectionView**()
 - **Usage**
 - * Build view of selection entities
 - **Returns** - AxilicePartnerSelectionView. SelectionView created
- - *buildSelectionSets*
 public abstract void **buildSelectionSets**()
 - **Usage**
 - * Build 2 sets of SubentityOT: lSelectionSubentities and lMultiSelectionSubentities

These sets are used to connect the widgets with the handlers for the selection buttons and the multiseleccion panels

It must implement only those clients who require it
- - *buildSubentityMultiSelection*
 public void **buildSubentityMultiSelection**()
- - *clearEntityData*
 public void **clearEntityData**()
- - *clearParentField*
 public OperationResult **clearParentField**(java.util.ArrayList **selRegistries**,
 java.lang.String **parentField**)
 - **Usage**
 - * Clears the value of the field indicated
 - **Parameters**
 - * **selRegistries** - ArrayList. Index list of records to process
 - * **parentField** - Name of the field to clear

- **Returns** - OperationResult. Result of operation

- *delEntity*
public OperationResult delEntity()
- *delEntity*
public OperationResult delEntity(java.util.ArrayList selRegistries)
- *delEntity*
public OperationResult delEntity(com.imagos.axilice.core.common.BaseVO b)
 - **Usage**
 - * Delete a entity from database
 - **Parameters**
 - * b - BaseVO. Entity to delete
 - **Returns** - OperationResult. Result of the operation

- *delSubentities*
public OperationResult delSubentities(java.util.ArrayList selRegistries)
 - **Usage**
 - * Delete a set of subentities from record list of controller, not from database
 - Set the parent field of subentity to null
 - **Parameters**
 - * selRegistries - List of index to delete
 - **Returns** - OperationResult. Result of the operation

- *editEntity*
public OperationResult editEntity()
- *fillEntityData*
public void fillEntityData()
- *fillEntityExtensionTable*
public void fillEntityExtensionTable(java.lang.Long id)
 - **Usage**
 - * Retrieve extension field values from the server for a given entity and displays them in the view
 - **Parameters**
 - * id - Long. Key of the entity whose values are to be displayed

- *fillEntityTable*
public void fillEntityTable()
 - **Usage**
 - * Load records from database and fill the entity view table and controller record list

- *fillEntityTable*
public void fillEntityTable(java.util.ArrayList lR)
 - **Usage**
 - * Fill the entity view table with the records given
 - **Parameters**
 - * lR - ArrayList. Records to be displayed

- *getClassEntity*
public Class getClassEntity()
 - **Returns** - Class of the entity for which the controller is built

- getDialogName
public String getDialogName()
- getEditGladePath
public String getEditGladePath()
- getEditionGui
public Gui getEditionGui()
 - **Usage**
 - * Returns the edition Gui instance.
 - **Returns** - Gui, entity edition Gui.
- getEditWindow
public SwtTopLevelWindow getEditWindow()
- getEntityData
public abstract BaseVO getEntityData()
- getEntityFields
public ArrayList getEntityFields()
- getLogger
public Logger getLogger()
- getParentEntityLauncher
public EntityLauncher getParentEntityLauncher()
 - **Returns** - EntityLauncher of parent entity if is a embedded entity or null otherwise
- getRegistries
public Collection getRegistries()
 - **Returns** - lRegistries
- getRegistries
public List getRegistries(java.util.ArrayList lIds)
 - **Usage**
 - * Get a list of entities stored by the controller
 - **Parameters**
 - * lIds - Index list
 - **Returns** - List of entities
- getView
public AxiliceEntityView getView()
 - **Returns** - AxiliceEntityView. Main view or Detail view if is a embedded entity.
- getWidgets
public abstract void getWidgets()
 - **Usage**
 - * Get the widgets from the glade and stored in local variables.

Must be overwritten
- init_specific_components
public abstract void init_specific_components()
 - **Usage**
 - * Initializes the specific components of the client
- printEntity
public OperationResult printEntity()

- *printEntity*
public OperationResult printEntity(java.lang.String reportPath, java.util.HashMap parameters)
- *queryEntity*
public ArrayList queryEntity()
- *queryEntity*
public ArrayList queryEntity(java.lang.String fieldId, java.lang.String value)
- *queryEntityWithFormat*
public ArrayList queryEntityWithFormat(java.lang.String fieldId, java.lang.String value)
- *queryFreeEntities*
public ArrayList queryFreeEntities(com.imagos.axilice.core.common.MultiSelectionSubentityOT mSubE)
 - Usage
 - * Search entities not associated with a parent entity (null parent field)
 - Parameters
 - * mSubE - SubentityOT for the parent entity that must be null
 - Returns - ArrayList. List of entities found
- *querySubentities*
public ArrayList querySubentities(com.imagos.axilice.core.installer.SubentityOT entData)
- *refreshEntity*
public OperationResult refreshEntity()
- *refreshEntity*
public BaseVO refreshEntity(com.imagos.axilice.core.common.BaseVO entity)
- *refreshParentSelectedEntity*
public OperationResult refreshParentSelectedEntity()
 - Usage
 - * Reload from database the value of the parent entity of the selected entity
 - Returns - OperationResult. The result of the operation.
- *removeRegistries*
public OperationResult removeRegistries(java.util.ArrayList selRegistries, java.lang.String parentField)
 - Usage
 - * Deletes a list of the records stored by the controller. Also clean your parent fields and updates the database
 - Parameters
 - * selRegistries - Index list to remove
 - * parentField - Field name that links to the parent entity
 - Returns - OperationResult. Result of the operation
- *saveEntity*
public OperationResult saveEntity()
- *saveEntity*
public OperationResult saveEntity(java.util.ArrayList lIds)
- *saveEntity*
public OperationResult saveEntity(com.imagos.axilice.core.common.BaseVO b)
 - Usage
 - * Save or update a entity to database
 - Parameters

- * **b** - BaseVO. Entity to save or update
 - **Returns** - OperationResult. Result of the operation

- *selectEntity*
public void selectEntity(java.lang.Long idEntity)
- *selectSubentity*
public void selectSubentity(java.lang.String parentFieldId, com.imagos.axilice.core.common.BaseVO subentityOT)
- *selectTreeItem*
public void selectTreeItem(java.lang.Long id)
 - **Usage**
 - * Select a item in the entity tree. No me gusta esto aqui
 - **Parameters**
 - * **id** - Long. Index of item to select

- *setDialogName*
public void setDialogName(java.lang.String dialogName)
- *setEditGladePath*
public void setEditGladePath(java.lang.String editGladePath)
- *setEditionMode*
public abstract void setEditionMode(boolean enabled)
- *setRegistries*
public void setRegistries(java.util.Collection collection)
 - **Usage**
 - * Setter for lRegistries

- *setWidgetsValues*
public abstract void setWidgetsValues(com.imagos.axilice.core.common.BaseVO entity)
 - **Usage**
 - * Set the values of the widgets on the edit dialog.
 - Must be overwritten.
 - **Parameters**
 - * **entity** - BaseVO. Entity that contains the values to be displayed

- *showErrorMessages*
public void showErrorMessages(com.imagos.axilice.core.util.OperationResult result)

- *updateEntityData*
public void updateEntityData()
- *updateRegistry*
public void updateRegistry(com.imagos.axilice.core.common.BaseVO ent)
 - **Usage**
 - * Add or update an entity (to the controller records)
 - **Parameters**
 - * **ent** - Entity to add or update

- *updateViews*
public void updateViews(boolean updateFromParent)
 - **Usage**
 - * Force the update of the views associated

- **Parameters**
 - * **updateFromParent** - Indicates whether the view must be updated by reading the values from the parent entity. takes effect only when the entity is detail (is embedded in another)
-
- *validateEntity*
`public abstract OperationResult validateEntity()`
 - **Usage**
 - * Validate the values of the widgets in the edit dialog.
 - Must be overwritten.
 - **Returns** - OperationResult. Result of the validation.
-
- *viewEntity*
`public OperationResult viewEntity()`

METHODS INHERITED FROM CLASS com.imagos.axilice.core.installer.EntityLauncher

(in 24.2.2, page 532)

- *addEntity*
`public abstract ArrayList addEntity()`
 - **Usage**
 - * Add entity button clicked handler. Show the entity content screen ready to start inserting data for a new instance of the entity (new registry).
 - **Returns** - ArrayList<ResultMessage>, list of errors if any has taken place, null in any other case.
-
- *buildEditEntityGUI*
`public void buildEditEntityGUI()`
 - **Usage**
 - * Builds the edit entity dialog window from the corresponding glade interface file. Stores the entity dialog instance for posterior accesses.
-
- *buildEntity*
`public AxiliceEntityView buildEntity()`
 - **Usage**
 - * Builds the entity GUI and show it on screen to the user.
-
- *buildEntity*
`public AxiliceDetailEntityView buildEntity(SwtWidget parent,
com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData,
java.lang.String rootName)`
 - **Usage**
 - * Builds the entity GUI and show it on screen to the user. The entity GUI will be embedded into the parent composite widget.
 - **Parameters**
 - * **parent** - Composite, parent holder for embedding the entity view.
 - * **subentityData** - SubentityOT, data of the subentity to build if the entity launcher will be used to handle a multi-selection detail entity view.
 - * **rootName** - String, name of the root widget that will be embedded into the master entity edition panel.
 - **Returns** - AxiliceDetailEntityView, the entity detail view built.
-

- *cancelEntityEdition*

```
public void cancelEntityEdition( )
```
- *clearEntityData*

```
public abstract void clearEntityData( )
```

 - **Usage**
 - * Clears the entity content panel by resetting the content widgets.

- *delEntity*

```
public abstract OperationResult delEntity( )
```

 - **Usage**
 - * Delete entity button clicked handler. Deletes the selected instance of the entity (selected registry) from the database.
 - **Returns** - **OperationResult**, result of the delete operation and list of messages generated during the operation.

- *delEntity*

```
public abstract OperationResult delEntity( java.util.ArrayList lSelRegistries )
```

 - **Usage**
 - * Delete entity button clicked handler. Deletes the selected instances of the entity (selected registries) from the database. Its a multiple delete method.
 - **Parameters**
 - * **lSelRegistries** - **ArrayList<Long>**, list of selected instances of the entity (selected registries). The list will contain only the internal id of the selected registries.
 - **Returns** - **OperationResult**, result of the delete operation and list of messages generated during the operation.

- *deselectSubentity*

```
public void deselectSubentity( java.lang.String parentFieldId,  
com.imagos.axilice.core.common.BaseVO subentityOT )
```

 - **Usage**
 - * Updates an entity instance by removing the selected value for a subentity field.
 - **Parameters**
 - * **parentFieldId** - **String**, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * **subentityOT** - **BaseVO**, subentity's deselected instance (registry) value.

- *editEntity*

```
public abstract OperationResult editEntity( )
```

 - **Usage**
 - * Edit entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), and allows the user to modify any content of the entity registry.
 - **Returns** - **OperationResult**, result of the delete operation and list of messages generated during the operation.

- *fillEntityData*

```
public abstract void fillEntityData( )
```

 - **Usage**
 - * Fills the entity content panel with the data of the selected instance (registry) of the entity.

- *fireExceptionAndExit*

```
public void fireExceptionAndExit( java.lang.String msg )
```

 - **Usage**

- * Fire a exception and stop the program
 - **Parameters**
 - * msg -
- formatRegistries
public ArrayList formatRegistries()
- formatRegistries
public ArrayList formatRegistries(java.util.ArrayList lRegistries)
- getDialog
public SwtDialog getDialog(java.lang.String dialogName)
- getEntity
public EntityOT getEntity()
 - **Usage**
 - * Gets the entity data.
 - **Returns** - EntityOT, entity data.
- getEntityData
public abstract BaseVO getEntityData()
 - **Usage**
 - * Gets the content panel's data typed by the user formatted as an entity VO object.
- getEntityEditViewName
public String getEntityEditViewName(java.lang.String parentGuiName)
- getEntityFields
public abstract ArrayList getEntityFields()
 - **Usage**
 - * Gets the list of defined fields for an entity
 - **Returns** - ArrayList, list of entity fields.
- getEntityViewName
public String getEntityViewName(java.lang.String parentGuiName)
- getLRegistries
public ArrayList getLRegistries()
 - **Usage**
 - * Getter for lRegistries
- getRegistry
public BaseVO getRegistry(long id)
- getRevision
public static String getRevision()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
- getSelectedEntity
public BaseVO getSelectedEntity()
 - **Usage**
 - * Gets the selected instance (registry) of the entity.
 - **Returns** - BaseVO, selected instance (registry) of the entity.
- getSelectedId
public Long getSelectedId()
 - **Usage**

- * Gets the internal id of the selected entity registry.
 - **Returns** - Long, internal id of the selected entity registry, null in case no entity registry has been selected yet.

- *getSwtWidget*
public SwtWidget getSwtWidget(java.lang.String widgetName)
- *getWidget*
public Widget getWidget(java.lang.String widgetName)
- *initEntityLauncher*
public void initEntityLauncher(Logger logger, Gui parentGui, SwtWidget parentPanel)
 - **Usage**
 - * Inits the entity launcher context and prepares the launcher to be started.
 - **Parameters**
 - * **logger** - Logger, logger instance for printing logging messages.
 - * **parentGuiName** - String, name of the JavaGlade instance that built the parent where the entity will be embedded.
 - * **parentPanel** - Composite, parent panel where the entity contents will be embedded.

- *isDetail*
public boolean isDetail()
 - **Usage**
 - * Indicates whether is a main entity or is embedded in another

- *on_entitySelectionButton_clicked*
public void on_entitySelectionButton_clicked(Event e)
 - **Usage**
 - * Handler for the subentity selection buttons (magnifying glass buttons) clicked. Shows the selection dialog for selecting a valid instance of a subentity.
 - **Parameters**
 - * **e** - Event, magnifying glass button clicked event data.

- *printEntity*
public abstract OperationResult printEntity()
 - **Usage**
 - * Print entity button clicked handler. Builds the report with the data of the selected instance of the entity (selected registry), and allows the user to print the elaborated report.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *printWidgets*
public void printWidgets()
- *queryEntity*
public abstract ArrayList queryEntity()
 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) available.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.

- *queryEntity*
public abstract ArrayList queryEntity(java.lang.String fieldId, java.lang.String value)

- **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria.
 - **Parameters**
 - * **fieldId** - String, name of the entity field that will be queried.
 - * **value** - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.
-
- *queryEntityWithFormat*
 public abstract ArrayList **queryEntityWithFormat**(java.lang.String **fieldId**,
 java.lang.String **value**)
 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria and show them (formatted as a list) to the user in the entity's screen.
 - **Parameters**
 - * **fieldId** - String, name of the entity field that will be queried.
 - * **value** - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList<RegistryOT>, list of entity instances (registries) taken as the result of the query operation.
-
- *querySubentities*
 public abstract ArrayList **querySubentities**(
 com.imagos.axilice.core.installer.SubentityOT **entData**)
 - **Usage**
 - * Queries the selected entity for all the detail subentities associated with it.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to query for.
 - **Returns** - ArrayList, list of detail subentity registries assigned to the master detail registry.
-
- *saveEntity*
 public abstract OperationResult **saveEntity**()
 - **Usage**
 - * Save entity button clicked handler. Update the system with the data of the edited entity registry. It's called for saving new entity registries (add a new registry) or for updating the contents of existing entity registries.
 - **Returns** - OperationResult, Result of the operation and list of messages generated through the operation.
-
- *saveEntityRegistry*
 public void **saveEntityRegistry**()
-
- *selectEntity*
 public abstract void **selectEntity**(java.lang.Long **idEntity**)
 - **Usage**
 - * Marks an instance (registry) of the entity as the selected.
 - **Parameters**
 - * **idEntity** - Long, internal id of the selected instance (registry).
-
- *selectSubentity*
 public void **selectSubentity**(java.lang.String **parentFieldId**,
 com.imagos.axilice.core.common.BaseV0 **subentityOT**)
 - **Usage**

- * Updates an entity instance by setting up the selected value for a subentity field.
- **Parameters**
 - * **parentFieldId** - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * **subentityOT** - BaseVO, subentity's selected instance (registry) value.

- *setEntity*

```
public void setEntity( com.imagos.axilice.core.installer.EntityOT entity )
```

 - **Usage**
 - * Sets up the entity name.
 - **Parameters**
 - * **name** - String, entity name.

- *setRegistries*

```
public void setRegistries( java.util.ArrayList lRegistries )
```

 - **Usage**
 - * Sets the list of registries that are showed in the entity view controlled by the Entity Launcher instance. It's used for detail entities where queries are made in the master entity launcher.
 - **Parameters**
 - * **lRegistries** - ArrayList<BaseVO>, list of shown registries of the entity.

- *showSelectionDialog*

```
public void showSelectionDialog( com.imagos.axilice.core.installer.SubentityOT entData )
```

 - **Usage**
 - * Opens the selection dialog that enables the user to select an instance of a subentity.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to select.

- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT entData, long selEntity, boolean readonly )
```

 - **Usage**
 - * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to select.
 - * **selEntity** - long, id of the selected instance of the subentity.
 - * **readonly** - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT entData, java.lang.String parentGuiName, long selEntity, boolean readonly )
```

 - **Usage**
 - * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to select.
 - * **parentGuiName** - String,
 - * **selEntity** - long, id of the selected instance of the subentity.
 - * **readonly** - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

-
- *showSubentitySelectionDialog*
 public void **showSubentitySelectionDialog**(
 com.imagos.axilice.core.installer.SubentityOT **entData**)
 – **Usage**
 * Opens the selection dialog that enables the user to select an instance of a multiple selection detail subentity.
 – **Parameters**
 * **entData** - SubentityOT, data of the detail subentity to select.
 - *undoEntityEdition*
 public void **undoEntityEdition**()
 - *updateEntityData*
 public abstract void **updateEntityData**()
 – **Usage**
 * Updates the selected instance (registry) data with the content panel's data typed by the user.
 - *viewEntity*
 public abstract **OperationResult** **viewEntity**()
 – **Usage**
 * View entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), in read-only mode.
 – **Returns** - **OperationResult**, result of the delete operation and list of messages generated during the operation.

27.1.5 CLASS DepartmentClient

DECLARATION

```
public class DepartmentClient
extends com.imagos.axilice.client.gui.common.CommonClient
```

CONSTRUCTORS

- *DepartmentClient*
 public **DepartmentClient**()

METHODS

- *build_extension_fields_view*
 public void **build_extension_fields_view**()
- *buildSelectionSets*
 public void **buildSelectionSets**()

- *buildSubentityMultiSelection*
public void **buildSubentityMultiSelection**()
- *clearEntityData*
public void **clearEntityData**()
- *connectSubentityButtons*
public void **connectSubentityButtons**()
- *fillEntityData*
public void **fillEntityData**()
- *getEntityData*
public BaseVO **getEntityData**()
- *getSelectedId*
public Long **getSelectedId**()
- *getWidgets*
public void **getWidgets**()
- *init_specific_components*
public void **init_specific_components**()
- *queryEntityWithFormat*
public ArrayList **queryEntityWithFormat**(java.lang.String fieldId,
java.lang.String value)
- *querySubentities*
public ArrayList **querySubentities**(
com.imagos.axilice.core.installer.SubentityOT entData)
- *selectEntity*
public void **selectEntity**(java.lang.Long idEntity)
- *setEditionMode*
public void **setEditionMode**(boolean enabled)
- *setWidgetsValues*
public void **setWidgetsValues**(com.imagos.axilice.core.common.BaseVO entity
)
- *updateEntityData*
public void **updateEntityData**()
- *validateEntity*
public OperationResult **validateEntity**()

METHODS INHERITED FROM CLASS com.imagos.axilice.client.gui.common.CommonClient

(in 17.1.2, page 402)

- *addAxiliceEventListener*
public void **addAxiliceEventListener**(
com.imagos.axilice.client.gui.AxiliceEventListener l)
- *addEntity*
public ArrayList **addEntity**()

- *addExistingSubentity*

```
public void addExistingSubentity( com.imagos.axilice.core.installer.SubentityOT
ent )
```

- **Usage**

- * Shows a dialog for selection of a new subentity

- **Parameters**

- * **ent** - SubentityOT. Encapsulate data type of subentity

- *addSubentity*

```
public void addSubentity( com.imagos.axilice.core.common.BaseVO subentity,
com.imagos.axilice.core.common.MultiSelectionSubentityOT subE )
```

- **Usage**

- * Adds a existent entity to a multiselection list

- **Parameters**

- * **subentity** - Subentity to be added

- * **subE** - Object that encapsulates information about the subentity

- *addTrazabilityTree*

```
public void addTrazabilityTree( TabFolder tabFolder )
```

- *build_extension_fields_view*

```
public abstract void build_extension_fields_view( )
```

- **Usage**

- * Build the view of the extension fields.

It must implement only those clients who require it

- *buildEntity*

```
public AxiliceEntityView buildEntity( )
```

- *buildEntityDetail*

```
public AxiliceEntityView buildEntityDetail(
com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData )
```

- *buildEntitySelectionView*

```
public AxiliceEntitySelectionView buildEntitySelectionView( )
```

- **Usage**

- * Build view of selection entities

- **Returns** - AxilicePartnerSelectionView. SelectionView created

- *buildSelectionSets*

```
public abstract void buildSelectionSets( )
```

- **Usage**

- * Build 2 sets of SubentityOT: lSelectionSubentities and lMultiSelectionSubentities

These sets are used to connect the widgets with the handlers for the selection buttons and the multiselection panels

It must implement only those clients who require it

- *buildSubentityMultiSelection*

```
public void buildSubentityMultiSelection( )
```

- *clearEntityData*

```
public void clearEntityData( )
```

- *clearParentField*
 public OperationResult **clearParentField**(java.util.ArrayList selRegistries,
 java.lang.String parentField)
 – **Usage**
 * Clears the value of the field indicated
 – **Parameters**
 * selRegistries - ArrayList. Index list of records to process
 * parentField - Name of the field to clear
 – **Returns** - OperationResult. Result of operation

- *delEntity*
 public OperationResult **delEntity**()
- *delEntity*
 public OperationResult **delEntity**(java.util.ArrayList selRegistries)
- *delEntity*
 public OperationResult **delEntity**(com.imagos.axilice.core.common.BaseVO b)
 – **Usage**
 * Delete a entity from database
 – **Parameters**
 * b - BaseVO. Entity to delete
 – **Returns** - OperationResult. Result of the operation

- *delSubentities*
 public OperationResult **delSubentities**(java.util.ArrayList selRegistries)
 – **Usage**
 * Delete a set of subentities from record list of controller, not from database
 Set the parent field of subentity to null
 – **Parameters**
 * selRegistries - List of index to delete
 – **Returns** - OperationResult. Result of the operation

- *editEntity*
 public OperationResult **editEntity**()
- *fillEntityData*
 public void **fillEntityData**()
- *fillEntityExtensionTable*
 public void **fillEntityExtensionTable**(java.lang.Long id)
 – **Usage**
 * Retrieve extension field values from the server for a given entity and displays them in the view
 – **Parameters**
 * id - Long. Key of the entity whose values are to be displayed

- *fillEntityTable*
 public void **fillEntityTable**()
 – **Usage**
 * Load records from database and fill the entity view table and controller record list

- *fillEntityTable*
 public void **fillEntityTable**(java.util.ArrayList IR)

- **Usage**
 - * Fill the entity view table with the records given
- **Parameters**
 - * lR - ArrayList. Records to be displayed

- *getClassEntity*
public Class getClassEntity()
 - **Returns** - Class of the entity for which the controller is built

- *getDialogName*
public String getDialogName()
- *getEditGladePath*
public String getEditGladePath()
- *getEditionGui*
public Gui getEditionGui()
 - **Usage**
 - * Returns the edition Gui instance.
 - **Returns** - Gui, entity edition Gui.

- *getEditWindow*
public SwtTopLevelWindow getEditWindow()
- *getEntityData*
public abstract BaseVO getEntityData()
- *getEntityFields*
public ArrayList getEntityFields()
- *getLogger*
public Logger getLogger()
- *getParentEntityLauncher*
public EntityLauncher getParentEntityLauncher()
 - **Returns** - EntityLauncher of parent entity if is a embedded entity or null otherwise

- *getRegistries*
public Collection getRegistries()
 - **Returns** - lRegistries

- *getRegistries*
public List getRegistries(java.util.ArrayList lIds)
 - **Usage**
 - * Get a list of entities stored by the controller
 - **Parameters**
 - * lIds - Index list
 - **Returns** - List of entities

- *getView*
public AxiliceEntityView getView()
 - **Returns** - AxiliceEntityView. Main view or Detail view if is a embedded entity.

- *getWidgets*
public abstract void getWidgets()
 - **Usage**
 - * Get the widgets from the glade and stored in local variables.

Must be overwritten

-
- *init_specific_components*
 public abstract void **init_specific_components**()
 – **Usage**
 * Initializes the specific components of the client

 - *printEntity*
 public OperationResult **printEntity**()

 - *printEntity*
 public OperationResult **printEntity**(java.lang.String reportPath, java.util.HashMap parameters)

 - *queryEntity*
 public ArrayList **queryEntity**()

 - *queryEntity*
 public ArrayList **queryEntity**(java.lang.String fieldId, java.lang.String value)

 - *queryEntityWithFormat*
 public ArrayList **queryEntityWithFormat**(java.lang.String fieldId, java.lang.String value)

 - *queryFreeEntities*
 public ArrayList **queryFreeEntities**(com.imagos.axilice.core.common.MultiSelectionSubentityOT mSubE)
 – **Usage**
 * Search entities not associated with a parent entity (null parent field)
 – **Parameters**
 * mSubE - SubentityOT for the parent entity that must be null
 – **Returns** - ArrayList. List of entities found

 - *querySubentities*
 public ArrayList **querySubentities**(com.imagos.axilice.core.installer.SubentityOT entData)

 - *refreshEntity*
 public OperationResult **refreshEntity**()

 - *refreshEntity*
 public BaseVO **refreshEntity**(com.imagos.axilice.core.common.BaseVO entity)

 - *refreshParentSelectedEntity*
 public OperationResult **refreshParentSelectedEntity**()
 – **Usage**
 * Reload from database the value of the parent entity of the selected entity
 – **Returns** - OperationResult. The result of the operation.

 - *removeRegistries*
 public OperationResult **removeRegistries**(java.util.ArrayList selRegistries, java.lang.String parentField)
 – **Usage**
 * Deletes a list of the records stored by the controller. Also clean your parent fields and updates the database
 – **Parameters**
 * selRegistries - Index list to remove
 * parentField - Field name that links to the parent entity
 – **Returns** - OperationResult. Result of the operation

- *saveEntity*
public OperationResult saveEntity()
- *saveEntity*
public OperationResult saveEntity(java.util.ArrayList lIds)
- *saveEntity*
public OperationResult saveEntity(com.imagos.axilice.core.common.BaseVO b)
 - **Usage**
 - * Save or update a entity to database
 - **Parameters**
 - * b - BaseVO. Entity to save or update
 - **Returns** - OperationResult. Result of the operation

- *selectEntity*
public void selectEntity(java.lang.Long idEntity)
- *selectSubentity*
public void selectSubentity(java.lang.String parentFieldId, com.imagos.axilice.core.common.BaseVO subentityOT)
- *selectTreeItem*
public void selectTreeItem(java.lang.Long id)
 - **Usage**
 - * Select a item in the entity tree. No me gusta esto aqui
 - **Parameters**
 - * id - Long. Index of item to select

- *setDialogName*
public void setDialogName(java.lang.String dialogName)
- *setEditGladePath*
public void setEditGladePath(java.lang.String editGladePath)
- *setEditionMode*
public abstract void setEditionMode(boolean enabled)
- *setRegistries*
public void setRegistries(java.util.Collection collection)
 - **Usage**
 - * Setter for lRegistries

- *setWidgetsValues*
public abstract void setWidgetsValues(com.imagos.axilice.core.common.BaseVO entity)
 - **Usage**
 - * Set the values of the widgets on the edit dialog.
 - Must be overwritten.
 - **Parameters**
 - * entity - BaseVO. Entity that contains the values to be displayed

- *showErrorMessage*
public void showErrorMessage(com.imagos.axilice.core.util.OperationResult result)

- *updateEntityData*
public void updateEntityData()
- *updateRegistry*
public void updateRegistry(com.imagos.axilice.core.common.BaseVO ent)

- **Usage**
 - * Add or update an entity (to the controller records)
 - **Parameters**
 - * **ent** - Entity to add or update
-
- *updateViews*
public void updateViews(boolean updateFromParent)
 - **Usage**
 - * Force the update of the views associated
 - **Parameters**
 - * **updateFromParent** - Indicates whether the view must be updated by reading the values from the parent entity. takes effect only when the entity is detail (is embedded in another)
-
- *validateEntity*
public abstract OperationResult validateEntity()
 - **Usage**
 - * Validate the values of the widgets in the edit dialog.
 - **Returns** - OperationResult. Result of the validation.
-
- *viewEntity*
public OperationResult viewEntity()

METHODS INHERITED FROM CLASS com.imagos.axilice.core.installer.EntityLauncher

(in 24.2.2, page 532)

- *addEntity*
public abstract ArrayList addEntity()
 - **Usage**
 - * Add entity button clicked handler. Show the entity content screen ready to start inserting data for a new instance of the entity (new registry).
 - **Returns** - ArrayList<ResultMessage>, list of errors if any has taken place, null in any other case.
-
- *buildEditEntityGUI*
public void buildEditEntityGUI()
 - **Usage**
 - * Builds the edit entity dialog window from the corresponding glade interface file. Stores the entity dialog instance for posterior accesses.
-
- *buildEntity*
public AxiliceEntityView buildEntity()
 - **Usage**
 - * Builds the entity GUI and show it on screen to the user.
-
- *buildEntity*
public AxiliceDetailEntityView buildEntity(SwtWidget parent, com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData, java.lang.String rootName)
 - **Usage**

- * Builds the entity GUI and show it on screen to the user. The entity GUI will be embedded into the parent composite widget.

– **Parameters**

- * **parent** - Composite, parent holder for embedding the entity view.
- * **subentityData** - SubentityOT, data of the subentity to build if the entity launcher will be used to handle a multi-selection detail entity view.
- * **rootName** - String, name of the root widget that will be embedded into the master entity edition panel.

– **Returns** - AxiliceDetailView, the entity detail view built.

• *cancelEntityEdition*

public void cancelEntityEdition()

• *clearEntityData*

public abstract void clearEntityData()

– **Usage**

- * Clears the entity content panel by resetting the content widgets.

• *delEntity*

public abstract OperationResult delEntity()

– **Usage**

- * Delete entity button clicked handler. Deletes the selected instance of the entity (selected registry) from the database.

– **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

• *delEntity*

public abstract OperationResult delEntity(java.util.ArrayList lSelRegistries)

– **Usage**

- * Delete entity button clicked handler. Deletes the selected instances of the entity (selected registries) from the database. Its a multiple delete method.

– **Parameters**

- * **lSelRegistries** - ArrayList<Long>, list of selected instances of the entity (selected registries). The list will contain only the internal id of the selected registries.

– **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

• *deselectSubentity*

**public void deselectSubentity(java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT)**

– **Usage**

- * Updates an entity instance by removing the selected value for a subentity field.

– **Parameters**

- * **parentFieldId** - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
- * **subentityOT** - BaseVO, subentity's deselected instance (registry) value.

• *editEntity*

public abstract OperationResult editEntity()

– **Usage**

- * Edit entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), and allows the user to modify any content of the entity registry.

– **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

-
- *fillEntityData*
 public abstract void **fillEntityData**()
 – **Usage**
 * Fills the entity content panel with the data of the selected instance (registry) of the entity.

 - *fireExceptionAndExit*
 public void **fireExceptionAndExit**(java.lang.String msg)
 – **Usage**
 * Fire a exception and stop the program
 – **Parameters**
 * msg -

 - *formatRegistries*
 public ArrayList **formatRegistries**()
 - *formatRegistries*
 public ArrayList **formatRegistries**(java.util.ArrayList lRegistries)
 - *getDialog*
 public SwtDialog **getDialog**(java.lang.String dialogName)
 - *getEntity*
 public EntityOT **getEntity**()
 – **Usage**
 * Gets the entity data.
 – **Returns** - EntityOT, entity data.

 - *getEntityData*
 public abstract BaseVO **getEntityData**()
 – **Usage**
 * Gets the content panel's data typed by the user formatted as an entity VO object.

 - *getEntityEditViewName*
 public String **getEntityEditViewName**(java.lang.String parentGuiName)
 - *getEntityFields*
 public abstract ArrayList **getEntityFields**()
 – **Usage**
 * Gets the list of defined fields for an entity
 – **Returns** - ArrayList, list of entity fields.

 - *getEntityViewName*
 public String **getEntityViewName**(java.lang.String parentGuiName)
 - *getLRegistries*
 public ArrayList **getLRegistries**()
 – **Usage**
 * Getter for lRegistries

 - *getRegistry*
 public BaseVO **getRegistry**(long id)
 - *getRevision*
 public static String **getRevision**()
 – **Usage**
 * Returns tag Id assigned to CVS source file.

- *getSelectedEntity*
public BaseVO getSelectedEntity()
 - **Usage**
 - * Gets the selected instance (registry) of the entity.
 - **Returns** - BaseVO, selected instance (registry) of the entity.

- *getSelectedId*
public Long getSelectedId()
 - **Usage**
 - * Gets the internal id of the selected entity registry.
 - **Returns** - Long, internal id of the selected entity registry, null in case no entity registry has been selected yet.

- *getSwtWidget*
public SwtWidget getSwtWidget(java.lang.String widgetName)

- *getWidget*
public Widget getWidget(java.lang.String widgetName)

- *initEntityLauncher*
public void initEntityLauncher(Logger logger, Gui parentGui, SwtWidget parentPanel)
 - **Usage**
 - * Initiates the entity launcher context and prepares the launcher to be started.
 - **Parameters**
 - * **logger** - Logger, logger instance for printing logging messages.
 - * **parentGuiName** - String, name of the JavaGlade instance that built the parent where the entity will be embedded.
 - * **parentPanel** - Composite, parent panel where the entity contents will be embedded.

- *isDetail*
public boolean isDetail()
 - **Usage**
 - * Indicates whether is a main entity or is embedded in another

- *on_entitySelectionButton_clicked*
public void on_entitySelectionButton_clicked(Event e)
 - **Usage**
 - * Handler for the subentity selection buttons (magnifying glass buttons) clicked. Shows the selection dialog for selecting a valid instance of a subentity.
 - **Parameters**
 - * **e** - Event, magnifying glass button clicked event data.

- *printEntity*
public abstract OperationResult printEntity()
 - **Usage**
 - * Print entity button clicked handler. Builds the report with the data of the selected instance of the entity (selected registry), and allows the user to print the elaborated report.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *printWidgets*
public void printWidgets()

- *queryEntity*
public abstract ArrayList queryEntity()

- **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) available.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.
-
- *queryEntity*
 public abstract ArrayList **queryEntity**(java.lang.String fieldId, java.lang.String value)
 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria.
 - **Parameters**
 - * fieldId - String, name of the entity field that will be queried.
 - * value - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.
-
- *queryEntityWithFormat*
 public abstract ArrayList **queryEntityWithFormat**(java.lang.String fieldId, java.lang.String value)
 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria and show them (formatted as a list) to the user in the entity's screen.
 - **Parameters**
 - * fieldId - String, name of the entity field that will be queried.
 - * value - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList<RegistryOT>, list of entity instances (registries) taken as the result of the query operation.
-
- *querySubentities*
 public abstract ArrayList **querySubentities**(com.imagos.axilice.core.installer.SubentityOT entData)
 - **Usage**
 - * Queries the selected entity for all the detail subentities associated with it.
 - **Parameters**
 - * entData - SubentityOT, data of the detail subentity to query for.
 - **Returns** - ArrayList, list of detail subentity registries assigned to the master detail registry.
-
- *saveEntity*
 public abstract OperationResult **saveEntity**()
 - **Usage**
 - * Save entity button clicked handler. Update the system with the data of the edited entity registry. It's called for saving new entity registries (add a new registry) or for updating the contents of existing entity registries.
 - **Returns** - OperationResult, Result of the operation and list of messages generated through the operation.
-
- *saveEntityRegistry*
 public void **saveEntityRegistry**()
 - *selectEntity*
 public abstract void **selectEntity**(java.lang.Long idEntity)

- **Usage**
 - * Marks an instance (registry) of the entity as the selected.
 - **Parameters**
 - * `idEntity` - Long, internal id of the selected instance (registry).
-
- *selectSubentity*

```
public void selectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

 - **Usage**
 - * Updates an entity instance by setting up the selected value for a subentity field.
 - **Parameters**
 - * `parentFieldId` - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * `subentityOT` - BaseVO, subentity's selected instance (registry) value.
-
- *setEntity*

```
public void setEntity( com.imagos.axilice.core.installer.EntityOT entity )
```

 - **Usage**
 - * Sets up the entity name.
 - **Parameters**
 - * `name` - String, entity name.
-
- *setRegistries*

```
public void setRegistries( java.util.ArrayList lRegistries )
```

 - **Usage**
 - * Sets the list of registries that are showed in the entity view controlled by the Entity Launcher instance. It's used for detail entities where queries are made in the master entity launcher.
 - **Parameters**
 - * `lRegistries` - ArrayList<BaseVO>, list of shown registries of the entity.
-
- *showSelectionDialog*

```
public void showSelectionDialog( com.imagos.axilice.core.installer.SubentityOT
entData )
```

 - **Usage**
 - * Opens the selection dialog that enables the user to select an instance of a subentity.
 - **Parameters**
 - * `entData` - SubentityOT, data of the detail subentity to select.
-
- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT
entData, long selEntity, boolean readonly )
```

 - **Usage**
 - * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.
 - **Parameters**
 - * `entData` - SubentityOT, data of the detail subentity to select.
 - * `selEntity` - long, id of the selected instance of the subentity.
 - * `readonly` - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.
-
- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT
entData, java.lang.String parentGuiName, long selEntity, boolean readonly )
```

- **Usage**
 - * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to select.
 - * **parentGuiName** - String,
 - * **selEntity** - long, id of the selected instance of the subentity.
 - * **readonly** - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.
-
- *showSubentitySelectionDialog*

```
public void showSubentitySelectionDialog(
com.imagos.axilice.core.installer.SubentityOT entData )
```

 - **Usage**
 - * Opens the selection dialog that enables the user to select an instance of a multiple selection detail subentity.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to select.
-
- *undoEntityEdition*

```
public void undoEntityEdition( )
```
 - *updateEntityData*

```
public abstract void updateEntityData( )
```

 - **Usage**
 - * Updates the selected instance (registry) data with the content panel's data typed by the user.
-
- *viewEntity*

```
public abstract OperationResult viewEntity( )
```

 - **Usage**
 - * View entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), in read-only mode.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

27.1.6 CLASS DiscountClient

DECLARATION

```
public class DiscountClient
extends com.imagos.axilice.client.gui.common.CommonClient
```

CONSTRUCTORS

- *DiscountClient*

```
public DiscountClient( )
```

METHODS

-
- *build_extension_fields_view*
public void build_extension_fields_view()
 - *buildSelectionSets*
public void buildSelectionSets()
 - *fillEntityData*
public void fillEntityData()
 - *getEntityData*
public BaseVO getEntityData()
 - *getWidgets*
public void getWidgets()
 - *init_specific_components*
public void init_specific_components()
 - *setEditionMode*
public void setEditionMode(boolean enabled)
 - *setWidgetsValues*
public void setWidgetsValues(com.imagos.axilice.core.common.BaseVO entity)
 - *validateEntity*
public OperationResult validateEntity()

METHODS INHERITED FROM CLASS com.imagos.axilice.client.gui.common.CommonClient

(in 17.1.2, page 402)

- *addAxiliceEventListener*
public void addAxiliceEventListener(com.imagos.axilice.client.gui.AxiliceEventListener l)
- *addEntity*
public ArrayList addEntity()
- *addExistingSubentity*
public void addExistingSubentity(com.imagos.axilice.core.installer.SubentityOT ent)
 - **Usage**
 - * Shows a dialog for selection of a new subentity
 - **Parameters**
 - * ent - SubentityOT. Encapsulate data type of subentity
- *addSubentity*
public void addSubentity(com.imagos.axilice.core.common.BaseVO subentity, com.imagos.axilice.core.common.MultiSelectionSubentityOT subE)
 - **Usage**
 - * Adds a existent entity to a multiselection list
 - **Parameters**
 - * subentity - Subentity to be added

* **subE** - Object that encapsulates information about the subentity

- *addTrazabilityTree*

public void addTrazabilityTree(TabFolder tabFolder)

- *build_extension_fields_view*

public abstract void build_extension_fields_view()

– **Usage**

* Build the view of the extension fields.

It must implement only those clients who require it

- *buildEntity*

public AxiliceEntityView buildEntity()

- *buildEntityDetail*

**public AxiliceEntityView buildEntityDetail(
com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData)**

- *buildEntitySelectionView*

public AxiliceEntitySelectionView buildEntitySelectionView()

– **Usage**

* Build view of selection entities

– **Returns** - AxilicePartnerSelectionView. SelectionView created

- *buildSelectionSets*

public abstract void buildSelectionSets()

– **Usage**

* Build 2 sets of SubentityOT: lSelectionSubentities and lMultiSelectionSubentities

These sets are used to connect the widgets with the handlers for the selection buttons and the multiselection panels

It must implement only those clients who require it

- *buildSubentityMultiSelection*

public void buildSubentityMultiSelection()

- *clearEntityData*

public void clearEntityData()

- *clearParentField*

**public OperationResult clearParentField(java.util.ArrayList selRegistries,
java.lang.String parentField)**

– **Usage**

* Clears the value of the field indicated

– **Parameters**

* **selRegistries** - ArrayList. Index list of records to process

* **parentField** - Name of the field to clear

– **Returns** - OperationResult. Result of operation

- *delEntity*

public OperationResult delEntity()

- *delEntity*

public OperationResult delEntity(java.util.ArrayList selRegistries)

- *delEntity*

public OperationResult delEntity(com.imagos.axilice.core.common.BaseVO b)

– **Usage**

- * Delete a entity from database
 - **Parameters**
 - * **b** - BaseVO. Entity to delete
 - **Returns** - OperationResult. Result of the operation
- delSubentities
 public OperationResult **delSubentities**(java.util.ArrayList **selRegistries**)
 - **Usage**
 - * Delete a set of subentities from record list of controller, not from database
 - Set the parent field of subentity to null
 - **Parameters**
 - * **selRegistries** - List of index to delete
 - **Returns** - OperationResult. Result of the operation
- editEntity
 public OperationResult **editEntity**()
- fillEntityData
 public void **fillEntityData**()
- fillEntityExtensionTable
 public void **fillEntityExtensionTable**(java.lang.Long **id**)
 - **Usage**
 - * Retrieve extension field values from the server for a given entity and displays them in the view
 - **Parameters**
 - * **id** - Long. Key of the entity whose values are to be displayed
- fillEntityTable
 public void **fillEntityTable**()
 - **Usage**
 - * Load records from database and fill the entity view table and controller record list
- fillEntityTable
 public void **fillEntityTable**(java.util.ArrayList **lR**)
 - **Usage**
 - * Fill the entity view table with the records given
 - **Parameters**
 - * **lR** - ArrayList. Records to be displayed
- getClassEntity
 public Class **getClassEntity**()
 - **Returns** - Class of the entity for which the controller is built
- getDialogName
 public String **getDialogName**()
- getEditGladePath
 public String **getEditGladePath**()
- getEditionGui
 public Gui **getEditionGui**()
 - **Usage**
 - * Returns the edition Gui instance.

- **Returns** - Gui, entity edition Gui.
- getEditWindow
public SwtTopLevelWindow **getEditWindow**()
- getEntityData
public abstract BaseVO **getEntityData**()
- getEntityFields
public ArrayList **getEntityFields**()
- getLogger
public Logger **getLogger**()
- getParentEntityLauncher
public EntityLauncher **getParentEntityLauncher**()
 - **Returns** - EntityLauncher of parent entity if is a embedded entity or null otherwise
- getRegistries
public Collection **getRegistries**()
 - **Returns** - IRegistries
- getRegistries
public List **getRegistries**(java.util.ArrayList lIds)
 - **Usage**
 - * Get a list of entities stored by the controller
 - **Parameters**
 - * lIds - Index list
 - **Returns** - List of entities
- getView
public AxiliceEntityView **getView**()
 - **Returns** - AxiliceEntityView. Main view or Detail view if is a embedded entity.
- getWidgets
public abstract void **getWidgets**()
 - **Usage**
 - * Get the widgets from the glade and stored in local variables.
 - Must be overwritten
- init_specific_components
public abstract void **init_specific_components**()
 - **Usage**
 - * Initializes the specific components of the client
- printEntity
public OperationResult **printEntity**()
- printEntity
public OperationResult **printEntity**(java.lang.String reportPath, java.util.HashMap parameters)
- queryEntity
public ArrayList **queryEntity**()
- queryEntity
public ArrayList **queryEntity**(java.lang.String fieldId, java.lang.String value)

- *queryEntityWithFormat*
public ArrayList queryEntityWithFormat(java.lang.String fieldId, java.lang.String value)
- *queryFreeEntities*
public ArrayList queryFreeEntities(com.imagos.axilice.core.common.MultiSelectionSubentityOT mSubE)
 - **Usage**
 - * Search entities not associated with a parent entity (null parent field)
 - **Parameters**
 - * mSubE - SubentityOT for the parent entity that must be null
 - **Returns** - ArrayList. List of entities found
- *querySubentities*
public ArrayList querySubentities(com.imagos.axilice.core.installer.SubentityOT entData)
- *refreshEntity*
public OperationResult refreshEntity()
- *refreshEntity*
public BaseVO refreshEntity(com.imagos.axilice.core.common.BaseVO entity)
- *refreshParentSelectedEntity*
public OperationResult refreshParentSelectedEntity()
 - **Usage**
 - * Reload from database the value of the parent entity of the selected entity
 - **Returns** - OperationResult. The result of the operation.
- *removeRegistries*
public OperationResult removeRegistries(java.util.ArrayList selRegistries, java.lang.String parentField)
 - **Usage**
 - * Deletes a list of the records stored by the controller. Also clean your parent fields and updates the database
 - **Parameters**
 - * selRegistries - Index list to remove
 - * parentField - Field name that links to the parent entity
 - **Returns** - OperationResult. Result of the operation
- *saveEntity*
public OperationResult saveEntity()
- *saveEntity*
public OperationResult saveEntity(java.util.ArrayList lIds)
- *saveEntity*
public OperationResult saveEntity(com.imagos.axilice.core.common.BaseVO b)
 - **Usage**
 - * Save or update a entity to database
 - **Parameters**
 - * b - BaseVO. Entity to save or update
 - **Returns** - OperationResult. Result of the operation
- *selectEntity*
public void selectEntity(java.lang.Long idEntity)

- *selectSubentity*
 public void **selectSubentity**(java.lang.String parentFieldId,
 com.imagos.axilice.core.common.BaseVO subentityOT)
- *selectTreeItem*
 public void **selectTreeItem**(java.lang.Long id)
 - **Usage**
 - * Select a item in the entity tree. No me gusta esto aqui
 - **Parameters**
 - * id - Long. Index of item to select
- *setDialogName*
 public void **setDialogName**(java.lang.String dialogName)
- *setEditGladePath*
 public void **setEditGladePath**(java.lang.String editGladePath)
- *setEditionMode*
 public abstract void **setEditionMode**(boolean enabled)
- *setRegistries*
 public void **setRegistries**(java.util.Collection collection)
 - **Usage**
 - * Setter for IRegistries
- *setWidgetsValues*
 public abstract void **setWidgetsValues**(com.imagos.axilice.core.common.BaseVO
 entity)
 - **Usage**
 - * Set the values of the widgets on the edit dialog.
 - Must be overwritten.
 - **Parameters**
 - * entity - BaseVO. Entity that contains the values to be displayed
- *showErrorMessages*
 public void **showErrorMessages**(com.imagos.axilice.core.util.OperationResult result)
- *updateEntityData*
 public void **updateEntityData**()
- *updateRegistry*
 public void **updateRegistry**(com.imagos.axilice.core.common.BaseVO ent)
 - **Usage**
 - * Add or update an entity (to the controller records)
 - **Parameters**
 - * ent - Entity to add or update
- *updateViews*
 public void **updateViews**(boolean updateFromParent)
 - **Usage**
 - * Force the update of the views associated
 - **Parameters**
 - * updateFromParent - Indicates whether the view must be updated by reading the values from the parent entity. takes effect only when the entity is detail (is embedded in another)

- *validateEntity*
 public abstract **OperationResult validateEntity()**
 - **Usage**
 - * Validate the values of the widgets in the edit dialog.
 - **Returns** - **OperationResult**. Result of the validation.
-
- *viewEntity*
 public **OperationResult viewEntity()**

METHODS INHERITED FROM CLASS `com.imagos.axilice.core.installer.EntityLauncher`

(in 24.2.2, page 532)

- *addEntity*
 public abstract **ArrayList addEntity()**
 - **Usage**
 - * Add entity button clicked handler. Show the entity content screen ready to start inserting data for a new instance of the entity (new registry).
 - **Returns** - **ArrayList<ResultMessage>**, list of errors if any has taken place, null in any other case.
-
- *buildEditEntityGUI*
 public void **buildEditEntityGUI()**
 - **Usage**
 - * Builds the edit entity dialog window from the corresponding glade interface file. Stores the entity dialog instance for posterior accesses.
-
- *buildEntity*
 public **AxiliceEntityView buildEntity()**
 - **Usage**
 - * Builds the entity GUI and show it on screen to the user.
-
- *buildEntity*
 public **AxiliceDetailEntityView buildEntity(SwtWidget parent, com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData, java.lang.String rootName)**
 - **Usage**
 - * Builds the entity GUI and show it on screen to the user. The entity GUI will be embedded into the parent composite widget.
 - **Parameters**
 - * **parent** - Composite, parent holder for embedding the entity view.
 - * **subentityData** - SubentityOT, data of the subentity to build if the entity launcher will be used to handle a multi-selection detail entity view.
 - * **rootName** - String, name of the root widget that will be embedded into the master entity edition panel.
 - **Returns** - **AxiliceDetailEntityView**, the entity detail view built.
-
- *cancelEntityEdition*
 public void **cancelEntityEdition()**
-
- *clearEntityData*
 public abstract void **clearEntityData()**

- **Usage**
 - * Clears the entity content panel by resetting the content widgets.

- *delEntity*

```
public abstract OperationResult delEntity( )
```

 - **Usage**
 - * Delete entity button clicked handler. Deletes the selected instance of the entity (selected registry) from the database.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *delEntity*

```
public abstract OperationResult delEntity( java.util.ArrayList lSelRegistries )
```

 - **Usage**
 - * Delete entity button clicked handler. Deletes the selected instances of the entity (selected registries) from the database. Its a multiple delete method.
 - **Parameters**
 - * lSelRegistries - ArrayList<Long>, list of selected instances of the entity (selected registries). The list will contain only the internal id of the selected registries.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *deselectSubentity*

```
public void deselectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

 - **Usage**
 - * Updates an entity instance by removing the selected value for a subentity field.
 - **Parameters**
 - * parentFieldId - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * subentityOT - BaseVO, subentity's deselected instance (registry) value.

- *editEntity*

```
public abstract OperationResult editEntity( )
```

 - **Usage**
 - * Edit entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), and allows the user to modify any content of the entity registry.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *fillEntityData*

```
public abstract void fillEntityData( )
```

 - **Usage**
 - * Fills the entity content panel with the data of the selected instance (registry) of the entity.

- *fireExceptionAndExit*

```
public void fireExceptionAndExit( java.lang.String msg )
```

 - **Usage**
 - * Fire a exception and stop the program
 - **Parameters**
 - * msg -

- *formatRegistries*
public ArrayList formatRegistries()
- *formatRegistries*
public ArrayList formatRegistries(java.util.ArrayList lRegistries)
- *getDialog*
public SwtDialog getDialog(java.lang.String dialogName)
- *getEntity*
public EntityOT getEntity()
 - **Usage**
 - * Gets the entity data.
 - **Returns** - EntityOT, entity data.

- *getEntityData*
public abstract BaseVO getEntityData()
 - **Usage**
 - * Gets the content panel's data typed by the user formatted as an entity VO object.

- *getEntityEditViewName*
public String getEntityEditViewName(java.lang.String parentGuiName)
- *getEntityFields*
public abstract ArrayList getEntityFields()
 - **Usage**
 - * Gets the list of defined fields for an entity
 - **Returns** - ArrayList, list of entity fields.

- *getEntityViewName*
public String getEntityViewName(java.lang.String parentGuiName)
- *getLRegistries*
public ArrayList getLRegistries()
 - **Usage**
 - * Getter for lRegistries

- *getRegistry*
public BaseVO getRegistry(long id)
- *getRevision*
public static String getRevision()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.

- *getSelectedEntity*
public BaseVO getSelectedEntity()
 - **Usage**
 - * Gets the selected instance (registry) of the entity.
 - **Returns** - BaseVO, selected instance (registry) of the entity.

- *getSelectedId*
public Long getSelectedId()
 - **Usage**
 - * Gets the internal id of the selected entity registry.
 - **Returns** - Long, internal id of the selected entity registry, null in case no entity registry has been selected yet.

- *getSwtWidget*
public SwtWidget getSwtWidget(java.lang.String widgetName)
- *getWidget*
public Widget getWidget(java.lang.String widgetName)
- *initEntityLauncher*
public void initEntityLauncher(Logger logger, Gui parentGui, SwtWidget parentPanel)
 - **Usage**
 - * Initiates the entity launcher context and prepares the launcher to be started.
 - **Parameters**
 - * **logger** - Logger, logger instance for printing logging messages.
 - * **parentGui** - String, name of the JavaGlade instance that built the parent where the entity will be embedded.
 - * **parentPanel** - Composite, parent panel where the entity contents will be embedded.

- *isDetail*
public boolean isDetail()
 - **Usage**
 - * Indicates whether is a main entity or is embedded in another

- *on_entitySelectionButton_clicked*
public void on_entitySelectionButton_clicked(Event e)
 - **Usage**
 - * Handler for the subentity selection buttons (magnifying glass buttons) clicked. Shows the selection dialog for selecting a valid instance of a subentity.
 - **Parameters**
 - * **e** - Event, magnifying glass button clicked event data.

- *printEntity*
public abstract OperationResult printEntity()
 - **Usage**
 - * Print entity button clicked handler. Builds the report with the data of the selected instance of the entity (selected registry), and allows the user to print the elaborated report.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *printWidgets*
public void printWidgets()
- *queryEntity*
public abstract ArrayList queryEntity()
 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) available.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.

- *queryEntity*
public abstract ArrayList queryEntity(java.lang.String fieldId, java.lang.String value)
 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria.
 - **Parameters**

- * **fieldId** - String, name of the entity field that will be queried.
 - * **value** - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.

- *queryEntityWithFormat*

```
public abstract ArrayList queryEntityWithFormat( java.lang.String fieldId,
java.lang.String value )
```

 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria and show them (formatted as a list) to the user in the entity's screen.
 - **Parameters**
 - * **fieldId** - String, name of the entity field that will be queried.
 - * **value** - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList<RegistryOT>, list of entity instances (registries) taken as the result of the query operation.

- *querySubentities*

```
public abstract ArrayList querySubentities(
com.imagos.axilice.core.installer.SubentityOT entData )
```

 - **Usage**
 - * Queries the selected entity for all the detail subentities associated with it.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to query for.
 - **Returns** - ArrayList, list of detail subentity registries assigned to the master detail registry.

- *saveEntity*

```
public abstract OperationResult saveEntity( )
```

 - **Usage**
 - * Save entity button clicked handler. Update the system with the data of the edited entity registry. It's called for saving new entity registries (add a new registry) or for updating the contents of existing entity registries.
 - **Returns** - OperationResult, Result of the operation and list of messages generated through the operation.

- *saveEntityRegistry*

```
public void saveEntityRegistry( )
```
- *selectEntity*

```
public abstract void selectEntity( java.lang.Long idEntity )
```

 - **Usage**
 - * Marks an instance (registry) of the entity as the selected.
 - **Parameters**
 - * **idEntity** - Long, internal id of the selected instance (registry).

- *selectSubentity*

```
public void selectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

 - **Usage**
 - * Updates an entity instance by setting up the selected value for a subentity field.
 - **Parameters**
 - * **parentFieldId** - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.

-
- * subentityOT - BaseVO, subentity's selected instance (registry) value.
-
- *setEntity*
 public void **setEntity**(com.imagos.axilice.core.installer.EntityOT entity)
 – **Usage**
 * Sets up the entity name.
 – **Parameters**
 * name - String, entity name.
-
- *setRegistries*
 public void **setRegistries**(java.util.ArrayList lRegistries)
 – **Usage**
 * Sets the list of registries that are showed in the entity view controlled by the Entity Launcher instance. It's used for detail entities where queries are made in the master entity launcher.
 – **Parameters**
 * lRegistries - ArrayList<BaseVO>, list of shown registries of the entity.
-
- *showSelectionDialog*
 public void **showSelectionDialog**(com.imagos.axilice.core.installer.SubentityOT entData)
 – **Usage**
 * Opens the selection dialog that enables the user to select an instance of a subentity.
 – **Parameters**
 * entData - SubentityOT, data of the detail subentity to select.
-
- *showSubentityDialog*
 public void **showSubentityDialog**(com.imagos.axilice.core.installer.SubentityOT entData, long selEntity, boolean readonly)
 – **Usage**
 * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.
 – **Parameters**
 * entData - SubentityOT, data of the detail subentity to select.
 * selEntity - long, id of the selected instance of the subentity.
 * readonly - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.
-
- *showSubentityDialog*
 public void **showSubentityDialog**(com.imagos.axilice.core.installer.SubentityOT entData, java.lang.String parentGuiName, long selEntity, boolean readonly)
 – **Usage**
 * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.
 – **Parameters**
 * entData - SubentityOT, data of the detail subentity to select.
 * parentGuiName - String,
 * selEntity - long, id of the selected instance of the subentity.
 * readonly - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.
-
- *showSubentitySelectionDialog*
 public void **showSubentitySelectionDialog**(
 com.imagos.axilice.core.installer.SubentityOT entData)

- **Usage**
 - * Opens the selection dialog that enables the user to select an instance of a multiple selection detail subentity.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to select.
-
- *undoEntityEdition*
public void undoEntityEdition()
 - *updateEntityData*
public abstract void updateEntityData()
 - **Usage**
 - * Updates the selected instance (registry) data with the content panel's data typed by the user.
-
- *viewEntity*
public abstract OperationResult viewEntity()
 - **Usage**
 - * View entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), in read-only mode.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

27.1.7 CLASS PartnerClient

DECLARATION

```
public class PartnerClient
extends com.imagos.axilice.client.gui.common.CommonClient
```

CONSTRUCTORS

- *PartnerClient*
public PartnerClient()

METHODS

- *build_extension_fields_view*
public void build_extension_fields_view()
- *buildSelectionSets*
public void buildSelectionSets()
- *fillEntityData*
public void fillEntityData()
- *getEntityData*
public BaseVO getEntityData()

- *getWidgets*
public void **getWidgets**()
- *init_specific_components*
public void **init_specific_components**()
- *on_compute_id_number_button_clicked*
public void **on_compute_id_number_button_clicked**(Event e)
- *printEntity*
public OperationResult **printEntity**()
- *setEditionMode*
public void **setEditionMode**(boolean editionMode)
- *setWidgetsValues*
public void **setWidgetsValues**(com.imagos.axilice.core.common.BaseVO ent)
- *validateEntity*
public OperationResult **validateEntity**()

METHODS INHERITED FROM CLASS com.imagos.axilice.client.gui.common.CommonClient

(in 17.1.2, page 402)

- *addAxiliceEventListener*
public void **addAxiliceEventListener**(
com.imagos.axilice.client.gui.AxiliceEventListener l)
- *addEntity*
public ArrayList **addEntity**()
- *addExistingSubentity*
public void **addExistingSubentity**(com.imagos.axilice.core.installer.SubentityOT
ent)
 - **Usage**
 - * Shows a dialog for selection of a new subentity
 - **Parameters**
 - * **ent** - SubentityOT. Encapsulate data type of subentity
- *addSubentity*
public void **addSubentity**(com.imagos.axilice.core.common.BaseVO subentity,
com.imagos.axilice.core.common.MultiSelectionSubentityOT subE)
 - **Usage**
 - * Adds a existent entity to a multiselection list
 - **Parameters**
 - * **subentity** - Subentity to be added
 - * **subE** - Object that encapsulates information about the subentity
- *addTrazabilityTree*
public void **addTrazabilityTree**(TabFolder tabFolder)
- *build_extension_fields_view*
public abstract void **build_extension_fields_view**()
 - **Usage**
 - * Build the view of the extension fields.

It must implement only those clients who require it

-
- *buildEntity*
 public AxiliceEntityView buildEntity()
 - *buildEntityDetail*
 public AxiliceEntityView buildEntityDetail(
 com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData)
 - *buildEntitySelectionView*
 public AxiliceEntitySelectionView buildEntitySelectionView()
 - Usage
 - * Build view of selection entities
 - Returns - AxilicePartnerSelectionView. SelectionView created
-
- *buildSelectionSets*
 public abstract void buildSelectionSets()
 - Usage
 - * Build 2 sets of SubentityOT: lSelectionSubentities and lMultiSelectionSubentities

These sets are used to connect the widgets with the handlers for the selection buttons and the multiseleccion panels

It must implement only those clients who require it
-
- *buildSubentityMultiSelection*
 public void buildSubentityMultiSelection()
 - *clearEntityData*
 public void clearEntityData()
 - *clearParentField*
 public OperationResult clearParentField(java.util.ArrayList selRegistries,
 java.lang.String parentField)
 - Usage
 - * Clears the value of the field indicated
 - Parameters
 - * selRegistries - ArrayList. Index list of records to process
 - * parentField - Name of the field to clear
 - Returns - OperationResult. Result of operation
-
- *delEntity*
 public OperationResult delEntity()
 - *delEntity*
 public OperationResult delEntity(java.util.ArrayList selRegistries)
 - *delEntity*
 public OperationResult delEntity(com.imagos.axilice.core.common.BaseVO b)
 - Usage
 - * Delete a entity from database
 - Parameters
 - * b - BaseVO. Entity to delete
 - Returns - OperationResult. Result of the operation
-
- *delSubentities*
 public OperationResult delSubentities(java.util.ArrayList selRegistries)
 - Usage

* Delete a set of subentities from record list of controller, not from database

Set the parent field of subentity to null

– **Parameters**

* **selRegistries** - List of index to delete

– **Returns** - **OperationResult**. Result of the operation

• *editEntity*

public **OperationResult** **editEntity**()

• *fillEntityData*

public void **fillEntityData**()

• *fillEntityExtensionTable*

public void **fillEntityExtensionTable**(java.lang.Long id)

– **Usage**

* Retrieve extension field values from the server for a given entity and displays them in the view

– **Parameters**

* **id** - Long. Key of the entity whose values are to be displayed

• *fillEntityTable*

public void **fillEntityTable**()

– **Usage**

* Load records from database and fill the entity view table and controller record list

• *fillEntityTable*

public void **fillEntityTable**(java.util.ArrayList lR)

– **Usage**

* Fill the entity view table with the records given

– **Parameters**

* **lR** - ArrayList. Records to be displayed

• *getClassEntity*

public Class **getClassEntity**()

– **Returns** - Class of the entity for which the controller is built

• *getDialogName*

public String **getDialogName**()

• *getEditGladePath*

public String **getEditGladePath**()

• *getEditionGui*

public Gui **getEditionGui**()

– **Usage**

* Returns the edition Gui instance.

– **Returns** - Gui, entity edition Gui.

• *getEditWindow*

public SwtTopLevelWindow **getEditWindow**()

• *getEntityData*

public abstract BaseVO **getEntityData**()

• *getEntityFields*

public ArrayList **getEntityFields**()

- *getLogger*
public Logger getLogger()
- *getParentEntityLauncher*
public EntityLauncher getParentEntityLauncher()
 – **Returns** - EntityLauncher of parent entity if is a embedded entity or null otherwise
- *getRegistries*
public Collection getRegistries()
 – **Returns** - lRegistries
- *getRegistries*
public List getRegistries(java.util.ArrayList lIds)
 – **Usage**
 * Get a list of entities stored by the controller
 – **Parameters**
 * lIds - Index list
 – **Returns** - List of entities
- *getView*
public AxiliceEntityView getView()
 – **Returns** - AxiliceEntityView. Main view or Detail view if is a embedded entity.
- *getWidgets*
public abstract void getWidgets()
 – **Usage**
 * Get the widgets from the glade and stored in local variables.

 Must be overwritten
- *init_specific_components*
public abstract void init_specific_components()
 – **Usage**
 * Initializes the specific components of the client
- *printEntity*
public OperationResult printEntity()
- *printEntity*
public OperationResult printEntity(java.lang.String reportPath, java.util.HashMap parameters)
- *queryEntity*
public ArrayList queryEntity()
- *queryEntity*
public ArrayList queryEntity(java.lang.String fieldId, java.lang.String value)
- *queryEntityWithFormat*
public ArrayList queryEntityWithFormat(java.lang.String fieldId, java.lang.String value)
- *queryFreeEntities*
public ArrayList queryFreeEntities(com.imagos.axilice.core.common.MultiSelectionSubentityOT mSubE)
 – **Usage**
 * Search entities not associated with a parent entity (null parent field)
 – **Parameters**

- * mSubE - SubentityOT for the parent entity that must be null
 - **Returns** - ArrayList. List of entities found

- *querySubentities*
 public ArrayList **querySubentities**(com.imagos.axilice.core.installer.SubentityOT
 entData)

- *refreshEntity*
 public OperationResult **refreshEntity**()

- *refreshEntity*
 public BaseVO **refreshEntity**(com.imagos.axilice.core.common.BaseVO entity)

- *refreshParentSelectedEntity*
 public OperationResult **refreshParentSelectedEntity**()
 - **Usage**
 - * Reload from database the value of the parent entity of the selected entity
 - **Returns** - OperationResult. The result of the operation.

- *removeRegistries*
 public OperationResult **removeRegistries**(java.util.ArrayList selRegistries,
 java.lang.String parentField)
 - **Usage**
 - * Deletes a list of the records stored by the controller. Also clean your parent fields and updates the database
 - **Parameters**
 - * selRegistries - Index list to remove
 - * parentField - Field name that links to the parent entity
 - **Returns** - OperationResult. Result of the operation

- *saveEntity*
 public OperationResult **saveEntity**()

- *saveEntity*
 public OperationResult **saveEntity**(java.util.ArrayList lIds)

- *saveEntity*
 public OperationResult **saveEntity**(com.imagos.axilice.core.common.BaseVO b)
 - **Usage**
 - * Save or update a entity to database
 - **Parameters**
 - * b - BaseVO. Entity to save or update
 - **Returns** - OperationResult. Result of the operation

- *selectEntity*
 public void **selectEntity**(java.lang.Long idEntity)

- *selectSubentity*
 public void **selectSubentity**(java.lang.String parentFieldId,
 com.imagos.axilice.core.common.BaseVO subentityOT)

- *selectTreeItem*
 public void **selectTreeItem**(java.lang.Long id)
 - **Usage**
 - * Select a item in the entity tree. No me gusta esto aqui
 - **Parameters**
 - * id - Long. Index of item to select

- *setDialogName*
public void setDialogName(java.lang.String dialogName)
- *setEditGladePath*
public void setEditGladePath(java.lang.String editGladePath)
- *setEditionMode*
public abstract void setEditionMode(boolean enabled)
- *setRegistries*
public void setRegistries(java.util.Collection collection)
 - **Usage**
 - * Setter for IRegistries

- *setWidgetsValues*
public abstract void setWidgetsValues(com.imagos.axilice.core.common.BaseVO entity)
 - **Usage**
 - * Set the values of the widgets on the edit dialog.
 - Must be overwritten.
 - **Parameters**
 - * **entity** - BaseVO. Entity that contains the values to be displayed

- *showErrorMessages*
public void showErrorMessages(com.imagos.axilice.core.util.OperationResult result)

- *updateEntityData*
public void updateEntityData()
- *updateRegistry*
public void updateRegistry(com.imagos.axilice.core.common.BaseVO ent)
 - **Usage**
 - * Add or update an entity (to the controller records)
 - **Parameters**
 - * **ent** - Entity to add or update

- *updateViews*
public void updateViews(boolean updateFromParent)
 - **Usage**
 - * Force the update of the views associated
 - **Parameters**
 - * **updateFromParent** - Indicates whether the view must be updated by reading the values from the parent entity. takes effect only when the entity is detail (is embedded in another)

- *validateEntity*
public abstract OperationResult validateEntity()
 - **Usage**
 - * Validate the values of the widgets in the edit dialog.
 - Must be overwritten.
 - **Returns** - OperationResult. Result of the validation.

- *viewEntity*
public OperationResult viewEntity()

METHODS INHERITED FROM CLASS com.imagos.axilice.core.installer.EntityLauncher

(in 24.2.2, page 532)

- *addEntity*
 public abstract ArrayList **addEntity**()
 — **Usage**
 * Add entity button clicked handler. Show the entity content screen ready to start inserting data for a new instance of the entity (new registry).
 — **Returns** - ArrayList<ResultMessage>, list of errors if any has taken place, null in any other case.

- *buildEditEntityGUI*
 public void **buildEditEntityGUI**()
 — **Usage**
 * Builds the edit entity dialog window from the corresponding glade interface file. Stores the entity dialog instance for posterior accesses.

- *buildEntity*
 public AxiliceEntityView **buildEntity**()
 — **Usage**
 * Builds the entity GUI and show it on screen to the user.

- *buildEntity*
 public AxiliceDetailEntityView **buildEntity**(SwtWidget **parent**,
 com.imagos.axilice.core.common.MultiSelectionSubentityOT **subentityData**,
 java.lang.String **rootName**)
 — **Usage**
 * Builds the entity GUI and show it on screen to the user. The entity GUI will be embedded into the parent composite widget.
 — **Parameters**
 * **parent** - Composite, parent holder for embedding the entity view.
 * **subentityData** - SubentityOT, data of the subentity to build if the entity launcher will be used to handle a multi-selection detail entity view.
 * **rootName** - String, name of the root widget that will be embedded into the master entity edition panel.
 — **Returns** - AxiliceDetailEntityView, the entity detail view built.

- *cancelEntityEdition*
 public void **cancelEntityEdition**()

- *clearEntityData*
 public abstract void **clearEntityData**()
 — **Usage**
 * Clears the entity content panel by resetting the content widgets.

- *delEntity*
 public abstract OperationResult **delEntity**()
 — **Usage**
 * Delete entity button clicked handler. Deletes the selected instance of the entity (selected registry) from the database.
 — **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *delEntity*

```
public abstract OperationResult delEntity( java.util.ArrayList lSelRegistries )
```

- **Usage**

- * Delete entity button clicked handler. Deletes the selected instances of the entity (selected registries) from the database. Its a multiple delete method.

- **Parameters**

- * lSelRegistries - ArrayList<Long>, list of selected instances of the entity (selected registries). The list will contain only the internal id of the selected registries.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *deselectSubentity*

```
public void deselectSubentity( java.lang.String parentFieldId,  
com.imagos.axilice.core.common.BaseVO subentityOT )
```

- **Usage**

- * Updates an entity instance by removing the selected value for a subentity field.

- **Parameters**

- * parentFieldId - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.

- * subentityOT - BaseVO, subentity's deselected instance (registry) value.

- *editEntity*

```
public abstract OperationResult editEntity( )
```

- **Usage**

- * Edit entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), and allows the user to modify any content of the entity registry.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *fillEntityData*

```
public abstract void fillEntityData( )
```

- **Usage**

- * Fills the entity content panel with the data of the selected instance (registry) of the entity.

- *fireExceptionAndExit*

```
public void fireExceptionAndExit( java.lang.String msg )
```

- **Usage**

- * Fire a exception and stop the program

- **Parameters**

- * msg -

- *formatRegistries*

```
public ArrayList formatRegistries( )
```

- *formatRegistries*

```
public ArrayList formatRegistries( java.util.ArrayList lRegistries )
```

- *getDialog*

```
public SwtDialog getDialog( java.lang.String dialogName )
```

- *getEntity*

```
public EntityOT getEntity( )
```

- **Usage**

- * Gets the entity data.

- **Returns** - EntityOT, entity data.
- getEntityData
 public abstract BaseVO **getEntityData**()
 - **Usage**
 - * Gets the content panel's data typed by the user formatted as an entity VO object.
- getEntityEditViewName
 public String **getEntityEditViewName**(java.lang.String parentGuiName)
- getEntityFields
 public abstract ArrayList **getEntityFields**()
 - **Usage**
 - * Gets the list of defined fields for an entity
 - **Returns** - ArrayList, list of entity fields.
- getEntityViewName
 public String **getEntityViewName**(java.lang.String parentGuiName)
- getLRegistries
 public ArrayList **getLRegistries**()
 - **Usage**
 - * Getter for lRegistries
- getRegistry
 public BaseVO **getRegistry**(long id)
- getRevision
 public static String **getRevision**()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
- getSelectedEntity
 public BaseVO **getSelectedEntity**()
 - **Usage**
 - * Gets the selected instance (registry) of the entity.
 - **Returns** - BaseVO, selected instance (registry) of the entity.
- getSelectedId
 public Long **getSelectedId**()
 - **Usage**
 - * Gets the internal id of the selected entity registry.
 - **Returns** - Long, internal id of the selected entity registry, null in case no entity registry has been selected yet.
- getSwtWidget
 public SwtWidget **getSwtWidget**(java.lang.String widgetName)
- getWidget
 public Widget **getWidget**(java.lang.String widgetName)
- initEntityLauncher
 public void **initEntityLauncher**(Logger logger, Gui parentGui, SwtWidget parentPanel)
 - **Usage**
 - * Inits the entity launcher context and prepares the launcher to be started.
 - **Parameters**

- * **logger** - Logger, logger instance for printing logging messages.
- * **parentGuiName** - String, name of the JavaGlade instance that built the parent where the entity will be embedded.
- * **parentPanel** - Composite, parent panel where the entity contents will be embedded.

- *isDetail*

public boolean **isDetail**()

– **Usage**

- * Indicates whether is a main entity or is embedded in another

- *on_entitySelectionButton_clicked*

public void **on_entitySelectionButton_clicked**(Event e)

– **Usage**

- * Handler for the subentity selection buttons (magnifying glass buttons) clicked. Shows the selection dialog for selecting a valid instance of a subentity.

– **Parameters**

- * **e** - Event, magnifying glass button clicked event data.

- *printEntity*

public abstract **OperationResult printEntity**()

– **Usage**

- * Print entity button clicked handler. Builds the report with the data of the selected instance of the entity (selected registry), and allows the user to print the elaborated report.

– **Returns** - **OperationResult**, result of the delete operation and list of messages generated during the operation.

- *printWidgets*

public void **printWidgets**()

- *queryEntity*

public abstract **ArrayList queryEntity**()

– **Usage**

- * Search entity button clicked handler. Searches the system for the entity instances (registries) available.

– **Returns** - **ArrayList**, list of entity instances (registries) taken as the result of the query operation.

- *queryEntity*

public abstract **ArrayList queryEntity**(java.lang.String **fieldId**, java.lang.String **value**)

– **Usage**

- * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria.

– **Parameters**

- * **fieldId** - String, name of the entity field that will be queried.
- * **value** - String, value of the entity field that will be searched for.

– **Returns** - **ArrayList**, list of entity instances (registries) taken as the result of the query operation.

- *queryEntityWithFormat*

public abstract **ArrayList queryEntityWithFormat**(java.lang.String **fieldId**, java.lang.String **value**)

– **Usage**

- * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria and show them (formatted as a list) to the user in the entity's screen.
 - **Parameters**
 - * **fieldId** - String, name of the entity field that will be queried.
 - * **value** - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList<RegistryOT>, list of entity instances (registries) taken as the result of the query operation.

- *querySubentities*

```
public abstract ArrayList querySubentities(
com.imagos.axilice.core.installer.SubentityOT  entData )
```

 - **Usage**
 - * Queries the selected entity for all the detail subentities associated with it.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to query for.
 - **Returns** - ArrayList, list of detail subentity registries assigned to the master detail registry.

- *saveEntity*

```
public abstract OperationResult saveEntity( )
```

 - **Usage**
 - * Save entity button clicked handler. Update the system with the data of the edited entity registry. It's called for saving new entity registries (add a new registry) or for updating the contents of existing entity registries.
 - **Returns** - OperationResult, Result of the operation and list of messages generated through the operation.

- *saveEntityRegistry*

```
public void saveEntityRegistry( )
```
- *selectEntity*

```
public abstract void selectEntity( java.lang.Long  idEntity )
```

 - **Usage**
 - * Marks an instance (registry) of the entity as the selected.
 - **Parameters**
 - * **idEntity** - Long, internal id of the selected instance (registry).

- *selectSubentity*

```
public void selectSubentity( java.lang.String  parentFieldId,
com.imagos.axilice.core.common.BaseVO  subentityOT )
```

 - **Usage**
 - * Updates an entity instance by setting up the selected value for a subentity field.
 - **Parameters**
 - * **parentFieldId** - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * **subentityOT** - BaseVO, subentity's selected instance (registry) value.

- *setEntity*

```
public void setEntity( com.imagos.axilice.core.installer.EntityOT  entity )
```

 - **Usage**
 - * Sets up the entity name.
 - **Parameters**
 - * **name** - String, entity name.

- *setRegistries*

```
public void setRegistries( java.util.ArrayList lRegistries )
```

- Usage

- * Sets the list of registries that are showed in the entity view controlled by the Entity Launcher instance. It's used for detail entities where queries are made in the master entity launcher.

- Parameters

- * lRegistries - ArrayList<BaseVO>, list of shown registries of the entity.

- *showSelectionDialog*

```
public void showSelectionDialog( com.imagos.axilice.core.installer.SubentityOT entData )
```

- Usage

- * Opens the selection dialog that enables the user to select an instance of a subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.

- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT entData, long selEntity, boolean readonly )
```

- Usage

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.
 - * selEntity - long, id of the selected instance of the subentity.
 - * readonly - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT entData, java.lang.String parentGuiName, long selEntity, boolean readonly )
```

- Usage

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.
 - * parentGuiName - String,
 - * selEntity - long, id of the selected instance of the subentity.
 - * readonly - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

- *showSubentitySelectionDialog*

```
public void showSubentitySelectionDialog( com.imagos.axilice.core.installer.SubentityOT entData )
```

- Usage

- * Opens the selection dialog that enables the user to select an instance of a multiple selection detail subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.

- *undoEntityEdition*

```
public void undoEntityEdition( )
```

- *updateEntityData*
 public abstract void **updateEntityData**()
 – **Usage**
 * Updates the selected instance (registry) data with the content panel’s data typed by the user.

- *viewEntity*
 public abstract **OperationResult viewEntity**()
 – **Usage**
 * View entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), in read-only mode.
 – **Returns** - **OperationResult**, result of the delete operation and list of messages generated during the operation.

27.1.8 CLASS PartnerGroupClient

DECLARATION

```
public class PartnerGroupClient
extends com.imagos.axilice.client.gui.common.CommonClient
```

CONSTRUCTORS

- *PartnerGroupClient*
 public **PartnerGroupClient**()

METHODS

- *build_extension_fields_view*
 public void **build_extension_fields_view**()

- *buildSelectionSets*
 public void **buildSelectionSets**()

- *fillEntityData*
 public void **fillEntityData**()

- *getEntityData*
 public **BaseVO getEntityData**()

- *getWidgets*
 public void **getWidgets**()

- *init_specific_components*
 public void **init_specific_components**()

- *setEditionMode*
 public void **setEditionMode**(boolean enabled)

- *setWidgetsValues*

```
public void setWidgetsValues( com.imagos.axilice.core.common.BaseVO entity
)
```

- *validateEntity*

```
public OperationResult validateEntity( )
```

METHODS INHERITED FROM CLASS com.imagos.axilice.client.gui.common.CommonClient

(in 17.1.2, page 402)

- *addAxiliceEventListener*

```
public void addAxiliceEventListener(
com.imagos.axilice.client.gui.AxiliceEventListener l )
```

- *addEntity*

```
public ArrayList addEntity( )
```

- *addExistingSubentity*

```
public void addExistingSubentity( com.imagos.axilice.core.installer.SubentityOT
ent )
```

– Usage

* Shows a dialog for selection of a new subentity

– Parameters

* ent - SubentityOT. Encapsulate data type of subentity

- *addSubentity*

```
public void addSubentity( com.imagos.axilice.core.common.BaseVO subentity,
com.imagos.axilice.core.common.MultiSelectionSubentityOT subE )
```

– Usage

* Adds a existent entity to a multiselection list

– Parameters

* subentity - Subentity to be added

* subE - Object that encapsulates information about the subentity

- *addTrazabilityTree*

```
public void addTrazabilityTree( TabFolder tabFolder )
```

- *build_extension_fields_view*

```
public abstract void build_extension_fields_view( )
```

– Usage

* Build the view of the extension fields.

It must implement only those clients who require it

- *buildEntity*

```
public AxiliceEntityView buildEntity( )
```

- *buildEntityDetail*

```
public AxiliceEntityView buildEntityDetail(
com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData )
```

- *buildEntitySelectionView*

```
public AxiliceEntitySelectionView buildEntitySelectionView( )
```

– Usage

* Build view of selection entities

– Returns - AxilicePartnerSelectionView. SelectionView created

- *buildSelectionSets*
 public abstract void **buildSelectionSets**()
 – **Usage**
 * Build 2 sets of SubentityOT: lSelectionSubentities and lMultiSelectionSubentities
 These sets are used to connect the widgets with the handlers for the selection buttons and the multiseleccion panels
 It must implement only those clients who require it

- *buildSubentityMultiSelection*
 public void **buildSubentityMultiSelection**()
- *clearEntityData*
 public void **clearEntityData**()
- *clearParentField*
 public OperationResult **clearParentField**(java.util.ArrayList selRegistries,
 java.lang.String parentField)
 – **Usage**
 * Clears the value of the field indicated
 – **Parameters**
 * selRegistries - ArrayList. Index list of records to process
 * parentField - Name of the field to clear
 – **Returns** - OperationResult. Result of operation

- *delEntity*
 public OperationResult **delEntity**()
- *delEntity*
 public OperationResult **delEntity**(java.util.ArrayList selRegistries)
- *delEntity*
 public OperationResult **delEntity**(com.imagos.axilice.core.common.BaseVO b)
 – **Usage**
 * Delete a entity from database
 – **Parameters**
 * b - BaseVO. Entity to delete
 – **Returns** - OperationResult. Result of the operation

- *delSubentities*
 public OperationResult **delSubentities**(java.util.ArrayList selRegistries)
 – **Usage**
 * Delete a set of subentities from record list of controller, not from database
 Set the parent field of subentity to null
 – **Parameters**
 * selRegistries - List of index to delete
 – **Returns** - OperationResult. Result of the operation

- *editEntity*
 public OperationResult **editEntity**()
- *fillEntityData*
 public void **fillEntityData**()
- *fillEntityExtensionTable*
 public void **fillEntityExtensionTable**(java.lang.Long id)

- **Usage**
 - * Retrieve extension field values from the server for a given entity and displays them in the view
- **Parameters**
 - * **id** - Long. Key of the entity whose values are to be displayed

- *fillEntityTable*
public void fillEntityTable()
 - **Usage**
 - * Load records from database and fill the entity view table and controller record list

- *fillEntityTable*
public void fillEntityTable(java.util.ArrayList lR)
 - **Usage**
 - * Fill the entity view table with the records given
 - **Parameters**
 - * **lR** - ArrayList. Records to be displayed

- *getClassEntity*
public Class getClassEntity()
 - **Returns** - Class of the entity for which the controller is built

- *getDialogName*
public String getDialogName()
- *getEditGladePath*
public String getEditGladePath()
- *getEditionGui*
public Gui getEditionGui()
 - **Usage**
 - * Returns the edition Gui instance.
 - **Returns** - Gui, entity edition Gui.

- *getEditWindow*
public SwtTopLevelWindow getEditWindow()
- *getEntityData*
public abstract BaseVO getEntityData()
- *getEntityFields*
public ArrayList getEntityFields()
- *getLogger*
public Logger getLogger()
- *getParentEntityLauncher*
public EntityLauncher getParentEntityLauncher()
 - **Returns** - EntityLauncher of parent entity if is a embedded entity or null otherwise

- *getRegistries*
public Collection getRegistries()
 - **Returns** - lRegistries

- *getRegistries*
public List getRegistries(java.util.ArrayList lIds)
 - **Usage**
 - * Get a list of entities stored by the controller

- **Parameters**
 - * lIds - Index list
 - **Returns** - List of entities

- *getView*
public AxiliceEntityView getView()
 - **Returns** - AxiliceEntityView. Main view or Detail view if is a embedded entity.

- *getWidgets*
public abstract void getWidgets()
 - **Usage**
 - * Get the widgets from the glade and stored in local variables.
 - Must be overwritten

- *init_specific_components*
public abstract void init_specific_components()
 - **Usage**
 - * Initializes the specific components of the client

- *printEntity*
public OperationResult printEntity()

- *printEntity*
public OperationResult printEntity(java.lang.String reportPath, java.util.HashMap parameters)

- *queryEntity*
public ArrayList queryEntity()

- *queryEntity*
public ArrayList queryEntity(java.lang.String fieldId, java.lang.String value)

- *queryEntityWithFormat*
public ArrayList queryEntityWithFormat(java.lang.String fieldId, java.lang.String value)

- *queryFreeEntities*
public ArrayList queryFreeEntities(com.imagos.axilice.core.common.MultiSelectionSubentityOT mSubE)
 - **Usage**
 - * Search entities not associated with a parent entity (null parent field)
 - **Parameters**
 - * mSubE - SubentityOT for the parent entity that must be null
 - **Returns** - ArrayList. List of entities found

- *querySubentities*
public ArrayList querySubentities(com.imagos.axilice.core.installer.SubentityOT entData)

- *refreshEntity*
public OperationResult refreshEntity()

- *refreshEntity*
public BaseV0 refreshEntity(com.imagos.axilice.core.common.BaseV0 entity)

- *refreshParentSelectedEntity*
public OperationResult refreshParentSelectedEntity()
 - **Usage**
 - * Reload from database the value of the parent entity of the selected entity

- **Returns** - OperationResult. The result of the operation.
- *removeRegistries*

```
public OperationResult removeRegistries( java.util.ArrayList< Integer> selRegistries,
java.lang.String parentField )
```

 - **Usage**
 - * Deletes a list of the records stored by the controller. Also clean your parent fields and updates the database
 - **Parameters**
 - * selRegistries - Index list to remove
 - * parentField - Field name that links to the parent entity
 - **Returns** - OperationResult. Result of the operation
- *saveEntity*

```
public OperationResult saveEntity( )
```
- *saveEntity*

```
public OperationResult saveEntity( java.util.ArrayList< Integer> lIds )
```
- *saveEntity*

```
public OperationResult saveEntity( com.imagos.axilice.core.common.BaseVO b )
```

 - **Usage**
 - * Save or update a entity to database
 - **Parameters**
 - * b - BaseVO. Entity to save or update
 - **Returns** - OperationResult. Result of the operation
- *selectEntity*

```
public void selectEntity( java.lang.Long idEntity )
```
- *selectSubentity*

```
public void selectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```
- *selectTreeItem*

```
public void selectTreeItem( java.lang.Long id )
```

 - **Usage**
 - * Select a item in the entity tree. No me gusta esto aqui
 - **Parameters**
 - * id - Long. Index of item to select
- *setDialogName*

```
public void setDialogName( java.lang.String dialogName )
```
- *setEditGladePath*

```
public void setEditGladePath( java.lang.String editGladePath )
```
- *setEditionMode*

```
public abstract void setEditionMode( boolean enabled )
```
- *setRegistries*

```
public void setRegistries( java.util.Collection< Integer> collection )
```

 - **Usage**
 - * Setter for lRegistries
- *setWidgetsValues*

```
public abstract void setWidgetsValues( com.imagos.axilice.core.common.BaseVO
entity )
```

 - **Usage**

- * Set the values of the widgets on the edit dialog.
- Must be overwritten.
- **Parameters**
 - * **entity** - BaseVO. Entity that contains the values to be displayed
- *showErrorMessages*

```
public void showErrorMessages( com.imagos.axilice.core.util.OperationResult  result )
```

- *updateEntityData*

```
public void updateEntityData( )
```

- *updateRegistry*

```
public void updateRegistry( com.imagos.axilice.core.common.BaseVO  ent )
```

 - **Usage**
 - * Add or update an entity (to the controller records)
 - **Parameters**
 - * **ent** - Entity to add or update
- *updateViews*

```
public void updateViews( boolean  updateFromParent )
```

 - **Usage**
 - * Force the update of the views associated
 - **Parameters**
 - * **updateFromParent** - Indicates whether the view must be updated by reading the values from the parent entity. takes effect only when the entity is detail (is embedded in another)
- *validateEntity*

```
public abstract OperationResult validateEntity( )
```

 - **Usage**
 - * Validate the values of the widgets in the edit dialog.
 - Must be overwritten.
 - **Returns** - OperationResult. Result of the validation.
- *viewEntity*

```
public OperationResult viewEntity( )
```

METHODS INHERITED FROM CLASS com.imagos.axilice.core.installer.EntityLauncher

(in 24.2.2, page 532)

- *addEntity*

```
public abstract ArrayList addEntity( )
```

 - **Usage**
 - * Add entity button clicked handler. Show the entity content screen ready to start inserting data for a new instance of the entity (new registry).
 - **Returns** - ArrayList<ResultMessage>, list of errors if any has taken place, null in any other case.
- *buildEditEntityGUI*

```
public void buildEditEntityGUI( )
```

 - **Usage**

- * Builds the edit entity dialog window from the corresponding glade interface file. Stores the entity dialog instance for posterior accesses.

- *buildEntity*

```
public AxiliceEntityView buildEntity( )
```

- **Usage**

- * Builds the entity GUI and show it on screen to the user.

- *buildEntity*

```
public AxiliceDetailEntityView buildEntity( SwtWidget parent,
com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData,
java.lang.String rootName )
```

- **Usage**

- * Builds the entity GUI and show it on screen to the user. The entity GUI will be embedded into the parent composite widget.

- **Parameters**

- * **parent** - Composite, parent holder for embedding the entity view.
- * **subentityData** - SubentityOT, data of the subentity to build if the entity launcher will be used to handle a multi-selection detail entity view.
- * **rootName** - String, name of the root widget that will be embedded into the master entity edition panel.

- **Returns** - AxiliceDetailEntityView, the entity detail view built.

- *cancelEntityEdition*

```
public void cancelEntityEdition( )
```

- *clearEntityData*

```
public abstract void clearEntityData( )
```

- **Usage**

- * Clears the entity content panel by resetting the content widgets.

- *delEntity*

```
public abstract OperationResult delEntity( )
```

- **Usage**

- * Delete entity button clicked handler. Deletes the selected instance of the entity (selected registry) from the database.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *delEntity*

```
public abstract OperationResult delEntity( java.util.ArrayList lSelRegistries )
```

- **Usage**

- * Delete entity button clicked handler. Deletes the selected instances of the entity (selected registries) from the database. Its a multiple delete method.

- **Parameters**

- * **lSelRegistries** - ArrayList<Long>, list of selected instances of the entity (selected registries). The list will contain only the internal id of the selected registries.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *deselectSubentity*

```
public void deselectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

- **Usage**

- * Updates an entity instance by removing the selected value for a subentity field.
- **Parameters**
 - * **parentFieldId** - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * **subentityOT** - BaseVO, subentity's deselected instance (registry) value.

- *editEntity*
 public abstract OperationResult **editEntity**()
 - **Usage**
 - * Edit entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), and allows the user to modify any content of the entity registry.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *fillEntityData*
 public abstract void **fillEntityData**()
 - **Usage**
 - * Fills the entity content panel with the data of the selected instance (registry) of the entity.

- *fireExceptionAndExit*
 public void **fireExceptionAndExit**(java.lang.String msg)
 - **Usage**
 - * Fire a exception and stop the program
 - **Parameters**
 - * **msg** -

- *formatRegistries*
 public ArrayList **formatRegistries**()

- *formatRegistries*
 public ArrayList **formatRegistries**(java.util.ArrayList lRegistries)

- *getDialog*
 public SwtDialog **getDialog**(java.lang.String dialogName)

- *getEntity*
 public EntityOT **getEntity**()
 - **Usage**
 - * Gets the entity data.
 - **Returns** - EntityOT, entity data.

- *getEntityData*
 public abstract BaseVO **getEntityData**()
 - **Usage**
 - * Gets the content panel's data typed by the user formatted as an entity VO object.

- *getEntityEditViewName*
 public String **getEntityEditViewName**(java.lang.String parentGuiName)

- *getEntityFields*
 public abstract ArrayList **getEntityFields**()
 - **Usage**
 - * Gets the list of defined fields for an entity
 - **Returns** - ArrayList, list of entity fields.

- *getEntityViewName*
public String **getEntityViewName**(java.lang.String parentGuiName)
- *getLRegistries*
public ArrayList **getLRegistries**()
 - **Usage**
 - * Getter for IRegistries
- *getRegistry*
public BaseVO **getRegistry**(long id)
- *getRevision*
public static String **getRevision**()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
- *getSelectedEntity*
public BaseVO **getSelectedEntity**()
 - **Usage**
 - * Gets the selected instance (registry) of the entity.
 - **Returns** - BaseVO, selected instance (registry) of the entity.
- *getSelectedId*
public Long **getSelectedId**()
 - **Usage**
 - * Gets the internal id of the selected entity registry.
 - **Returns** - Long, internal id of the selected entity registry, null in case no entity registry has been selected yet.
- *getSwtWidget*
public SwtWidget **getSwtWidget**(java.lang.String widgetName)
- *getWidget*
public Widget **getWidget**(java.lang.String widgetName)
- *initEntityLauncher*
public void **initEntityLauncher**(Logger logger, Gui parentGui, SwtWidget parentPanel)
 - **Usage**
 - * Inits the entity launcher context and prepares the launcher to be started.
 - **Parameters**
 - * **logger** - Logger, logger instance for printing logging messages.
 - * **parentGuiName** - String, name of the JavaGlade instance that built the parent where the entity will be embedded.
 - * **parentPanel** - Composite, parent panel where the entity contents will be embedded.
- *isDetail*
public boolean **isDetail**()
 - **Usage**
 - * Indicates whether is a main entity or is embedded in another
- *on_entitySelectionButton_clicked*
public void **on_entitySelectionButton_clicked**(Event e)
 - **Usage**
 - * Handler for the subentity selection buttons (magnifying glass buttons) clicked. Shows the selection dialog for selecting a valid instance of a subentity.

- **Parameters**
 - * **e** - Event, magnifying glass button clicked event data.

- *printEntity*

```
public abstract OperationResult printEntity( )
```

 - **Usage**
 - * Print entity button clicked handler. Builds the report with the data of the selected instance of the entity (selected registry), and allows the user to print the elaborated report.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *printWidgets*

```
public void printWidgets( )
```

- *queryEntity*

```
public abstract ArrayList queryEntity( )
```

 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) available.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.

- *queryEntity*

```
public abstract ArrayList queryEntity( java.lang.String fieldId, java.lang.String value )
```

 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria.
 - **Parameters**
 - * **fieldId** - String, name of the entity field that will be queried.
 - * **value** - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.

- *queryEntityWithFormat*

```
public abstract ArrayList queryEntityWithFormat( java.lang.String fieldId, java.lang.String value )
```

 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria and show them (formatted as a list) to the user in the entity's screen.
 - **Parameters**
 - * **fieldId** - String, name of the entity field that will be queried.
 - * **value** - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList<RegistryOT>, list of entity instances (registries) taken as the result of the query operation.

- *querySubentities*

```
public abstract ArrayList querySubentities( com.imagos.axilice.core.installer.SubentityOT entData )
```

 - **Usage**
 - * Queries the selected entity for all the detail subentities associated with it.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to query for.

- **Returns** - ArrayList, list of detail subentity registries assigned to the master detail registry.

- *saveEntity*
public abstract OperationResult saveEntity()
 - **Usage**
 - * Save entity button clicked handler. Update the system with the data of the edited entity registry. It's called for saving new entity registries (add a new registry) or for updating the contents of existing entity registries.
 - **Returns** - OperationResult, Result of the operation and list of messages generated through the operation.

- *saveEntityRegistry*
public void saveEntityRegistry()
- *selectEntity*
public abstract void selectEntity(java.lang.Long idEntity)
 - **Usage**
 - * Marks an instance (registry) of the entity as the selected.
 - **Parameters**
 - * idEntity - Long, internal id of the selected instance (registry).

- *selectSubentity*
public void selectSubentity(java.lang.String parentFieldId, com.imagos.axilice.core.common.BaseVO subentityOT)
 - **Usage**
 - * Updates an entity instance by setting up the selected value for a subentity field.
 - **Parameters**
 - * parentFieldId - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * subentityOT - BaseVO, subentity's selected instance (registry) value.

- *setEntity*
public void setEntity(com.imagos.axilice.core.installer.EntityOT entity)
 - **Usage**
 - * Sets up the entity name.
 - **Parameters**
 - * name - String, entity name.

- *setRegistries*
public void setRegistries(java.util.ArrayList lRegistries)
 - **Usage**
 - * Sets the list of registries that are showed in the entity view controlled by the Entity Launcher instance. It's used for detail entities where queries are made in the master entity launcher.
 - **Parameters**
 - * lRegistries - ArrayList<BaseVO>, list of shown registries of the entity.

- *showSelectionDialog*
public void showSelectionDialog(com.imagos.axilice.core.installer.SubentityOT entData)
 - **Usage**
 - * Opens the selection dialog that enables the user to select an instance of a subentity.
 - **Parameters**
 - * entData - SubentityOT, data of the detail subentity to select.

- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT
entData, long selEntity, boolean readonly )
```

- **Usage**

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

- **Parameters**

- * **entData** - SubentityOT, data of the detail subentity to select.
 - * **selEntity** - long, id of the selected instance of the subentity.
 - * **readonly** - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.
-

- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT
entData, java.lang.String parentGuiName, long selEntity, boolean readonly )
```

- **Usage**

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

- **Parameters**

- * **entData** - SubentityOT, data of the detail subentity to select.
 - * **parentGuiName** - String,
 - * **selEntity** - long, id of the selected instance of the subentity.
 - * **readonly** - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.
-

- *showSubentitySelectionDialog*

```
public void showSubentitySelectionDialog(
com.imagos.axilice.core.installer.SubentityOT entData )
```

- **Usage**

- * Opens the selection dialog that enables the user to select an instance of a multiple selection detail subentity.

- **Parameters**

- * **entData** - SubentityOT, data of the detail subentity to select.
-

- *undoEntityEdition*

```
public void undoEntityEdition( )
```

- *updateEntityData*

```
public abstract void updateEntityData( )
```

- **Usage**

- * Updates the selected instance (registry) data with the content panel's data typed by the user.
-

- *viewEntity*

```
public abstract OperationResult viewEntity( )
```

- **Usage**

- * View entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), in read-only mode.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

DECLARATION

```
public class PaymentMethodClient
extends com.imagos.axilice.client.gui.common.CommonClient
```

CONSTRUCTORS

- *PaymentMethodClient*
public **PaymentMethodClient**()

METHODS

- *build_extension_fields_view*
public void **build_extension_fields_view**()
- *buildSelectionSets*
public void **buildSelectionSets**()
- *fillEntityData*
public void **fillEntityData**()
- *getEntityData*
public BaseVO **getEntityData**()
- *getWidgets*
public void **getWidgets**()
- *init_specific_components*
public void **init_specific_components**()
- *setEditionMode*
public void **setEditionMode**(boolean enabled)
- *setWidgetsValues*
public void **setWidgetsValues**(com.imagos.axilice.core.common.BaseVO entity)
- *validateEntity*
public **OperationResult** **validateEntity**()

METHODS INHERITED FROM CLASS com.imagos.axilice.client.gui.common.CommonClient

(in 17.1.2, page 402)

- *addAxiliceEventListener*
public void **addAxiliceEventListener**(
com.imagos.axilice.client.gui.AxiliceEventListener l)
- *addEntity*
public **ArrayList** **addEntity**()
- *addExistingSubentity*
public void **addExistingSubentity**(com.imagos.axilice.core.installer.SubentityOT ent)

- **Usage**
 - * Shows a dialog for selection of a new subentity
 - **Parameters**
 - * **ent** - SubentityOT. Encapsulate data type of subentity
-
- *addSubentity*
public void addSubentity(com.imagos.axilice.core.common.BaseVO subentity, com.imagos.axilice.core.common.MultiSelectionSubentityOT subE)
 - **Usage**
 - * Adds a existent entity to a multiselection list
 - **Parameters**
 - * **subentity** - Subentity to be added
 - * **subE** - Object that encapsulates information about the subentity
-
- *addTrazabilityTree*
public void addTrazabilityTree(TabFolder tabFolder)
 - *build_extension_fields_view*
public abstract void build_extension_fields_view()
 - **Usage**
 - * Build the view of the extension fields.

It must implement only those clients who require it
-
- *buildEntity*
public AxiliceEntityView buildEntity()
 - *buildEntityDetail*
public AxiliceEntityView buildEntityDetail(com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData)
 - *buildEntitySelectionView*
public AxiliceEntitySelectionView buildEntitySelectionView()
 - **Usage**
 - * Build view of selection entities
 - **Returns** - AxilicePartnerSelectionView. SelectionView created
-
- *buildSelectionSets*
public abstract void buildSelectionSets()
 - **Usage**
 - * Build 2 sets of SubentityOT: lSelectionSubentities and lMultiSelectionSubentities

These sets are used to connect the widgets with the handlers for the selection buttons and the multiseleccion panels

It must implement only those clients who require it
-
- *buildSubentityMultiSelection*
public void buildSubentityMultiSelection()
 - *clearEntityData*
public void clearEntityData()
 - *clearParentField*
public OperationResult clearParentField(java.util.ArrayList selRegistries, java.lang.String parentField)
 - **Usage**
 - * Clears the value of the field indicated

- **Parameters**
 - * selRegistries - ArrayList. Index list of records to process
 - * parentField - Name of the field to clear
 - **Returns** - OperationResult. Result of operation
-
- *delEntity*
public OperationResult delEntity()
 - *delEntity*
public OperationResult delEntity(java.util.ArrayList selRegistries)
 - *delEntity*
public OperationResult delEntity(com.imagos.axilice.core.common.BaseVO b)
- **Usage**
 - * Delete a entity from database
 - **Parameters**
 - * b - BaseVO. Entity to delete
 - **Returns** - OperationResult. Result of the operation
-
- *delSubentities*
public OperationResult delSubentities(java.util.ArrayList selRegistries)
- **Usage**
 - * Delete a set of subentities from record list of controller, not from database
 - Set the parent field of subentity to null
- **Parameters**
 - * selRegistries - List of index to delete
 - **Returns** - OperationResult. Result of the operation
-
- *editEntity*
public OperationResult editEntity()
 - *fillEntityData*
public void fillEntityData()
 - *fillEntityExtensionTable*
public void fillEntityExtensionTable(java.lang.Long id)
- **Usage**
 - * Retrieve extension field values from the server for a given entity and displays them in the view
 - **Parameters**
 - * id - Long. Key of the entity whose values are to be displayed
-
- *fillEntityTable*
public void fillEntityTable()
- **Usage**
 - * Load records from database and fill the entity view table and controller record list
-
- *fillEntityTable*
public void fillEntityTable(java.util.ArrayList lR)
- **Usage**
 - * Fill the entity view table with the records given
 - **Parameters**
 - * lR - ArrayList. Records to be displayed
-

- *getClassEntity*
public Class getClassEntity()
 – **Returns** - Class of the entity for which the controller is built

- *getDialogName*
public String getDialogName()
- *getEditGladePath*
public String getEditGladePath()
- *getEditionGui*
public Gui getEditionGui()
 – **Usage**
 * Returns the edition Gui instance.
 – **Returns** - Gui, entity edition Gui.

- *getEditWindow*
public SwtTopLevelWindow getEditWindow()
- *getEntityData*
public abstract BaseVO getEntityData()
- *getEntityFields*
public ArrayList getEntityFields()
- *getLogger*
public Logger getLogger()
- *getParentEntityLauncher*
public EntityLauncher getParentEntityLauncher()
 – **Returns** - EntityLauncher of parent entity if is a embedded entity or null otherwise

- *getRegistries*
public Collection getRegistries()
 – **Returns** - IRegistries

- *getRegistries*
public List getRegistries(java.util.ArrayList lIds)
 – **Usage**
 * Get a list of entities stored by the controller
 – **Parameters**
 * lIds - Index list
 – **Returns** - List of entities

- *getView*
public AxiliceEntityView getView()
 – **Returns** - AxiliceEntityView. Main view or Detail view if is a embedded entity.

- *getWidgets*
public abstract void getWidgets()
 – **Usage**
 * Get the widgets from the glade and stored in local variables.

 Must be overwritten

- *init_specific_components*
public abstract void init_specific_components()
 – **Usage**

* Initializes the specific components of the client

- *printEntity*
public OperationResult printEntity()
- *printEntity*
public OperationResult printEntity(java.lang.String reportPath, java.util.HashMap parameters)
- *queryEntity*
public ArrayList queryEntity()
- *queryEntity*
public ArrayList queryEntity(java.lang.String fieldId, java.lang.String value)
- *queryEntityWithFormat*
public ArrayList queryEntityWithFormat(java.lang.String fieldId, java.lang.String value)
- *queryFreeEntities*
public ArrayList queryFreeEntities(com.imagos.axilice.core.common.MultiSelectionSubentityOT mSubE)
 - Usage
 - * Search entities not associated with a parent entity (null parent field)
 - Parameters
 - * mSubE - SubentityOT for the parent entity that must be null
 - Returns - ArrayList. List of entities found
- *querySubentities*
public ArrayList querySubentities(com.imagos.axilice.core.installer.SubentityOT entData)
- *refreshEntity*
public OperationResult refreshEntity()
- *refreshEntity*
public BaseVO refreshEntity(com.imagos.axilice.core.common.BaseVO entity)
- *refreshParentSelectedEntity*
public OperationResult refreshParentSelectedEntity()
 - Usage
 - * Reload from database the value of the parent entity of the selected entity
 - Returns - OperationResult. The result of the operation.
- *removeRegistries*
public OperationResult removeRegistries(java.util.ArrayList selRegistries, java.lang.String parentField)
 - Usage
 - * Deletes a list of the records stored by the controller. Also clean your parent fields and updates the database
 - Parameters
 - * selRegistries - Index list to remove
 - * parentField - Field name that links to the parent entity
 - Returns - OperationResult. Result of the operation
- *saveEntity*
public OperationResult saveEntity()
- *saveEntity*
public OperationResult saveEntity(java.util.ArrayList lIds)
- *saveEntity*
public OperationResult saveEntity(com.imagos.axilice.core.common.BaseVO b)

- **Usage**
 - * Save or update a entity to database
 - **Parameters**
 - * **b** - BaseVO. Entity to save or update
 - **Returns** - OperationResult. Result of the operation
-
- *selectEntity*
public void selectEntity(java.lang.Long idEntity)
 - *selectSubentity*
public void selectSubentity(java.lang.String parentFieldId, com.imagos.axilice.core.common.BaseVO subentityOT)
 - *selectTreeItem*
public void selectTreeItem(java.lang.Long id)
- **Usage**
 - * Select a item in the entity tree. No me gusta esto aqui
 - **Parameters**
 - * **id** - Long. Index of item to select
-
- *setDialogName*
public void setDialogName(java.lang.String dialogName)
 - *setEditGladePath*
public void setEditGladePath(java.lang.String editGladePath)
 - *setEditionMode*
public abstract void setEditionMode(boolean enabled)
 - *setRegistries*
public void setRegistries(java.util.Collection collection)
- **Usage**
 - * Setter for IRegistries
-
- *setWidgetsValues*
public abstract void setWidgetsValues(com.imagos.axilice.core.common.BaseVO entity)
- **Usage**
 - * Set the values of the widgets on the edit dialog.
 - **Parameters**
 - * **entity** - BaseVO. Entity that contains the values to be displayed
-
- *showErrorMessages*
public void showErrorMessages(com.imagos.axilice.core.util.OperationResult result)
-
- *updateEntityData*
public void updateEntityData()
 - *updateRegistry*
public void updateRegistry(com.imagos.axilice.core.common.BaseVO ent)
- **Usage**
 - * Add or update an entity (to the controller records)
 - **Parameters**
 - * **ent** - Entity to add or update
-
- *updateViews*
public void updateViews(boolean updateFromParent)

- **Usage**
 - * Force the update of the views associated
 - **Parameters**
 - * **updateFromParent** - Indicates whether the view must be updated by reading the values from the parent entity. takes effect only when the entity is detail (is embedded in another)
-
- *validateEntity*
 public abstract **OperationResult validateEntity()**
 - **Usage**
 - * Validate the values of the widgets in the edit dialog.
 - **Returns** - **OperationResult**. Result of the validation.
-
- *viewEntity*
 public **OperationResult viewEntity()**

METHODS INHERITED FROM CLASS com.imagos.axilice.core.installer.EntityLauncher

(in 24.2.2, page 532)

- *addEntity*
 public abstract **ArrayList addEntity()**
 - **Usage**
 - * Add entity button clicked handler. Show the entity content screen ready to start inserting data for a new instance of the entity (new registry).
 - **Returns** - **ArrayList<ResultMessage>**, list of errors if any has taken place, null in any other case.
-
- *buildEditEntityGUI*
 public void **buildEditEntityGUI()**
 - **Usage**
 - * Builds the edit entity dialog window from the corresponding glade interface file. Stores the entity dialog instance for posterior accesses.
-
- *buildEntity*
 public **AxiliceEntityView buildEntity()**
 - **Usage**
 - * Builds the entity GUI and show it on screen to the user.
-
- *buildEntity*
 public **AxiliceDetailEntityView buildEntity(SwtWidget parent, com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData, java.lang.String rootName)**
 - **Usage**
 - * Builds the entity GUI and show it on screen to the user. The entity GUI will be embedded into the parent composite widget.
 - **Parameters**
 - * **parent** - Composite, parent holder for embedding the entity view.
 - * **subentityData** - SubentityOT, data of the subentity to build if the entity launcher will be used to handle a multi-selection detail entity view.
 - * **rootName** - String, name of the root widget that will be embedded into the master entity edition panel.

- **Returns** - AxiliceDetailView, the entity detail view built.

- *cancelEntityEdition*
public void cancelEntityEdition()
- *clearEntityData*
public abstract void clearEntityData()
 - **Usage**
 - * Clears the entity content panel by resetting the content widgets.

- *delEntity*
public abstract OperationResult delEntity()
 - **Usage**
 - * Delete entity button clicked handler. Deletes the selected instance of the entity (selected registry) from the database.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *delEntity*
public abstract OperationResult delEntity(java.util.ArrayList lSelRegistries)
 - **Usage**
 - * Delete entity button clicked handler. Deletes the selected instances of the entity (selected registries) from the database. Its a multiple delete method.
 - **Parameters**
 - * lSelRegistries - ArrayList<Long>, list of selected instances of the entity (selected registries). The list will contain only the internal id of the selected registries.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *deselectSubentity*
public void deselectSubentity(java.lang.String parentFieldId, com.imagos.axilice.core.common.BaseVO subentityOT)
 - **Usage**
 - * Updates an entity instance by removing the selected value for a subentity field.
 - **Parameters**
 - * parentFieldId - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * subentityOT - BaseVO, subentity's deselected instance (registry) value.

- *editEntity*
public abstract OperationResult editEntity()
 - **Usage**
 - * Edit entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), and allows the user to modify any content of the entity registry.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *fillEntityData*
public abstract void fillEntityData()
 - **Usage**
 - * Fills the entity content panel with the data of the selected instance (registry) of the entity.

- *fireExceptionAndExit*
public void fireExceptionAndExit(java.lang.String msg)

- **Usage**
 - * Fire a exception and stop the program
 - **Parameters**
 - * msg -
- formatRegistries
public ArrayList **formatRegistries**()
- formatRegistries
public ArrayList **formatRegistries**(java.util.ArrayList lRegistries)
- getDialog
public SwtDialog **getDialog**(java.lang.String dialogName)
- getEntity
public EntityOT **getEntity**()
 - **Usage**
 - * Gets the entity data.
 - **Returns** - EntityOT, entity data.
- getEntityData
public abstract BaseVO **getEntityData**()
 - **Usage**
 - * Gets the content panel's data typed by the user formatted as an entity VO object.
- getEntityEditViewName
public String **getEntityEditViewName**(java.lang.String parentGuiName)
- getEntityFields
public abstract ArrayList **getEntityFields**()
 - **Usage**
 - * Gets the list of defined fields for an entity
 - **Returns** - ArrayList, list of entity fields.
- getEntityViewName
public String **getEntityViewName**(java.lang.String parentGuiName)
- getLRegistries
public ArrayList **getLRegistries**()
 - **Usage**
 - * Getter for lRegistries
- getRegistry
public BaseVO **getRegistry**(long id)
- getRevision
public static String **getRevision**()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
- getSelectedEntity
public BaseVO **getSelectedEntity**()
 - **Usage**
 - * Gets the selected instance (registry) of the entity.
 - **Returns** - BaseVO, selected instance (registry) of the entity.
- getSelectedId
public Long **getSelectedId**()

- **Usage**
 - * Gets the internal id of the selected entity registry.
 - **Returns** - Long, internal id of the selected entity registry, null in case no entity registry has been selected yet.

- *getSwtWidget*
public SwtWidget getSwtWidget(java.lang.String widgetName)
- *getWidget*
public Widget getWidget(java.lang.String widgetName)
- *initEntityLauncher*
public void initEntityLauncher(Logger logger, Gui parentGui, SwtWidget parentPanel)
 - **Usage**
 - * Inits the entity launcher context and prepares the launcher to be started.
 - **Parameters**
 - * **logger** - Logger, logger instance for printing logging messages.
 - * **parentGuiName** - String, name of the JavaGlade instance that built the parent where the entity will be embedded.
 - * **parentPanel** - Composite, parent panel where the entity contents will be embedded.

- *isDetail*
public boolean isDetail()
 - **Usage**
 - * Indicates whether is a main entity or is embedded in another

- *on_entitySelectionButton_clicked*
public void on_entitySelectionButton_clicked(Event e)
 - **Usage**
 - * Handler for the subentity selection buttons (magnifying glass buttons) clicked. Shows the selection dialog for selecting a valid instance of a subentity.
 - **Parameters**
 - * **e** - Event, magnifying glass button clicked event data.

- *printEntity*
public abstract OperationResult printEntity()
 - **Usage**
 - * Print entity button clicked handler. Builds the report with the data of the selected instance of the entity (selected registry), and allows the user to print the elaborated report.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *printWidgets*
public void printWidgets()
- *queryEntity*
public abstract ArrayList queryEntity()
 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) available.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.

- *queryEntity*
public abstract ArrayList queryEntity(java.lang.String fieldId, java.lang.String value)

- **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria.
 - **Parameters**
 - * **fieldId** - String, name of the entity field that will be queried.
 - * **value** - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.
-
- *queryEntityWithFormat*
 public abstract ArrayList **queryEntityWithFormat**(java.lang.String **fieldId**,
 java.lang.String **value**)
 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria and show them (formatted as a list) to the user in the entity's screen.
 - **Parameters**
 - * **fieldId** - String, name of the entity field that will be queried.
 - * **value** - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList<RegistryOT>, list of entity instances (registries) taken as the result of the query operation.
-
- *querySubentities*
 public abstract ArrayList **querySubentities**(
 com.imagos.axilice.core.installer.SubentityOT **entData**)
 - **Usage**
 - * Queries the selected entity for all the detail subentities associated with it.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to query for.
 - **Returns** - ArrayList, list of detail subentity registries assigned to the master detail registry.
-
- *saveEntity*
 public abstract OperationResult **saveEntity**()
 - **Usage**
 - * Save entity button clicked handler. Update the system with the data of the edited entity registry. It's called for saving new entity registries (add a new registry) or for updating the contents of existing entity registries.
 - **Returns** - OperationResult, Result of the operation and list of messages generated through the operation.
-
- *saveEntityRegistry*
 public void **saveEntityRegistry**()
-
- *selectEntity*
 public abstract void **selectEntity**(java.lang.Long **idEntity**)
 - **Usage**
 - * Marks an instance (registry) of the entity as the selected.
 - **Parameters**
 - * **idEntity** - Long, internal id of the selected instance (registry).
-
- *selectSubentity*
 public void **selectSubentity**(java.lang.String **parentFieldId**,
 com.imagos.axilice.core.common.BaseV0 **subentityOT**)
 - **Usage**

- * Updates an entity instance by setting up the selected value for a subentity field.
 - **Parameters**
 - * **parentFieldId** - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * **subentityOT** - BaseVO, subentity's selected instance (registry) value.
-
- *setEntity*

```
public void setEntity( com.imagos.axilice.core.installer.EntityOT entity )
```

 - **Usage**
 - * Sets up the entity name.
 - **Parameters**
 - * **name** - String, entity name.
-
- *setRegistries*

```
public void setRegistries( java.util.ArrayList lRegistries )
```

 - **Usage**
 - * Sets the list of registries that are showed in the entity view controlled by the Entity Launcher instance. It's used for detail entities where queries are made in the master entity launcher.
 - **Parameters**
 - * **lRegistries** - ArrayList<BaseVO>, list of shown registries of the entity.
-
- *showSelectionDialog*

```
public void showSelectionDialog( com.imagos.axilice.core.installer.SubentityOT entData )
```

 - **Usage**
 - * Opens the selection dialog that enables the user to select an instance of a subentity.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to select.
-
- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT entData, long selEntity, boolean readonly )
```

 - **Usage**
 - * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to select.
 - * **selEntity** - long, id of the selected instance of the subentity.
 - * **readonly** - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.
-
- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT entData, java.lang.String parentGuiName, long selEntity, boolean readonly )
```

 - **Usage**
 - * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to select.
 - * **parentGuiName** - String,
 - * **selEntity** - long, id of the selected instance of the subentity.
 - * **readonly** - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

-
- *showSubentitySelectionDialog*
 public void **showSubentitySelectionDialog**(
 com.imagos.axilice.core.installer.SubentityOT **entData**)
 – **Usage**
 * Opens the selection dialog that enables the user to select an instance of a multiple selection detail subentity.
 – **Parameters**
 * **entData** - SubentityOT, data of the detail subentity to select.

 - *undoEntityEdition*
 public void **undoEntityEdition**()

 - *updateEntityData*
 public abstract void **updateEntityData**()
 – **Usage**
 * Updates the selected instance (registry) data with the content panel's data typed by the user.

 - *viewEntity*
 public abstract **OperationResult** **viewEntity**()
 – **Usage**
 * View entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), in read-only mode.
 – **Returns** - **OperationResult**, result of the delete operation and list of messages generated during the operation.

27.1.10 CLASS *RappelIntervalClient*

DECLARATION

```
public class RappelIntervalClient
extends com.imagos.axilice.client.gui.common.CommonClient
```

CONSTRUCTORS

- *RappelIntervalClient*
 public **RappelIntervalClient**()

METHODS

- *build_extension_fields_view*
 public void **build_extension_fields_view**()

- *buildSelectionSets*
 public void **buildSelectionSets**()

- *clearEntityData*
public void clearEntityData()
- *fillEntityData*
public void fillEntityData()
- *getEntityData*
public BaseVO getEntityData()
- *getWidgets*
public void getWidgets()
- *init_specific_components*
public void init_specific_components()
- *setEditionMode*
public void setEditionMode(boolean enabled)
- *setWidgetsValues*
public void setWidgetsValues(com.imagos.axilice.core.common.BaseVO entity)
- *validateEntity*
public OperationResult validateEntity()

METHODS INHERITED FROM CLASS com.imagos.axilice.client.gui.common.CommonClient

(in 17.1.2, page 402)

- *addAxiliceEventListener*
public void addAxiliceEventListener(com.imagos.axilice.client.gui.AxiliceEventListener l)
- *addEntity*
public ArrayList addEntity()
- *addExistingSubentity*
public void addExistingSubentity(com.imagos.axilice.core.installer.SubentityOT ent)
 - **Usage**
 - * Shows a dialog for selection of a new subentity
 - **Parameters**
 - * ent - SubentityOT. Encapsulate data type of subentity
- *addSubentity*
public void addSubentity(com.imagos.axilice.core.common.BaseVO subentity, com.imagos.axilice.core.common.MultiSelectionSubentityOT subE)
 - **Usage**
 - * Adds a existent entity to a multiselection list
 - **Parameters**
 - * subentity - Subentity to be added
 - * subE - Object that encapsulates information about the subentity
- *addTrazabilityTree*
public void addTrazabilityTree(TabFolder tabFolder)
- *build_extension_fields_view*
public abstract void build_extension_fields_view()

– **Usage**

- * Build the view of the extension fields.

It must implement only those clients who require it

- *buildEntity*
public AxiliceEntityView buildEntity()
- *buildEntityDetail*
public AxiliceEntityView buildEntityDetail(
com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData)
- *buildEntitySelectionView*
public AxiliceEntitySelectionView buildEntitySelectionView()

– **Usage**

- * Build view of selection entities

– **Returns** - AxilicePartnerSelectionView. SelectionView created

- *buildSelectionSets*
public abstract void buildSelectionSets()

– **Usage**

- * Build 2 sets of SubentityOT: lSelectionSubentities and lMultiSelectionSubentities

These sets are used to connect the widgets with the handlers for the selection buttons and the multiselection panels

It must implement only those clients who require it

- *buildSubentityMultiSelection*
public void buildSubentityMultiSelection()
- *clearEntityData*
public void clearEntityData()
- *clearParentField*
public OperationResult clearParentField(java.util.ArrayList selRegistries,
java.lang.String parentField)

– **Usage**

- * Clears the value of the field indicated

– **Parameters**

- * selRegistries - ArrayList. Index list of records to process
- * parentField - Name of the field to clear

– **Returns** - OperationResult. Result of operation

- *delEntity*
public OperationResult delEntity()
- *delEntity*
public OperationResult delEntity(java.util.ArrayList selRegistries)
- *delEntity*
public OperationResult delEntity(com.imagos.axilice.core.common.BaseVO b)

– **Usage**

- * Delete a entity from database

– **Parameters**

- * b - BaseVO. Entity to delete

– **Returns** - OperationResult. Result of the operation

- *delSubentities*
public OperationResult delSubentities(java.util.ArrayList selRegistries)
 - **Usage**
 - * Delete a set of subentities from record list of controller, not from database
 - Set the parent field of subentity to null
 - **Parameters**
 - * selRegistries - List of index to delete
 - **Returns** - OperationResult. Result of the operation

- *editEntity*
public OperationResult editEntity()
- *fillEntityData*
public void fillEntityData()
- *fillEntityExtensionTable*
public void fillEntityExtensionTable(java.lang.Long id)
 - **Usage**
 - * Retrieve extension field values from the server for a given entity and displays them in the view
 - **Parameters**
 - * id - Long. Key of the entity whose values are to be displayed

- *fillEntityTable*
public void fillEntityTable()
 - **Usage**
 - * Load records from database and fill the entity view table and controller record list

- *fillEntityTable*
public void fillEntityTable(java.util.ArrayList lR)
 - **Usage**
 - * Fill the entity view table with the records given
 - **Parameters**
 - * lR - ArrayList. Records to be displayed

- *getClassEntity*
public Class getClassEntity()
 - **Returns** - Class of the entity for which the controller is built

- *getDialogName*
public String getDialogName()
- *getEditGladePath*
public String getEditGladePath()
- *getEditionGui*
public Gui getEditionGui()
 - **Usage**
 - * Returns the edition Gui instance.
 - **Returns** - Gui, entity edition Gui.

- *getEditWindow*
public SwtTopLevelWindow getEditWindow()

- *getEntityData*
public abstract BaseVO **getEntityData**()
- *getEntityFields*
public ArrayList **getEntityFields**()
- *getLogger*
public Logger **getLogger**()
- *getParentEntityLauncher*
public EntityLauncher **getParentEntityLauncher**()
 - **Returns** - EntityLauncher of parent entity if is a embedded entity or null otherwise
- *getRegistries*
public Collection **getRegistries**()
 - **Returns** - IRegistries
- *getRegistries*
public List **getRegistries**(java.util.ArrayList lIds)
 - **Usage**
 - * Get a list of entities stored by the controller
 - **Parameters**
 - * lIds - Index list
 - **Returns** - List of entities
- *getView*
public AxiliceEntityView **getView**()
 - **Returns** - AxiliceEntityView. Main view or Detail view if is a embedded entity.
- *getWidgets*
public abstract void **getWidgets**()
 - **Usage**
 - * Get the widgets from the glade and stored in local variables.
 - Must be overwritten
- *init_specific_components*
public abstract void **init_specific_components**()
 - **Usage**
 - * Initializes the specific components of the client
- *printEntity*
public OperationResult **printEntity**()
- *printEntity*
public OperationResult **printEntity**(java.lang.String reportPath, java.util.HashMap parameters)
- *queryEntity*
public ArrayList **queryEntity**()
- *queryEntity*
public ArrayList **queryEntity**(java.lang.String fieldId, java.lang.String value)
- *queryEntityWithFormat*
public ArrayList **queryEntityWithFormat**(java.lang.String fieldId, java.lang.String value)
- *queryFreeEntities*
public ArrayList **queryFreeEntities**(com.imagos.axilice.core.common.MultiSelectionSubentityOT mSubE)

- **Usage**
 - * Search entities not associated with a parent entity (null parent field)
 - **Parameters**
 - * mSubE - SubentityOT for the parent entity that must be null
 - **Returns** - ArrayList. List of entities found
-
- *querySubentities*
 public ArrayList **querySubentities**(com.imagos.axilice.core.installer.SubentityOT
 entData)
-
- *refreshEntity*
 public OperationResult **refreshEntity**()
-
- *refreshEntity*
 public BaseVO **refreshEntity**(com.imagos.axilice.core.common.BaseVO entity)
-
- *refreshParentSelectedEntity*
 public OperationResult **refreshParentSelectedEntity**()
- **Usage**
 - * Reload from database the value of the parent entity of the selected entity
 - **Returns** - OperationResult. The result of the operation.
-
- *removeRegistries*
 public OperationResult **removeRegistries**(java.util.ArrayList selRegistries,
 java.lang.String parentField)
- **Usage**
 - * Deletes a list of the records stored by the controller. Also clean your parent fields and updates the database
 - **Parameters**
 - * selRegistries - Index list to remove
 - * parentField - Field name that links to the parent entity
 - **Returns** - OperationResult. Result of the operation
-
- *saveEntity*
 public OperationResult **saveEntity**()
-
- *saveEntity*
 public OperationResult **saveEntity**(java.util.ArrayList IIds)
-
- *saveEntity*
 public OperationResult **saveEntity**(com.imagos.axilice.core.common.BaseVO b)
- **Usage**
 - * Save or update a entity to database
 - **Parameters**
 - * b - BaseVO. Entity to save or update
 - **Returns** - OperationResult. Result of the operation
-
- *selectEntity*
 public void **selectEntity**(java.lang.Long idEntity)
-
- *selectSubentity*
 public void **selectSubentity**(java.lang.String parentFieldId,
 com.imagos.axilice.core.common.BaseVO subentityOT)
-
- *selectTreeItem*
 public void **selectTreeItem**(java.lang.Long id)
- **Usage**
 - * Select a item in the entity tree. No me gusta esto aqui
 - **Parameters**

- * *id* - Long. Index of item to select
- - *setDialogName*
public void setDialogName(java.lang.String *dialogName*)
 - *setEditGladePath*
public void setEditGladePath(java.lang.String *editGladePath*)
 - *setEditionMode*
public abstract void setEditionMode(boolean *enabled*)
 - *setRegistries*
public void setRegistries(java.util.Collection *collection*)
 - **Usage**
 - * Setter for IRegistries

 - *setWidgetsValues*
public abstract void setWidgetsValues(com.imagos.axilice.core.common.BaseVO *entity*)
 - **Usage**
 - * Set the values of the widgets on the edit dialog.
 - Must be overwritten.
 - **Parameters**
 - * *entity* - BaseVO. Entity that contains the values to be displayed

 - *showErrorMessages*
public void showErrorMessages(com.imagos.axilice.core.util.OperationResult *result*)

 - *updateEntityData*
public void updateEntityData()
 - *updateRegistry*
public void updateRegistry(com.imagos.axilice.core.common.BaseVO *ent*)
 - **Usage**
 - * Add or update an entity (to the controller records)
 - **Parameters**
 - * *ent* - Entity to add or update

 - *updateViews*
public void updateViews(boolean *updateFromParent*)
 - **Usage**
 - * Force the update of the views associated
 - **Parameters**
 - * *updateFromParent* - Indicates whether the view must be updated by reading the values from the parent entity. takes effect only when the entity is detail (is embedded in another)

 - *validateEntity*
public abstract OperationResult validateEntity()
 - **Usage**
 - * Validate the values of the widgets in the edit dialog.
 - Must be overwritten.
 - **Returns** - OperationResult. Result of the validation.

 - *viewEntity*
public OperationResult viewEntity()

METHODS INHERITED FROM CLASS `com.imagos.axilice.core.installer.EntityLauncher`

(in 24.2.2, page 532)

- *addEntity*
`public abstract ArrayList addEntity()`
 - **Usage**
 - * Add entity button clicked handler. Show the entity content screen ready to start inserting data for a new instance of the entity (new registry).
 - **Returns** - `ArrayList<ResultMessage>`, list of errors if any has taken place, null in any other case.

 - *buildEditEntityGUI*
`public void buildEditEntityGUI()`
 - **Usage**
 - * Builds the edit entity dialog window from the corresponding glade interface file. Stores the entity dialog instance for posterior accesses.

 - *buildEntity*
`public AxiliceEntityView buildEntity()`
 - **Usage**
 - * Builds the entity GUI and show it on screen to the user.

 - *buildEntity*
`public AxiliceDetailEntityView buildEntity(SwtWidget parent,
com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData,
java.lang.String rootName)`
 - **Usage**
 - * Builds the entity GUI and show it on screen to the user. The entity GUI will be embedded into the parent composite widget.
 - **Parameters**
 - * **parent** - Composite, parent holder for embedding the entity view.
 - * **subentityData** - SubentityOT, data of the subentity to build if the entity launcher will be used to handle a multi-selection detail entity view.
 - * **rootName** - String, name of the root widget that will be embedded into the master entity edition panel.
 - **Returns** - `AxiliceDetailEntityView`, the entity detail view built.

 - *cancelEntityEdition*
`public void cancelEntityEdition()`

 - *clearEntityData*
`public abstract void clearEntityData()`
 - **Usage**
 - * Clears the entity content panel by resetting the content widgets.

 - *delEntity*
`public abstract OperationResult delEntity()`
 - **Usage**
 - * Delete entity button clicked handler. Deletes the selected instance of the entity (selected registry) from the database.
 - **Returns** - `OperationResult`, result of the delete operation and list of messages generated during the operation.
-

- *delEntity*

```
public abstract OperationResult delEntity( java.util.ArrayList lSelRegistries )
```

- **Usage**

- * Delete entity button clicked handler. Deletes the selected instances of the entity (selected registries) from the database. Its a multiple delete method.

- **Parameters**

- * lSelRegistries - ArrayList<Long>, list of selected instances of the entity (selected registries). The list will contain only the internal id of the selected registries.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *deselectSubentity*

```
public void deselectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

- **Usage**

- * Updates an entity instance by removing the selected value for a subentity field.

- **Parameters**

- * parentFieldId - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.

- * subentityOT - BaseVO, subentity's deselected instance (registry) value.

- *editEntity*

```
public abstract OperationResult editEntity( )
```

- **Usage**

- * Edit entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), and allows the user to modify any content of the entity registry.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *fillEntityData*

```
public abstract void fillEntityData( )
```

- **Usage**

- * Fills the entity content panel with the data of the selected instance (registry) of the entity.

- *fireExceptionAndExit*

```
public void fireExceptionAndExit( java.lang.String msg )
```

- **Usage**

- * Fire a exception and stop the program

- **Parameters**

- * msg -

- *formatRegistries*

```
public ArrayList formatRegistries( )
```

- *formatRegistries*

```
public ArrayList formatRegistries( java.util.ArrayList lRegistries )
```

- *getDialog*

```
public SwtDialog getDialog( java.lang.String dialogName )
```

- *getEntity*

```
public EntityOT getEntity( )
```

- **Usage**

- * Gets the entity data.

- **Returns** - EntityOT, entity data.
- getEntityData
 public abstract BaseVO **getEntityData**()
 - **Usage**
 - * Gets the content panel's data typed by the user formatted as an entity VO object.
- getEntityEditViewName
 public String **getEntityEditViewName**(java.lang.String parentGuiName)
- getEntityFields
 public abstract ArrayList **getEntityFields**()
 - **Usage**
 - * Gets the list of defined fields for an entity
 - **Returns** - ArrayList, list of entity fields.
- getEntityViewName
 public String **getEntityViewName**(java.lang.String parentGuiName)
- getLRegistries
 public ArrayList **getLRegistries**()
 - **Usage**
 - * Getter for lRegistries
- getRegistry
 public BaseVO **getRegistry**(long id)
- getRevision
 public static String **getRevision**()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
- getSelectedEntity
 public BaseVO **getSelectedEntity**()
 - **Usage**
 - * Gets the selected instance (registry) of the entity.
 - **Returns** - BaseVO, selected instance (registry) of the entity.
- getSelectedId
 public Long **getSelectedId**()
 - **Usage**
 - * Gets the internal id of the selected entity registry.
 - **Returns** - Long, internal id of the selected entity registry, null in case no entity registry has been selected yet.
- getSwtWidget
 public SwtWidget **getSwtWidget**(java.lang.String widgetName)
- getWidget
 public Widget **getWidget**(java.lang.String widgetName)
- initEntityLauncher
 public void **initEntityLauncher**(Logger logger, Gui parentGui, SwtWidget parentPanel)
 - **Usage**
 - * Inits the entity launcher context and prepares the launcher to be started.
 - **Parameters**

- * **logger** - Logger, logger instance for printing logging messages.
- * **parentGuiName** - String, name of the JavaGlade instance that built the parent where the entity will be embedded.
- * **parentPanel** - Composite, parent panel where the entity contents will be embedded.

- *isDetail*

public boolean **isDetail**()

– **Usage**

- * Indicates whether is a main entity or is embedded in another

- *on_entitySelectionButton_clicked*

public void **on_entitySelectionButton_clicked**(Event e)

– **Usage**

- * Handler for the subentity selection buttons (magnifying glass buttons) clicked. Shows the selection dialog for selecting a valid instance of a subentity.

– **Parameters**

- * **e** - Event, magnifying glass button clicked event data.

- *printEntity*

public abstract **OperationResult printEntity**()

– **Usage**

- * Print entity button clicked handler. Builds the report with the data of the selected instance of the entity (selected registry), and allows the user to print the elaborated report.

– **Returns** - **OperationResult**, result of the delete operation and list of messages generated during the operation.

- *printWidgets*

public void **printWidgets**()

- *queryEntity*

public abstract **ArrayList queryEntity**()

– **Usage**

- * Search entity button clicked handler. Searches the system for the entity instances (registries) available.

– **Returns** - **ArrayList**, list of entity instances (registries) taken as the result of the query operation.

- *queryEntity*

public abstract **ArrayList queryEntity**(java.lang.String fieldId, java.lang.String value)

– **Usage**

- * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria.

– **Parameters**

- * **fieldId** - String, name of the entity field that will be queried.
- * **value** - String, value of the entity field that will be searched for.

– **Returns** - **ArrayList**, list of entity instances (registries) taken as the result of the query operation.

- *queryEntityWithFormat*

public abstract **ArrayList queryEntityWithFormat**(java.lang.String fieldId, java.lang.String value)

– **Usage**

- * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria and show them (formatted as a list) to the user in the entity's screen.

– **Parameters**

- * **fieldId** - String, name of the entity field that will be queried.
- * **value** - String, value of the entity field that will be searched for.

- **Returns** - ArrayList<RegistryOT>, list of entity instances (registries) taken as the result of the query operation.

• *querySubentities*

```
public abstract ArrayList querySubentities(
com.imagos.axilice.core.installer.SubentityOT entData )
```

– **Usage**

- * Queries the selected entity for all the detail subentities associated with it.

– **Parameters**

- * **entData** - SubentityOT, data of the detail subentity to query for.

- **Returns** - ArrayList, list of detail subentity registries assigned to the master detail registry.

• *saveEntity*

```
public abstract OperationResult saveEntity( )
```

– **Usage**

- * Save entity button clicked handler. Update the system with the data of the edited entity registry. It's called for saving new entity registries (add a new registry) or for updating the contents of existing entity registries.

- **Returns** - OperationResult, Result of the operation and list of messages generated through the operation.

• *saveEntityRegistry*

```
public void saveEntityRegistry( )
```

• *selectEntity*

```
public abstract void selectEntity( java.lang.Long idEntity )
```

– **Usage**

- * Marks an instance (registry) of the entity as the selected.

– **Parameters**

- * **idEntity** - Long, internal id of the selected instance (registry).

• *selectSubentity*

```
public void selectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

– **Usage**

- * Updates an entity instance by setting up the selected value for a subentity field.

– **Parameters**

- * **parentFieldId** - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
- * **subentityOT** - BaseVO, subentity's selected instance (registry) value.

• *setEntity*

```
public void setEntity( com.imagos.axilice.core.installer.EntityOT entity )
```

– **Usage**

- * Sets up the entity name.

– **Parameters**

- * **name** - String, entity name.
-

- *setRegistries*

```
public void setRegistries( java.util.ArrayList lRegistries )
```

- Usage

- * Sets the list of registries that are showed in the entity view controlled by the Entity Launcher instance. It's used for detail entities where queries are made in the master entity launcher.

- Parameters

- * lRegistries - ArrayList<BaseVO>, list of shown registries of the entity.

- *showSelectionDialog*

```
public void showSelectionDialog( com.imagos.axilice.core.installer.SubentityOT entData )
```

- Usage

- * Opens the selection dialog that enables the user to select an instance of a subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.

- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT entData, long selEntity, boolean readonly )
```

- Usage

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.
 - * selEntity - long, id of the selected instance of the subentity.
 - * readonly - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT entData, java.lang.String parentGuiName, long selEntity, boolean readonly )
```

- Usage

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.
 - * parentGuiName - String,
 - * selEntity - long, id of the selected instance of the subentity.
 - * readonly - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

- *showSubentitySelectionDialog*

```
public void showSubentitySelectionDialog( com.imagos.axilice.core.installer.SubentityOT entData )
```

- Usage

- * Opens the selection dialog that enables the user to select an instance of a multiple selection detail subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.

- *undoEntityEdition*

```
public void undoEntityEdition( )
```

- *updateEntityData*
 public abstract void **updateEntityData**()
 – **Usage**
 * Updates the selected instance (registry) data with the content panel's data typed by the user.

- *viewEntity*
 public abstract **OperationResult** **viewEntity**()
 – **Usage**
 * View entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), in read-only mode.
 – **Returns** - **OperationResult**, result of the delete operation and list of messages generated during the operation.

27.1.11 CLASS *RappelTypeClient*

DECLARATION

```
public class RappelTypeClient
extends com.imagos.axilice.client.gui.common.CommonClient
```

CONSTRUCTORS

- *RappelTypeClient*
 public **RappelTypeClient**()

METHODS

- *build_extension_fields_view*
 public void **build_extension_fields_view**()

- *buildSelectionSets*
 public void **buildSelectionSets**()

- *fillEntityData*
 public void **fillEntityData**()

- *getEntityData*
 public **BaseVO** **getEntityData**()

- *getWidgets*
 public void **getWidgets**()

- *init_specific_components*
 public void **init_specific_components**()

- *setEditionMode*
 public void **setEditionMode**(boolean enabled)

- *setWidgetsValues*
public void setWidgetsValues(com.imagos.axilice.core.common.BaseVO entity)
- *validateEntity*
public OperationResult validateEntity()

METHODS INHERITED FROM CLASS com.imagos.axilice.client.gui.common.CommonClient

(in 17.1.2, page 402)

- *addAxiliceEventListener*
public void addAxiliceEventListener(com.imagos.axilice.client.gui.AxiliceEventListener l)
 - *addEntity*
public ArrayList addEntity()
 - *addExistingSubentity*
public void addExistingSubentity(com.imagos.axilice.core.installer.SubentityOT ent)
 - **Usage**
 - * Shows a dialog for selection of a new subentity
 - **Parameters**
 - * ent - SubentityOT. Encapsulate data type of subentity
 - *addSubentity*
public void addSubentity(com.imagos.axilice.core.common.BaseVO subentity, com.imagos.axilice.core.common.MultiSelectionSubentityOT subE)
 - **Usage**
 - * Adds a existent entity to a multiselection list
 - **Parameters**
 - * subentity - Subentity to be added
 - * subE - Object that encapsulates information about the subentity
 - *addTrazabilityTree*
public void addTrazabilityTree(TabFolder tabFolder)
 - *build_extension_fields_view*
public abstract void build_extension_fields_view()
 - **Usage**
 - * Build the view of the extension fields.

It must implement only those clients who require it
 - *buildEntity*
public AxiliceEntityView buildEntity()
 - *buildEntityDetail*
public AxiliceEntityView buildEntityDetail(com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData)
 - *buildEntitySelectionView*
public AxiliceEntitySelectionView buildEntitySelectionView()
 - **Usage**
 - * Build view of selection entities
 - **Returns** - AxilicePartnerSelectionView. SelectionView created
-

- *buildSelectionSets*
 public abstract void **buildSelectionSets**()
 – **Usage**
 * Build 2 sets of SubentityOT: lSelectionSubentities and lMultiSelectionSubentities
 These sets are used to connect the widgets with the handlers for the selection buttons and the multiseleccion panels
 It must implement only those clients who require it

- *buildSubentityMultiSelection*
 public void **buildSubentityMultiSelection**()
- *clearEntityData*
 public void **clearEntityData**()
- *clearParentField*
 public OperationResult **clearParentField**(java.util.ArrayList selRegistries,
 java.lang.String parentField)
 – **Usage**
 * Clears the value of the field indicated
 – **Parameters**
 * selRegistries - ArrayList. Index list of records to process
 * parentField - Name of the field to clear
 – **Returns** - OperationResult. Result of operation

- *delEntity*
 public OperationResult **delEntity**()
- *delEntity*
 public OperationResult **delEntity**(java.util.ArrayList selRegistries)
- *delEntity*
 public OperationResult **delEntity**(com.imagos.axilice.core.common.BaseVO b)
 – **Usage**
 * Delete a entity from database
 – **Parameters**
 * b - BaseVO. Entity to delete
 – **Returns** - OperationResult. Result of the operation

- *delSubentities*
 public OperationResult **delSubentities**(java.util.ArrayList selRegistries)
 – **Usage**
 * Delete a set of subentities from record list of controller, not from database
 Set the parent field of subentity to null
 – **Parameters**
 * selRegistries - List of index to delete
 – **Returns** - OperationResult. Result of the operation

- *editEntity*
 public OperationResult **editEntity**()
- *fillEntityData*
 public void **fillEntityData**()
- *fillEntityExtensionTable*
 public void **fillEntityExtensionTable**(java.lang.Long id)

- **Usage**
 - * Retrieve extension field values from the server for a given entity and displays them in the view
- **Parameters**
 - * **id** - Long. Key of the entity whose values are to be displayed

- *fillEntityTable*
public void fillEntityTable()
 - **Usage**
 - * Load records from database and fill the entity view table and controller record list

- *fillEntityTable*
public void fillEntityTable(java.util.ArrayList lR)
 - **Usage**
 - * Fill the entity view table with the records given
 - **Parameters**
 - * **lR** - ArrayList. Records to be displayed

- *getClassEntity*
public Class getClassEntity()
 - **Returns** - Class of the entity for which the controller is built

- *getDialogName*
public String getDialogName()
- *getEditGladePath*
public String getEditGladePath()
- *getEditionGui*
public Gui getEditionGui()
 - **Usage**
 - * Returns the edition Gui instance.
 - **Returns** - Gui, entity edition Gui.

- *getEditWindow*
public SwtTopLevelWindow getEditWindow()
- *getEntityData*
public abstract BaseVO getEntityData()
- *getEntityFields*
public ArrayList getEntityFields()
- *getLogger*
public Logger getLogger()
- *getParentEntityLauncher*
public EntityLauncher getParentEntityLauncher()
 - **Returns** - EntityLauncher of parent entity if is a embedded entity or null otherwise

- *getRegistries*
public Collection getRegistries()
 - **Returns** - lRegistries

- *getRegistries*
public List getRegistries(java.util.ArrayList lIds)
 - **Usage**
 - * Get a list of entities stored by the controller

- **Parameters**
 - * lIds - Index list
 - **Returns** - List of entities

- *getView*
public AxiliceEntityView getView()
 - **Returns** - AxiliceEntityView. Main view or Detail view if is a embedded entity.

- *getWidgets*
public abstract void getWidgets()
 - **Usage**
 - * Get the widgets from the glade and stored in local variables.
 - Must be overwritten

- *init_specific_components*
public abstract void init_specific_components()
 - **Usage**
 - * Initializes the specific components of the client

- *printEntity*
public OperationResult printEntity()

- *printEntity*
public OperationResult printEntity(java.lang.String reportPath, java.util.HashMap parameters)

- *queryEntity*
public ArrayList queryEntity()

- *queryEntity*
public ArrayList queryEntity(java.lang.String fieldId, java.lang.String value)

- *queryEntityWithFormat*
public ArrayList queryEntityWithFormat(java.lang.String fieldId, java.lang.String value)

- *queryFreeEntities*
public ArrayList queryFreeEntities(com.imagos.axilice.core.common.MultiSelectionSubentityOT mSubE)
 - **Usage**
 - * Search entities not associated with a parent entity (null parent field)
 - **Parameters**
 - * mSubE - SubentityOT for the parent entity that must be null
 - **Returns** - ArrayList. List of entities found

- *querySubentities*
public ArrayList querySubentities(com.imagos.axilice.core.installer.SubentityOT entData)

- *refreshEntity*
public OperationResult refreshEntity()

- *refreshEntity*
public BaseV0 refreshEntity(com.imagos.axilice.core.common.BaseV0 entity)

- *refreshParentSelectedEntity*
public OperationResult refreshParentSelectedEntity()
 - **Usage**
 - * Reload from database the value of the parent entity of the selected entity

- **Returns** - OperationResult. The result of the operation.
- - *removeRegistries*
 public OperationResult removeRegistries(java.util.ArrayList selRegistries,
 java.lang.String parentField)
 - **Usage**
 - * Deletes a list of the records stored by the controller. Also clean your parent fields and updates the database
 - **Parameters**
 - * selRegistries - Index list to remove
 - * parentField - Field name that links to the parent entity
 - **Returns** - OperationResult. Result of the operation
- - *saveEntity*
 public OperationResult saveEntity()
- - *saveEntity*
 public OperationResult saveEntity(java.util.ArrayList lIds)
- - *saveEntity*
 public OperationResult saveEntity(com.imagos.axilice.core.common.BaseVO b)
 - **Usage**
 - * Save or update a entity to database
 - **Parameters**
 - * b - BaseVO. Entity to save or update
 - **Returns** - OperationResult. Result of the operation
- - *selectEntity*
 public void selectEntity(java.lang.Long idEntity)
- - *selectSubentity*
 public void selectSubentity(java.lang.String parentFieldId,
 com.imagos.axilice.core.common.BaseVO subentityOT)
- - *selectTreeItem*
 public void selectTreeItem(java.lang.Long id)
 - **Usage**
 - * Select a item in the entity tree. No me gusta esto aqui
 - **Parameters**
 - * id - Long. Index of item to select
- - *setDialogName*
 public void setDialogName(java.lang.String dialogName)
- - *setEditGladePath*
 public void setEditGladePath(java.lang.String editGladePath)
- - *setEditionMode*
 public abstract void setEditionMode(boolean enabled)
- - *setRegistries*
 public void setRegistries(java.util.Collection collection)
 - **Usage**
 - * Setter for lRegistries
- - *setWidgetsValues*
 public abstract void setWidgetsValues(com.imagos.axilice.core.common.BaseVO
 entity)
 - **Usage**

- * Set the values of the widgets on the edit dialog.
- Must be overwritten.
- **Parameters**
 - * **entity** - BaseVO. Entity that contains the values to be displayed
- *showErrorMessages*

public void **showErrorMessages**(com.imagos.axilice.core.util.OperationResult **result**)

- *updateEntityData*

public void **updateEntityData**()

- *updateRegistry*
public void **updateRegistry**(com.imagos.axilice.core.common.BaseVO **ent**)
 - **Usage**
 - * Add or update an entity (to the controller records)
 - **Parameters**
 - * **ent** - Entity to add or update
- *updateViews*

public void **updateViews**(boolean **updateFromParent**)
 - **Usage**
 - * Force the update of the views associated
 - **Parameters**
 - * **updateFromParent** - Indicates whether the view must be updated by reading the values from the parent entity. takes effect only when the entity is detail (is embedded in another)
- *validateEntity*

public abstract OperationResult **validateEntity**()
 - **Usage**
 - * Validate the values of the widgets in the edit dialog.
 - Must be overwritten.
 - **Returns** - OperationResult. Result of the validation.
- *viewEntity*

public OperationResult **viewEntity**()

METHODS INHERITED FROM CLASS com.imagos.axilice.core.installer.EntityLauncher

(in 24.2.2, page 532)

- *addEntity*
public abstract ArrayList **addEntity**()
 - **Usage**
 - * Add entity button clicked handler. Show the entity content screen ready to start inserting data for a new instance of the entity (new registry).
 - **Returns** - ArrayList<ResultMessage>, list of errors if any has taken place, null in any other case.
- *buildEditEntityGUI*

public void **buildEditEntityGUI**()
 - **Usage**

- * Builds the edit entity dialog window from the corresponding glade interface file. Stores the entity dialog instance for posterior accesses.

- *buildEntity*

```
public AxiliceEntityView buildEntity( )
```

- **Usage**

- * Builds the entity GUI and show it on screen to the user.

- *buildEntity*

```
public AxiliceDetailEntityView buildEntity( SwtWidget parent,
com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData,
java.lang.String rootName )
```

- **Usage**

- * Builds the entity GUI and show it on screen to the user. The entity GUI will be embedded into the parent composite widget.

- **Parameters**

- * **parent** - Composite, parent holder for embedding the entity view.
- * **subentityData** - SubentityOT, data of the subentity to build if the entity launcher will be used to handle a multi-selection detail entity view.
- * **rootName** - String, name of the root widget that will be embedded into the master entity edition panel.

- **Returns** - AxiliceDetailEntityView, the entity detail view built.

- *cancelEntityEdition*

```
public void cancelEntityEdition( )
```

- *clearEntityData*

```
public abstract void clearEntityData( )
```

- **Usage**

- * Clears the entity content panel by resetting the content widgets.

- *delEntity*

```
public abstract OperationResult delEntity( )
```

- **Usage**

- * Delete entity button clicked handler. Deletes the selected instance of the entity (selected registry) from the database.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *delEntity*

```
public abstract OperationResult delEntity( java.util.ArrayList lSelRegistries )
```

- **Usage**

- * Delete entity button clicked handler. Deletes the selected instances of the entity (selected registries) from the database. Its a multiple delete method.

- **Parameters**

- * **lSelRegistries** - ArrayList<Long>, list of selected instances of the entity (selected registries). The list will contain only the internal id of the selected registries.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *deselectSubentity*

```
public void deselectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

- **Usage**

- * Updates an entity instance by removing the selected value for a subentity field.
- **Parameters**
 - * **parentFieldId** - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * **subentityOT** - BaseVO, subentity's deselected instance (registry) value.

- *editEntity*
 public abstract OperationResult **editEntity**()
 - **Usage**
 - * Edit entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), and allows the user to modify any content of the entity registry.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *fillEntityData*
 public abstract void **fillEntityData**()
 - **Usage**
 - * Fills the entity content panel with the data of the selected instance (registry) of the entity.

- *fireExceptionAndExit*
 public void **fireExceptionAndExit**(java.lang.String msg)
 - **Usage**
 - * Fire a exception and stop the program
 - **Parameters**
 - * **msg** -

- *formatRegistries*
 public ArrayList **formatRegistries**()
- *formatRegistries*
 public ArrayList **formatRegistries**(java.util.ArrayList lRegistries)
- *getDialog*
 public SwtDialog **getDialog**(java.lang.String dialogName)
- *getEntity*
 public EntityOT **getEntity**()
 - **Usage**
 - * Gets the entity data.
 - **Returns** - EntityOT, entity data.

- *getEntityData*
 public abstract BaseVO **getEntityData**()
 - **Usage**
 - * Gets the content panel's data typed by the user formatted as an entity VO object.

- *getEntityEditViewName*
 public String **getEntityEditViewName**(java.lang.String parentGuiName)
- *getEntityFields*
 public abstract ArrayList **getEntityFields**()
 - **Usage**
 - * Gets the list of defined fields for an entity
 - **Returns** - ArrayList, list of entity fields.

- *getEntityViewName*
public String **getEntityViewName**(java.lang.String parentGuiName)
- *getLRegistries*
public ArrayList **getLRegistries**()
 - **Usage**
 - * Getter for IRegistries
- *getRegistry*
public BaseVO **getRegistry**(long id)
- *getRevision*
public static String **getRevision**()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
- *getSelectedEntity*
public BaseVO **getSelectedEntity**()
 - **Usage**
 - * Gets the selected instance (registry) of the entity.
 - **Returns** - BaseVO, selected instance (registry) of the entity.
- *getSelectedId*
public Long **getSelectedId**()
 - **Usage**
 - * Gets the internal id of the selected entity registry.
 - **Returns** - Long, internal id of the selected entity registry, null in case no entity registry has been selected yet.
- *getSwtWidget*
public SwtWidget **getSwtWidget**(java.lang.String widgetName)
- *getWidget*
public Widget **getWidget**(java.lang.String widgetName)
- *initEntityLauncher*
public void **initEntityLauncher**(Logger logger, Gui parentGui, SwtWidget parentPanel)
 - **Usage**
 - * Inits the entity launcher context and prepares the launcher to be started.
 - **Parameters**
 - * **logger** - Logger, logger instance for printing logging messages.
 - * **parentGuiName** - String, name of the JavaGlade instance that built the parent where the entity will be embedded.
 - * **parentPanel** - Composite, parent panel where the entity contents will be embedded.
- *isDetail*
public boolean **isDetail**()
 - **Usage**
 - * Indicates whether is a main entity or is embedded in another
- *on_entitySelectionButton_clicked*
public void **on_entitySelectionButton_clicked**(Event e)
 - **Usage**
 - * Handler for the subentity selection buttons (magnifying glass buttons) clicked. Shows the selection dialog for selecting a valid instance of a subentity.

- **Parameters**
 - * **e** - Event, magnifying glass button clicked event data.

- *printEntity*

```
public abstract OperationResult printEntity( )
```

 - **Usage**
 - * Print entity button clicked handler. Builds the report with the data of the selected instance of the entity (selected registry), and allows the user to print the elaborated report.
 - **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *printWidgets*

```
public void printWidgets( )
```

- *queryEntity*

```
public abstract ArrayList queryEntity( )
```

 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) available.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.

- *queryEntity*

```
public abstract ArrayList queryEntity( java.lang.String fieldId, java.lang.String value )
```

 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria.
 - **Parameters**
 - * **fieldId** - String, name of the entity field that will be queried.
 - * **value** - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList, list of entity instances (registries) taken as the result of the query operation.

- *queryEntityWithFormat*

```
public abstract ArrayList queryEntityWithFormat( java.lang.String fieldId, java.lang.String value )
```

 - **Usage**
 - * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria and show them (formatted as a list) to the user in the entity's screen.
 - **Parameters**
 - * **fieldId** - String, name of the entity field that will be queried.
 - * **value** - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList<RegistryOT>, list of entity instances (registries) taken as the result of the query operation.

- *querySubentities*

```
public abstract ArrayList querySubentities( com.imagos.axilice.core.installer.SubentityOT entData )
```

 - **Usage**
 - * Queries the selected entity for all the detail subentities associated with it.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to query for.

- **Returns** - ArrayList, list of detail subentity registries assigned to the master detail registry.

- *saveEntity*

```
public abstract OperationResult saveEntity( )
```

 - **Usage**
 - * Save entity button clicked handler. Update the system with the data of the edited entity registry. It's called for saving new entity registries (add a new registry) or for updating the contents of existing entity registries.
 - **Returns** - OperationResult, Result of the operation and list of messages generated through the operation.

- *saveEntityRegistry*

```
public void saveEntityRegistry( )
```
- *selectEntity*

```
public abstract void selectEntity( java.lang.Long idEntity )
```

 - **Usage**
 - * Marks an instance (registry) of the entity as the selected.
 - **Parameters**
 - * idEntity - Long, internal id of the selected instance (registry).

- *selectSubentity*

```
public void selectSubentity( java.lang.String parentFieldId,
com.imagos.axilice.core.common.BaseVO subentityOT )
```

 - **Usage**
 - * Updates an entity instance by setting up the selected value for a subentity field.
 - **Parameters**
 - * parentFieldId - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * subentityOT - BaseVO, subentity's selected instance (registry) value.

- *setEntity*

```
public void setEntity( com.imagos.axilice.core.installer.EntityOT entity )
```

 - **Usage**
 - * Sets up the entity name.
 - **Parameters**
 - * name - String, entity name.

- *setRegistries*

```
public void setRegistries( java.util.ArrayList lRegistries )
```

 - **Usage**
 - * Sets the list of registries that are showed in the entity view controlled by the Entity Launcher instance. It's used for detail entities where queries are made in the master entity launcher.
 - **Parameters**
 - * lRegistries - ArrayList<BaseVO>, list of shown registries of the entity.

- *showSelectionDialog*

```
public void showSelectionDialog( com.imagos.axilice.core.installer.SubentityOT
entData )
```

 - **Usage**
 - * Opens the selection dialog that enables the user to select an instance of a subentity.
 - **Parameters**
 - * entData - SubentityOT, data of the detail subentity to select.

- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT
entData, long selEntity, boolean readonly )
```

- Usage

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

- Parameters

- * **entData** - SubentityOT, data of the detail subentity to select.
 - * **selEntity** - long, id of the selected instance of the subentity.
 - * **readonly** - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT
entData, java.lang.String parentGuiName, long selEntity, boolean readonly )
```

- Usage

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

- Parameters

- * **entData** - SubentityOT, data of the detail subentity to select.
 - * **parentGuiName** - String,
 - * **selEntity** - long, id of the selected instance of the subentity.
 - * **readonly** - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

- *showSubentitySelectionDialog*

```
public void showSubentitySelectionDialog(
com.imagos.axilice.core.installer.SubentityOT entData )
```

- Usage

- * Opens the selection dialog that enables the user to select an instance of a multiple selection detail subentity.

- Parameters

- * **entData** - SubentityOT, data of the detail subentity to select.

- *undoEntityEdition*

```
public void undoEntityEdition( )
```

- *updateEntityData*

```
public abstract void updateEntityData( )
```

- Usage

- * Updates the selected instance (registry) data with the content panel's data typed by the user.

- *viewEntity*

```
public abstract OperationResult viewEntity( )
```

- Usage

- * View entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), in read-only mode.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

DECLARATION

```
public class RappelTypeListener
extends GenericHandler
```

CONSTRUCTORS

- *RappelTypeListener*

```
public RappelTypeListener(
    com.imagos.axilice.client.partners.RappelIntervalClient eL )
```

METHODS

- *on.button.cancel.rappel.type.rappel.interval.dialog.clicked*

```
public void on.button.cancel.rappel.type.rappel.interval.dialog.clicked( Event
    e )
```

METHODS INHERITED FROM CLASS GenericHandler

27.1.13 CLASS VendorClient

DECLARATION

```
public class VendorClient
extends com.imagos.axilice.client.gui.common.CommonClient
```

CONSTRUCTORS

- *VendorClient*

```
public VendorClient( )
```

METHODS

- *build.extension.fields.view*

```
public void build.extension.fields.view( )
```
- *buildSelectionSets*

```
public void buildSelectionSets( )
```
- *fillEntityData*

```
public void fillEntityData( )
```

- *getEntityData*
public BaseVO **getEntityData**()
- *getWidgets*
public void **getWidgets**()
- *init_specific_components*
public void **init_specific_components**()
- *setEditionMode*
public void **setEditionMode**(boolean **enabled**)
- *setWidgetsValues*
public void **setWidgetsValues**(com.imagos.axilice.core.common.BaseVO **entity**)
- *validateEntity*
public **OperationResult** **validateEntity**()

METHODS INHERITED FROM CLASS com.imagos.axilice.client.gui.common.CommonClient

(in 17.1.2, page 402)

- *addAxiliceEventListener*
public void **addAxiliceEventListener**(
com.imagos.axilice.client.gui.AxiliceEventListener **l**)
- *addEntity*
public ArrayList **addEntity**()
- *addExistingSubentity*
public void **addExistingSubentity**(com.imagos.axilice.core.installer.SubentityOT
ent)
 - **Usage**
 - * Shows a dialog for selection of a new subentity
 - **Parameters**
 - * **ent** - SubentityOT. Encapsulate data type of subentity
- *addSubentity*
public void **addSubentity**(com.imagos.axilice.core.common.BaseVO **subentity**,
com.imagos.axilice.core.common.MultiSelectionSubentityOT **subE**)
 - **Usage**
 - * Adds a existent entity to a multiselection list
 - **Parameters**
 - * **subentity** - Subentity to be added
 - * **subE** - Object that encapsulates information about the subentity
- *addTrazabilityTree*
public void **addTrazabilityTree**(TabFolder **tabFolder**)
- *build_extension_fields_view*
public abstract void **build_extension_fields_view**()
 - **Usage**
 - * Build the view of the extension fields.

It must implement only those clients who require it

- *buildEntity*
public AxiliceEntityView buildEntity()
- *buildEntityDetail*
public AxiliceEntityView buildEntityDetail(
com.imagos.axilice.core.common.MultiSelectionSubentityOT subentityData)
- *buildEntitySelectionView*
public AxiliceEntitySelectionView buildEntitySelectionView()
 - **Usage**
 - * Build view of selection entities
 - **Returns** - AxilicePartnerSelectionView. SelectionView created
- *buildSelectionSets*
public abstract void buildSelectionSets()
 - **Usage**
 - * Build 2 sets of SubentityOT: lSelectionSubentities and lMultiSelectionSubentities

These sets are used to connect the widgets with the handlers for the selection buttons and the multiseleccion panels

It must implement only those clients who require it
- *buildSubentityMultiSelection*
public void buildSubentityMultiSelection()
- *clearEntityData*
public void clearEntityData()
- *clearParentField*
public OperationResult clearParentField(java.util.ArrayList selRegistries,
java.lang.String parentField)
 - **Usage**
 - * Clears the value of the field indicated
 - **Parameters**
 - * selRegistries - ArrayList. Index list of records to process
 - * parentField - Name of the field to clear
 - **Returns** - OperationResult. Result of operation
- *delEntity*
public OperationResult delEntity()
- *delEntity*
public OperationResult delEntity(java.util.ArrayList selRegistries)
- *delEntity*
public OperationResult delEntity(com.imagos.axilice.core.common.BaseVO b)
 - **Usage**
 - * Delete a entity from database
 - **Parameters**
 - * b - BaseVO. Entity to delete
 - **Returns** - OperationResult. Result of the operation
- *delSubentities*
public OperationResult delSubentities(java.util.ArrayList selRegistries)
 - **Usage**

* Delete a set of subentities from record list of controller, not from database

Set the parent field of subentity to null

– **Parameters**

* **selRegistries** - List of index to delete

– **Returns** - **OperationResult**. Result of the operation

• *editEntity*

public OperationResult editEntity()

• *fillEntityData*

public void fillEntityData()

• *fillEntityExtensionTable*

public void fillEntityExtensionTable(java.lang.Long id)

– **Usage**

* Retrieve extension field values from the server for a given entity and displays them in the view

– **Parameters**

* **id** - Long. Key of the entity whose values are to be displayed

• *fillEntityTable*

public void fillEntityTable()

– **Usage**

* Load records from database and fill the entity view table and controller record list

• *fillEntityTable*

public void fillEntityTable(java.util.ArrayList lR)

– **Usage**

* Fill the entity view table with the records given

– **Parameters**

* **lR** - ArrayList. Records to be displayed

• *getClassEntity*

public Class getClassEntity()

– **Returns** - Class of the entity for which the controller is built

• *getDialogName*

public String getDialogName()

• *getEditGladePath*

public String getEditGladePath()

• *getEditionGui*

public Gui getEditionGui()

– **Usage**

* Returns the edition Gui instance.

– **Returns** - Gui, entity edition Gui.

• *getEditWindow*

public SwtTopLevelWindow getEditWindow()

• *getEntityData*

public abstract BaseVO getEntityData()

• *getEntityFields*

public ArrayList getEntityFields()

- *getLogger*
public Logger getLogger()
- *getParentEntityLauncher*
public EntityLauncher getParentEntityLauncher()
 – **Returns** - EntityLauncher of parent entity if is a embedded entity or null otherwise
- *getRegistries*
public Collection getRegistries()
 – **Returns** - lRegistries
- *getRegistries*
public List getRegistries(java.util.ArrayList lIds)
 – **Usage**
 * Get a list of entities stored by the controller
 – **Parameters**
 * lIds - Index list
 – **Returns** - List of entities
- *getView*
public AxiliceEntityView getView()
 – **Returns** - AxiliceEntityView. Main view or Detail view if is a embedded entity.
- *getWidgets*
public abstract void getWidgets()
 – **Usage**
 * Get the widgets from the glade and stored in local variables.

 Must be overwritten
- *init_specific_components*
public abstract void init_specific_components()
 – **Usage**
 * Initializes the specific components of the client
- *printEntity*
public OperationResult printEntity()
- *printEntity*
public OperationResult printEntity(java.lang.String reportPath, java.util.HashMap parameters)
- *queryEntity*
public ArrayList queryEntity()
- *queryEntity*
public ArrayList queryEntity(java.lang.String fieldId, java.lang.String value)
- *queryEntityWithFormat*
public ArrayList queryEntityWithFormat(java.lang.String fieldId, java.lang.String value)
- *queryFreeEntities*
public ArrayList queryFreeEntities(com.imagos.axilice.core.common.MultiSelectionSubentityOT mSubE)
 – **Usage**
 * Search entities not associated with a parent entity (null parent field)
 – **Parameters**

- * mSubE - SubentityOT for the parent entity that must be null
 - **Returns** - ArrayList. List of entities found

- *querySubentities*
 public ArrayList **querySubentities**(com.imagos.axilice.core.installer.SubentityOT entData)

- *refreshEntity*
 public OperationResult **refreshEntity**()

- *refreshEntity*
 public BaseVO **refreshEntity**(com.imagos.axilice.core.common.BaseVO entity)

- *refreshParentSelectedEntity*
 public OperationResult **refreshParentSelectedEntity**()
 - **Usage**
 - * Reload from database the value of the parent entity of the selected entity
 - **Returns** - OperationResult. The result of the operation.

- *removeRegistries*
 public OperationResult **removeRegistries**(java.util.ArrayList selRegistries, java.lang.String parentField)
 - **Usage**
 - * Deletes a list of the records stored by the controller. Also clean your parent fields and updates the database
 - **Parameters**
 - * selRegistries - Index list to remove
 - * parentField - Field name that links to the parent entity
 - **Returns** - OperationResult. Result of the operation

- *saveEntity*
 public OperationResult **saveEntity**()

- *saveEntity*
 public OperationResult **saveEntity**(java.util.ArrayList lIds)

- *saveEntity*
 public OperationResult **saveEntity**(com.imagos.axilice.core.common.BaseVO b)
 - **Usage**
 - * Save or update a entity to database
 - **Parameters**
 - * b - BaseVO. Entity to save or update
 - **Returns** - OperationResult. Result of the operation

- *selectEntity*
 public void **selectEntity**(java.lang.Long idEntity)

- *selectSubentity*
 public void **selectSubentity**(java.lang.String parentFieldId, com.imagos.axilice.core.common.BaseVO subentityOT)

- *selectTreeItem*
 public void **selectTreeItem**(java.lang.Long id)
 - **Usage**
 - * Select a item in the entity tree. No me gusta esto aqui
 - **Parameters**
 - * id - Long. Index of item to select

- *setDialogName*
public void setDialogName(java.lang.String dialogName)
- *setEditGladePath*
public void setEditGladePath(java.lang.String editGladePath)
- *setEditionMode*
public abstract void setEditionMode(boolean enabled)
- *setRegistries*
public void setRegistries(java.util.Collection collection)
 - **Usage**
 - * Setter for IRegistries

- *setWidgetsValues*
public abstract void setWidgetsValues(com.imagos.axilice.core.common.BaseVO entity)
 - **Usage**
 - * Set the values of the widgets on the edit dialog.
 - Must be overwritten.
 - **Parameters**
 - * **entity** - BaseVO. Entity that contains the values to be displayed

- *showErrorMessages*
public void showErrorMessages(com.imagos.axilice.core.util.OperationResult result)

- *updateEntityData*
public void updateEntityData()
- *updateRegistry*
public void updateRegistry(com.imagos.axilice.core.common.BaseVO ent)
 - **Usage**
 - * Add or update an entity (to the controller records)
 - **Parameters**
 - * **ent** - Entity to add or update

- *updateViews*
public void updateViews(boolean updateFromParent)
 - **Usage**
 - * Force the update of the views associated
 - **Parameters**
 - * **updateFromParent** - Indicates whether the view must be updated by reading the values from the parent entity. takes effect only when the entity is detail (is embedded in another)

- *validateEntity*
public abstract OperationResult validateEntity()
 - **Usage**
 - * Validate the values of the widgets in the edit dialog.
 - Must be overwritten.
 - **Returns** - OperationResult. Result of the validation.

- *viewEntity*
public OperationResult viewEntity()

METHODS INHERITED FROM CLASS com.imagos.axilice.core.installer.EntityLauncher

(in 24.2.2, page 532)

- *addEntity*
 public abstract ArrayList **addEntity**()
 – **Usage**
 * Add entity button clicked handler. Show the entity content screen ready to start inserting data for a new instance of the entity (new registry).
 – **Returns** - ArrayList<ResultMessage>, list of errors if any has taken place, null in any other case.

- *buildEditEntityGUI*
 public void **buildEditEntityGUI**()
 – **Usage**
 * Builds the edit entity dialog window from the corresponding glade interface file. Stores the entity dialog instance for posterior accesses.

- *buildEntity*
 public AxiliceEntityView **buildEntity**()
 – **Usage**
 * Builds the entity GUI and show it on screen to the user.

- *buildEntity*
 public AxiliceDetailEntityView **buildEntity**(SwtWidget **parent**,
 com.imagos.axilice.core.common.MultiSelectionSubentityOT **subentityData**,
 java.lang.String **rootName**)
 – **Usage**
 * Builds the entity GUI and show it on screen to the user. The entity GUI will be embedded into the parent composite widget.
 – **Parameters**
 * **parent** - Composite, parent holder for embedding the entity view.
 * **subentityData** - SubentityOT, data of the subentity to build if the entity launcher will be used to handle a multi-selection detail entity view.
 * **rootName** - String, name of the root widget that will be embedded into the master entity edition panel.
 – **Returns** - AxiliceDetailEntityView, the entity detail view built.

- *cancelEntityEdition*
 public void **cancelEntityEdition**()

- *clearEntityData*
 public abstract void **clearEntityData**()
 – **Usage**
 * Clears the entity content panel by resetting the content widgets.

- *delEntity*
 public abstract OperationResult **delEntity**()
 – **Usage**
 * Delete entity button clicked handler. Deletes the selected instance of the entity (selected registry) from the database.
 – **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *delEntity*

```
public abstract OperationResult delEntity( java.util.ArrayList lSelRegistries )
```

- **Usage**

- * Delete entity button clicked handler. Deletes the selected instances of the entity (selected registries) from the database. Its a multiple delete method.

- **Parameters**

- * lSelRegistries - ArrayList<Long>, list of selected instances of the entity (selected registries). The list will contain only the internal id of the selected registries.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *deselectSubentity*

```
public void deselectSubentity( java.lang.String parentFieldId,  
com.imagos.axilice.core.common.BaseVO subentityOT )
```

- **Usage**

- * Updates an entity instance by removing the selected value for a subentity field.

- **Parameters**

- * parentFieldId - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * subentityOT - BaseVO, subentity's deselected instance (registry) value.

- *editEntity*

```
public abstract OperationResult editEntity( )
```

- **Usage**

- * Edit entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), and allows the user to modify any content of the entity registry.

- **Returns** - OperationResult, result of the delete operation and list of messages generated during the operation.

- *fillEntityData*

```
public abstract void fillEntityData( )
```

- **Usage**

- * Fills the entity content panel with the data of the selected instance (registry) of the entity.

- *fireExceptionAndExit*

```
public void fireExceptionAndExit( java.lang.String msg )
```

- **Usage**

- * Fire a exception and stop the program

- **Parameters**

- * msg -

- *formatRegistries*

```
public ArrayList formatRegistries( )
```

- *formatRegistries*

```
public ArrayList formatRegistries( java.util.ArrayList lRegistries )
```

- *getDialog*

```
public SwtDialog getDialog( java.lang.String dialogName )
```

- *getEntity*

```
public EntityOT getEntity( )
```

- **Usage**

- * Gets the entity data.

- **Returns** - EntityOT, entity data.
- getEntityData
 public abstract BaseVO **getEntityData**()
 - **Usage**
 - * Gets the content panel's data typed by the user formatted as an entity VO object.
- getEntityEditViewName
 public String **getEntityEditViewName**(java.lang.String parentGuiName)
- getEntityFields
 public abstract ArrayList **getEntityFields**()
 - **Usage**
 - * Gets the list of defined fields for an entity
 - **Returns** - ArrayList, list of entity fields.
- getEntityViewName
 public String **getEntityViewName**(java.lang.String parentGuiName)
- getLRegistries
 public ArrayList **getLRegistries**()
 - **Usage**
 - * Getter for lRegistries
- getRegistry
 public BaseVO **getRegistry**(long id)
- getRevision
 public static String **getRevision**()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
- getSelectedEntity
 public BaseVO **getSelectedEntity**()
 - **Usage**
 - * Gets the selected instance (registry) of the entity.
 - **Returns** - BaseVO, selected instance (registry) of the entity.
- getSelectedId
 public Long **getSelectedId**()
 - **Usage**
 - * Gets the internal id of the selected entity registry.
 - **Returns** - Long, internal id of the selected entity registry, null in case no entity registry has been selected yet.
- getSwtWidget
 public SwtWidget **getSwtWidget**(java.lang.String widgetName)
- getWidget
 public Widget **getWidget**(java.lang.String widgetName)
- initEntityLauncher
 public void **initEntityLauncher**(Logger logger, Gui parentGui, SwtWidget parentPanel)
 - **Usage**
 - * Inits the entity launcher context and prepares the launcher to be started.
 - **Parameters**

- * **logger** - Logger, logger instance for printing logging messages.
- * **parentGuiName** - String, name of the JavaGlade instance that built the parent where the entity will be embedded.
- * **parentPanel** - Composite, parent panel where the entity contents will be embedded.

- *isDetail*

public boolean **isDetail**()

– **Usage**

- * Indicates whether is a main entity or is embedded in another

- *on_entitySelectionButton_clicked*

public void **on_entitySelectionButton_clicked**(Event e)

– **Usage**

- * Handler for the subentity selection buttons (magnifying glass buttons) clicked. Shows the selection dialog for selecting a valid instance of a subentity.

– **Parameters**

- * **e** - Event, magnifying glass button clicked event data.

- *printEntity*

public abstract **OperationResult printEntity**()

– **Usage**

- * Print entity button clicked handler. Builds the report with the data of the selected instance of the entity (selected registry), and allows the user to print the elaborated report.

– **Returns** - **OperationResult**, result of the delete operation and list of messages generated during the operation.

- *printWidgets*

public void **printWidgets**()

- *queryEntity*

public abstract **ArrayList queryEntity**()

– **Usage**

- * Search entity button clicked handler. Searches the system for the entity instances (registries) available.

– **Returns** - **ArrayList**, list of entity instances (registries) taken as the result of the query operation.

- *queryEntity*

public abstract **ArrayList queryEntity**(java.lang.String **fieldId**, java.lang.String **value**)

– **Usage**

- * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria.

– **Parameters**

- * **fieldId** - String, name of the entity field that will be queried.
- * **value** - String, value of the entity field that will be searched for.

– **Returns** - **ArrayList**, list of entity instances (registries) taken as the result of the query operation.

- *queryEntityWithFormat*

public abstract **ArrayList queryEntityWithFormat**(java.lang.String **fieldId**, java.lang.String **value**)

– **Usage**

- * Search entity button clicked handler. Searches the system for the entity instances (registries) that fit a searching criteria and show them (formatted as a list) to the user in the entity's screen.
 - **Parameters**
 - * **fieldId** - String, name of the entity field that will be queried.
 - * **value** - String, value of the entity field that will be searched for.
 - **Returns** - ArrayList<RegistryOT>, list of entity instances (registries) taken as the result of the query operation.

- *querySubentities*

```
public abstract ArrayList querySubentities(
com.imagos.axilice.core.installer.SubentityOT  entData )
```

 - **Usage**
 - * Queries the selected entity for all the detail subentities associated with it.
 - **Parameters**
 - * **entData** - SubentityOT, data of the detail subentity to query for.
 - **Returns** - ArrayList, list of detail subentity registries assigned to the master detail registry.

- *saveEntity*

```
public abstract OperationResult saveEntity( )
```

 - **Usage**
 - * Save entity button clicked handler. Update the system with the data of the edited entity registry. It's called for saving new entity registries (add a new registry) or for updating the contents of existing entity registries.
 - **Returns** - OperationResult, Result of the operation and list of messages generated through the operation.

- *saveEntityRegistry*

```
public void saveEntityRegistry( )
```
- *selectEntity*

```
public abstract void selectEntity( java.lang.Long  idEntity )
```

 - **Usage**
 - * Marks an instance (registry) of the entity as the selected.
 - **Parameters**
 - * **idEntity** - Long, internal id of the selected instance (registry).

- *selectSubentity*

```
public void selectSubentity( java.lang.String  parentFieldId,
com.imagos.axilice.core.common.BaseVO  subentityOT )
```

 - **Usage**
 - * Updates an entity instance by setting up the selected value for a subentity field.
 - **Parameters**
 - * **parentFieldId** - String, name of the parent entity field updated. It's the parent entity field whose subentity value has been setted up.
 - * **subentityOT** - BaseVO, subentity's selected instance (registry) value.

- *setEntity*

```
public void setEntity( com.imagos.axilice.core.installer.EntityOT  entity )
```

 - **Usage**
 - * Sets up the entity name.
 - **Parameters**
 - * **name** - String, entity name.

- *setRegistries*

```
public void setRegistries( java.util.ArrayList lRegistries )
```

- Usage

- * Sets the list of registries that are showed in the entity view controlled by the Entity Launcher instance. It's used for detail entities where queries are made in the master entity launcher.

- Parameters

- * lRegistries - ArrayList<BaseVO>, list of shown registries of the entity.

- *showSelectionDialog*

```
public void showSelectionDialog( com.imagos.axilice.core.installer.SubentityOT entData )
```

- Usage

- * Opens the selection dialog that enables the user to select an instance of a subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.

- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT entData, long selEntity, boolean readonly )
```

- Usage

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.
 - * selEntity - long, id of the selected instance of the subentity.
 - * readonly - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

- *showSubentityDialog*

```
public void showSubentityDialog( com.imagos.axilice.core.installer.SubentityOT entData, java.lang.String parentGuiName, long selEntity, boolean readonly )
```

- Usage

- * Opens the edition dialog that enables the user to update or view the instance data of a multiple selection detail subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.
 - * parentGuiName - String,
 - * selEntity - long, id of the selected instance of the subentity.
 - * readonly - boolean, true for enabling the read-only mode (only view); false for allowing modification of the showed data.

- *showSubentitySelectionDialog*

```
public void showSubentitySelectionDialog( com.imagos.axilice.core.installer.SubentityOT entData )
```

- Usage

- * Opens the selection dialog that enables the user to select an instance of a multiple selection detail subentity.

- Parameters

- * entData - SubentityOT, data of the detail subentity to select.

- *undoEntityEdition*

```
public void undoEntityEdition( )
```

- *updateEntityData*

```
public abstract void updateEntityData( )
```

- **Usage**

- * Updates the selected instance (registry) data with the content panel's data typed by the user.
-

- *viewEntity*

```
public abstract OperationResult viewEntity( )
```

- **Usage**

- * View entity button clicked handler. Show the entity content screen filled with the data of the selected instance of the entity (selected registry), in read-only mode.

- **Returns** - **OperationResult**, result of the delete operation and list of messages generated during the operation.

Chapter 28

Package com.imagos.axilice.server.common

<i>Package Contents</i>	<i>Page</i>
<hr/>	
Classes	
BaseDAO	720
<i>This class implements the general application data access object.</i>	
CommonManagerBuilder	721
<i>...no description...</i>	
CommonManagerImpl	722
<i>...no description...</i>	
<hr/>	

28.1 Classes

28.1.1 CLASS BaseDAO

This class implements the general application data access object. It's a part of the data access layer and gives access to the database data. It implements generic methods for accessing database data. It knows the database structure and implementation.
This class is implemented to access a PostgreSQL database implementation.
Responsibilities:

- Provide statements for general querying of the elements of the database.

DECLARATION

```
public class BaseDAO
extends java.lang.Object
```

CONSTRUCTORS

- *BaseDAO*

```
public BaseDAO( com.imagos.axilice.core.util.ResultMessageCatalog
msgCatalog )
```

 - **Usage**
 - * Default constructor of the BaseDAO class.
 - **Parameters**
 - * **msgCatalog** - ResultMessageCatalog, instance of the catalog of result messages available for reporting any error during the DAO operations.

METHODS

- *existsElement*

```
public boolean existsElement( java.lang.String tableName, java.lang.String
idfieldName, java.lang.Object idElement )
```

 - **Usage**
 - * Checks if an element exists in a database table.
 - **Parameters**
 - * **tableName** - String, name of the database table to be queried.
 - * **idfieldName** - String, name of the internal identifier field of the table.
 - * **idElement** - Object, value of the element to be checked.
 - **Returns** - boolean, True if the element is part of the queried table; False in any other case.
 - **Exceptions**
 - * **AxiliceException**, - in case any error occurs during the query.

- *getRevision*

```
public static String getRevision( )
```

– **Usage**

- * Returns tag Id assigned to CVS source file.

• *getTableElements*

```
public List getTableElements( java.lang.String  tableName,
java.lang.String  idfieldName, java.lang.String  desfieldName )
```

– **Usage**

- * Returns all the elements stored in a database table. Returns a list of pairs key-value with the id of each element and its value (the description of the element that will be showed to the user on the screen).

– **Parameters**

- * **tableName** - String, name of the database table to be queried.
- * **idfieldName** - String, name of the internal identifier field of the table.
- * **valfieldName** - String, name of the value field of the table.

– **Returns** - List, list of elements contained in the table.

– **Exceptions**

- * **AxiliceException**, - in case any error occurs during the query.
-

• *getTableIds*

```
public List getTableIds( java.lang.String  tableName, java.lang.String
idfieldName )
```

– **Usage**

- * Returns all the elements stored in a database table. Returns only the internal identifier (key) of each element.

– **Parameters**

- * **tableName** - String, name of the database table to be queried.
- * **idfieldName** - String, name of the internal identifier field of the table.

– **Returns** - List, list of elements contained in the table.

– **Exceptions**

- * **AxiliceException**, - in case any error occurs during the query.

28.1.2 CLASS CommonManagerBuilder

DECLARATION

```
public class CommonManagerBuilder
extends java.lang.Object
```

CONSTRUCTORS

• *CommonManagerBuilder*

```
public CommonManagerBuilder( )
```

METHODS

- *initCommonManager*

```
public static void initCommonManager( )
```

– Usage

- * Initializes the common manager remote object. Makes the registration and binding of the remote CommonManager service.

28.1.3 CLASS CommonManagerImpl

DECLARATION

```
public class CommonManagerImpl
extends java.rmi.server.UnicastRemoteObject
implements com.imagos.axilice.core.common.CommonManager
```

CONSTRUCTORS

- *CommonManagerImpl*

```
public CommonManagerImpl( )
```

METHODS

- *delBarcodeType*

```
public OperationResult delBarcodeType( java.lang.Long idBarcodeType )
```

- *delBarcodeTypes*

```
public OperationResult delBarcodeTypes( java.util.ArrayList
lSelBarcodeTypes )
```

- *delEntities*

```
public OperationResult delEntities( java.util.ArrayList sel, java.lang.Class
myClass )
```

- *delEntity*

```
public OperationResult delEntity( com.imagos.axilice.core.common.BaseVO b )
```

- *delVatType*

```
public OperationResult delVatType( java.lang.Long idVatType )
```

- *delVatTypes*

```
public OperationResult delVatTypes( java.util.ArrayList lSelVatTypes )
```

- *getBarcodeType*

```
public BarcodeTypeOT getBarcodeType( java.lang.Long idBarcodeType )
```

- *getEntity*

```
public BaseVO getEntity( java.lang.Long id, java.lang.Class myClass )
```

- *getEntity*

```
public BaseVO getEntity( java.lang.String fieldId, java.lang.String value,  
java.lang.Class myClass )
```

- *getExtendedItemValues*

```
public Map getExtendedItemValues( java.lang.Long idValue,  
java.lang.String codExtTable )
```

 - **Usage**
 - * Return all the values stored for an extended item. Include all the fields defined for the extension table and their correspondent value (if anyone has been defined).
 - **Parameters**
 - * idValue - Long, internal id of the extended item.
 - * codExtTable - String, external code of the extension table.
 - **Returns** - Map, set of all the extension fields and their values for the extended item.
 - **Exceptions**
 - * com.imagos.axilice.core.extTables.ExtTableException - , in case any error take place during the query operation.

- *getExtensionTableDefinition*

```
public ExtTableOT getExtensionTableDefinition( java.lang.String  
codExtTable )
```

- *getVatType*

```
public VatTypeOT getVatType( java.lang.Long idVatType )
```

- *queryBarcodeTypes*

```
public List queryBarcodeTypes( java.lang.String fieldId, java.lang.String  
value )
```

- *queryEntities*

```
public List queryEntities( java.lang.String hql )
```

- *queryEntities*

```
public List queryEntities( java.lang.String fieldId,  
com.imagos.axilice.core.common.BaseVO value, java.lang.Class myClass )
```

- *queryEntities*

```
public List queryEntities( java.lang.String fieldId, java.lang.String value,  
java.lang.Class myClass )
```

- *queryVatTypes*

```
public List queryVatTypes( java.lang.String fieldId, java.lang.String  
value )
```

- *saveBarcodeType*

```
public OperationResult saveBarcodeType(  
com.imagos.axilice.core.common.BarcodeTypeOT barcodeType )
```

- *saveEntity*

```
public OperationResult saveEntity( com.imagos.axilice.core.common.BaseVO  
entity )
```

- *saveExtendedItemValues*

```
public OperationResult saveExtendedItemValues( java.util.ArrayList
lExtFields, long idItem )
```

- **Usage**

- * Update the value of all the extension fields defined for an extended item.

- **Parameters**

- * lExtFields - ArrayList, list of all the extension fields whose values must be updated.

- * idItem - long, internal id of the extended item.

- **Returns** - OperationResult, result of the save extended values operation.

- **Exceptions**

- * com.imagos.axilice.core.extTables.ExtTableException - , in case any error take place during the update operation.

- *saveExtendedItemValues*

```
public void saveExtendedItemValues( java.util.Map mExtFields, long
idItem )
```

- **Usage**

- * Update the value of all the extension fields defined for an extended item.

- **Parameters**

- * mExtFields - Map, set of all the extension fields whose values must be updated.

- * idItem - long, internal id of the extended item.

- **Exceptions**

- * com.imagos.axilice.core.extTables.ExtTableException - , in case any error take place during the update operation.

- *saveImage*

```
public OperationResult saveImage( java.lang.String imgName, byte []
contents )
```

- *saveVatType*

```
public OperationResult saveVatType(
com.imagos.axilice.core.common.VatTypeOT vatType )
```

METHODS INHERITED FROM CLASS java.rmi.server.UnicastRemoteObject

- *clone*

```
public Object clone( )
```

- *exportObject*

```
public static RemoteStub exportObject( java.rmi.Remote arg0 )
```

- *exportObject*

```
public static Remote exportObject( java.rmi.Remote arg0, int arg1 )
```

- *exportObject*

```
public static Remote exportObject( java.rmi.Remote arg0, int arg1,
java.rmi.server.RMIClientSocketFactory arg2, java.rmi.server.RMIServerSocketFactory
arg3 )
```

- *unexportObject*

```
public static boolean unexportObject( java.rmi.Remote arg0, boolean arg1 )
```

METHODS INHERITED FROM CLASS java.rmi.server.RemoteServer

- *getClientHost*
public static String getClientHost()
- *getLog*
public static PrintStream getLog()
- *setLog*
public static void setLog(java.io.OutputStream arg0)

METHODS INHERITED FROM CLASS java.rmi.server.RemoteObject

- *equals*
public boolean equals(java.lang.Object arg0)
- *getRef*
public RemoteRef getRef()
- *hashCode*
public int hashCode()
- *toString*
public String toString()
- *toStub*
public static Remote toStub(java.rmi.Remote arg0)

Chapter 29

Package com.imagos.axilice.server.warehouse

<i>Package Contents</i>	<i>Page</i>
<hr/>	
Classes	
WarehouseManagerBuilder.....	727
...no description...	
WarehouseManagerImpl.....	727
...no description...	
<hr/>	

29.1 Classes

29.1.1 CLASS WarehouseManagerBuilder

DECLARATION

```
public class WarehouseManagerBuilder
extends java.lang.Object
```

CONSTRUCTORS

- *WarehouseManagerBuilder*
`public WarehouseManagerBuilder()`

METHODS

- *getLogger*
`public static final Logger getLogger()`
- *initWarehouseManager*
`public static void initWarehouseManager()`

- **Usage**

- * Initializes the warehouse manager remote object. Makes the registration and binding of the remote WarehouseManager service.

29.1.2 CLASS WarehouseManagerImpl

DECLARATION

```
public class WarehouseManagerImpl
extends java.rmi.server.UnicastRemoteObject
implements com.imagos.axilice.core.warehouse.WarehouseManager
```

CONSTRUCTORS

- *WarehouseManagerImpl*
`public WarehouseManagerImpl()`

METHODS

- *delMeasureUnit*
public OperationResult delMeasureUnit(java.lang.Long idMeasureUnit)
- *delMeasureUnits*
public OperationResult delMeasureUnits(java.util.ArrayList lSelMeasureUnits)
- *delProduct*
public OperationResult delProduct(java.lang.Long idProduct)
- *delProductGroup*
public OperationResult delProductGroup(java.lang.Long idGroup)
- *delProductGroups*
public OperationResult delProductGroups(java.util.ArrayList lSelGroups)
- *delProducts*
public OperationResult delProducts(java.util.ArrayList lSelProducts)
- *getMeasureUnit*
public MeasureUnitOT getMeasureUnit(java.lang.Long idMeasureUnit)
- *getNextProductGroupId*
public long getNextProductGroupId()
- *getProduct*
public ProductOT getProduct(java.lang.Long idProduct)
- *getProductGroup*
public ProductGroupOT getProductGroup(java.lang.Long idGroup)
- *queryMeasureUnits*
public List queryMeasureUnits(java.lang.String fieldId, java.lang.String value)
- *queryProductGroups*
public List queryProductGroups(java.lang.String fieldId, java.lang.String value)
- *queryProducts*
public List queryProducts(java.lang.String fieldId, java.lang.String value)
- *resetNextProductGroupId*
public void resetNextProductGroupId(long idGroup)
- *saveMeasureUnit*
public OperationResult saveMeasureUnit(com.imagos.axilice.core.warehouse.MeasureUnitOT measureUnit)
- *saveProduct*
public OperationResult saveProduct(com.imagos.axilice.core.warehouse.ProductOT product)
- *saveProductGroup*
public OperationResult saveProductGroup(com.imagos.axilice.core.warehouse.ProductGroupOT group)

METHODS INHERITED FROM CLASS `java.rmi.server.UnicastRemoteObject`

- *clone*
`public Object clone()`
- *exportObject*
`public static RemoteStub exportObject(java.rmi.Remote arg0)`
- *exportObject*
`public static Remote exportObject(java.rmi.Remote arg0, int arg1)`
- *exportObject*
`public static Remote exportObject(java.rmi.Remote arg0, int arg1, java.rmi.server.RMIClientSocketFactory arg2, java.rmi.server.RMIServerSocketFactory arg3)`
- *unexportObject*
`public static boolean unexportObject(java.rmi.Remote arg0, boolean arg1)`

METHODS INHERITED FROM CLASS `java.rmi.server.RemoteServer`

- *getClientHost*
`public static String getClientHost()`
- *getLog*
`public static PrintStream getLog()`
- *setLog*
`public static void setLog(java.io.OutputStream arg0)`

METHODS INHERITED FROM CLASS `java.rmi.server.RemoteObject`

- *equals*
`public boolean equals(java.lang.Object arg0)`
- *getRef*
`public RemoteRef getRef()`
- *hashCode*
`public int hashCode()`
- *toString*
`public String toString()`
- *toStub*
`public static Remote toStub(java.rmi.Remote arg0)`

Chapter 30

Package

com.imagos.axilice.core.traceability.tree.bl

Package Contents

Page

Classes

TraceabilityTreeBLL	731
<i>Business Logic class that implements the Traceability Tree Viewer business logic processes.</i>	

30.1 Classes

30.1.1 CLASS TraceabilityTreeBLL

Business Logic class that implements the Traceability Tree Viewer business logic processes.

Responsibilities:

- Validation of the traceability tree contents.
- Traceability tree creation.
- Save traceability tree content to a traceability file.
- Remove traceability tree content.
- Build a traceability tree from a traceability file.

DECLARATION

```
public class TraceabilityTreeBLL
extends java.lang.Object
```

METHODS

- *getInstance*

```
public static final TraceabilityTreeBLL getInstance( )
```

 - **Usage**
 - * Returns the unique instance of the TraceabilityTreeBLL class. Implements the singleton design pattern.
 - **Returns** - TraceabilityTreeBLL, unique instance of TraceabilityTreeBLL.
- *getRevision*

```
public static String getRevision( )
```

 - **Usage**
 - * Returns tag Id assigned to CVS source file.
- *isTreeItemCorrect*

```
public boolean isTreeItemCorrect( java.util.ArrayList lResult )
```

 - **Usage**
 - * This method analyses the list of result messages got from a tree item validation process, to determine if the item is valid or not.
 - **Parameters**
 - * lResult - ArrayList<ResultMessage>, List of messages got from the tree item validation process.
 - **Returns** - boolean, True if the item is valid; False in any other case.
 - **See Also**
 - *
com.imagos.axilice.traceability.bl.TraceabilityTreeBLL.validateTreeItem

- *validateTreeItem*

```
public ArrayList validateTreeItem( int    type, java.lang.Object  info, int  
parentType )
```

- **Usage**

- * Validates the item, checking that the item data is correct. It also checks the compatibility amongst the item and its parent item.

- **Parameters**

- * **type** - int, item type.
 - * **info** - Object, item data.
 - * **parentType** - int, parent item type.

- **Returns** - ArrayList<ResultMessage>, List of messages with the result of the validation process. It may be an OK message if the validation was correct, or a list of detailed error messages ready to be showed to the user in case the validation was incorrect.

Chapter 31

Package

com.imagos.axilice.core.common

Package Contents

Page

Interfaces

CommonManager	735
<i>...no description...</i>	

Classes

BarcodeTypeOT	737
<i>Store class to hold the data of a barcode type.</i>	
BaseVO	740
<i>...no description...</i>	
CountryOT	741
<i>Store class to hold the data of a country.</i>	
CurrencyOT	743
<i>...no description...</i>	
DocumentTotal	746
<i>...no description...</i>	
FlagOT	749
<i>Store class to hold the data of a flag.</i>	
ImageComparator	751
<i>...no description...</i>	
ImageOT	752
<i>Store class to hold the data of an application image resource.</i>	
LineaDesgloseIVA	755
<i>...no description...</i>	
MultiSelectionSubentityOT	756
<i>...no description...</i>	
PriceListLineOT	758
<i>...no description...</i>	
PriceListOT	761
<i>Store class to hold the data of a price list.</i>	
RateOT	763
<i>...no description...</i>	
RegistryOT	765
<i>...no description...</i>	

SelectionSubentityOT	766
<i>...no description...</i>	
StateOT	768
<i>Store class to hold the data of a province.</i>	
ValueOT	770
<i>This class implements a generic container for storing the data of a pair key-value, suitable for filling a combobox.</i>	
VatTypeOT	772
<i>Store class to hold the data of a VAT type.</i>	

31.1 Interfaces

31.1.1 INTERFACE CommonManager

DECLARATION

```
public interface CommonManager
implements java.rmi.Remote
```

METHODS

- *delBarcodeType*
 public **OperationResult** **delBarcodeType**(java.lang.Long idBarcodeType)
- *delBarcodeTypes*
 public **OperationResult** **delBarcodeTypes**(java.util.ArrayList lSelBarcodeTypes)
- *delEntities*
 public **OperationResult** **delEntities**(java.util.ArrayList lSel, java.lang.Class myClass)
- *delEntity*
 public **OperationResult** **delEntity**(com.imagos.axilice.core.common.BaseVO b)
- *delVatType*
 public **OperationResult** **delVatType**(java.lang.Long idVatType)
- *delVatTypes*
 public **OperationResult** **delVatTypes**(java.util.ArrayList lSelVatTypes)
- *getBarcodeType*
 public **BarcodeTypeOT** **getBarcodeType**(java.lang.Long idBarcodeType)
- *getEntity*
 public **BaseVO** **getEntity**(java.lang.Long id, java.lang.Class myClass)
- *getEntity*
 public **BaseVO** **getEntity**(java.lang.String fieldId, java.lang.String value, java.lang.Class myClass)
- *getExtendedItemValues*
 public **Map** **getExtendedItemValues**(java.lang.Long idValue, java.lang.String codExtTable)

– Usage

- * Return all the values stored for an extended item. Include all the fields defined for the extension table and their correspondent value (if anyone has been defined).

– Parameters

- * idValue - Long, internal id of the extended item.

- * `codExtTable` - String, external code of the extension table.
 - **Returns** - Map, set of all the extension fields and their values for the extended item.
 - **Exceptions**
 - * `ExtTableException`, - in case any error take place during the query operation.

- *getExtensionTableDefinition*

```
public ExtTableOT getExtensionTableDefinition( java.lang.String
codExtTable )
```

 - **Usage**
 - * Return the extension table definition, including the list of all the extension fields defined for the table.
 - **Parameters**
 - * `codExtTable` - String, external code of the extension table.
 - **Returns** - `ExtTableOT`, extension table definition.
 - **Exceptions**
 - * `com.imagos.axilice.core.extTables.ExtTableException` - in case any error take place during the query operation.

- *getVatType*

```
public VatTypeOT getVatType( java.lang.Long idVatType )
```

- *queryBarcodeTypes*

```
public List queryBarcodeTypes( java.lang.String fieldId, java.lang.String
value )
```

- *queryEntities*

```
public List queryEntities( java.lang.String sql )
```

- *queryEntities*

```
public List queryEntities( java.lang.String fieldId,
com.imagos.axilice.core.common.BaseVO value, java.lang.Class myClass )
```

- *queryEntities*

```
public List queryEntities( java.lang.String fieldId, java.lang.String value,
java.lang.Class myClass )
```

- *queryVatTypes*

```
public List queryVatTypes( java.lang.String fieldId, java.lang.String
value )
```

- *saveBarcodeType*

```
public OperationResult saveBarcodeType(
com.imagos.axilice.core.common.BarcodeTypeOT barcodeType )
```

- *saveEntity*

```
public OperationResult saveEntity( com.imagos.axilice.core.common.BaseVO
entity )
```

- *saveExtendedItemValues*

```
public OperationResult saveExtendedItemValues( java.util.ArrayList
lExtFields, long idItem )
```

 - **Usage**

- * Update the value of all the extension fields defined for an extended item.
 - **Parameters**
 - * **lExtFields** - ArrayList, list of all the extension fields whose values must be updated.
 - * **idItem** - long, internal id of the extended item.
 - **Returns** - OperationResult, result of the save extended values operation.
 - **Exceptions**
 - * **ExtTableException**, - in case any error take place during the update operation.
-
- *saveExtendedItemValues*

```
public void saveExtendedItemValues( java.util.Map  mExtFields, long
idItem )
```

 - **Usage**
 - * Update the value of all the extension fields defined for an extended item.
 - **Parameters**
 - * **mExtFields** - Map, set of all the extension fields whose values must be updated.
 - * **idItem** - long, internal id of the extended item.
 - **Exceptions**
 - * **ExtTableException**, - in case any error take place during the update operation.
-
- *saveImage*

```
public OperationResult saveImage( java.lang.String  imgName, byte []
contents )
```
-
- *saveVatType*

```
public OperationResult saveVatType(
com.imagos.axilice.core.common.VatTypeOT  vatType )
```

31.2 Classes

31.2.1 CLASS BarcodeTypeOT

Store class to hold the data of a barcode type.
Responsibilities:

Store of barcode type data for exchanging data among presentation, business and data access layers.

.

DECLARATION

```
public class BarcodeTypeOT
extends com.imagos.axilice.core.common.BaseVO
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idBarcodeType
 - Internal id of the barcode type.
- private String description
 - Description of the barcode type.
- private Integer version
 - Version of the barcode type used by Hibernate for optimistic concurrency control.

CONSTRUCTORS

- *BarcodeTypeOT*
 public **BarcodeTypeOT**()
 - **Usage**
 - * Default BarcodeTypeOT constructor.

METHODS

- *equals*
 public boolean **equals**(java.lang.Long id)

- *equals*
 public boolean **equals**(java.lang.Object other)
 - **Usage**
 - * Implementation of the equals method provided by the serializable interface.

- *getDescription*
 public String **getDescription**()
 - **Usage**
 - * Gets the description of the barcode type.
 - **Returns** - String, description of the barcode type.

- *getEntityId*
 public String **getEntityId**()

- *getIdBarcodeType*
 public long **getIdBarcodeType**()
 - **Usage**
 - * Gets the identifier of the barcode type.
 - **Returns** - String, identifier of the barcode type.

- *getInternalId*
 public long **getInternalId**()

- *getInternalValue*
`public String getInternalValue()`

- *getRevision*
`public static String getRevision()`
 - **Usage**
 - * Returns tag Id assigned to CVS source file.

- *getVersion*
`public Integer getVersion()`
 - **Usage**
 - * Gets the version of the barcode type used by Hibernate for optimistic concurrency control.
 - **Returns** - Integer, version of the group.

- *hashCode*
`public int hashCode()`
 - **Usage**
 - * Implementation of the hashCode method provided by the serializable interface.

- *setDescription*
`public void setDescription(java.lang.String description)`
 - **Usage**
 - * Sets up the description of the barcode type.
 - **Parameters**
 - * **description** - String, description of the barcode type.

- *setIdBarcodeType*
`public void setIdBarcodeType(long idBarcodeType)`
 - **Usage**
 - * Sets up the identifier of the barcode type.
 - **Parameters**
 - * **idGroup** - String, identifier of the barcode type.

- *setVersion*
`public void setVersion(java.lang.Integer version)`
 - **Usage**
 - * Sets up the version of the barcode type used by Hibernate for optimistic concurrency control.
 - **Parameters**
 - * **version** - Integer, version of the barcode type.

- *toString*
`public String toString()`
 - **Usage**
 - * Implementation of the toString() method to show the data of the ExtFieldOT formatted.

METHODS INHERITED FROM CLASS com.imagos.axilice.core.common.BaseVO

(in 31.2.2, page 740)

- *equals*
public abstract boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *format*
public String format(java.lang.String fieldId, java.util.ArrayList IFields)
- *formatValue*
public String formatValue(java.lang.String fieldId)
- *getEntityId*
public abstract String getEntityId()
- *getInternalId*
public abstract long getInternalId()
- *getInternalValue*
public abstract String getInternalValue()
- *getVisibleFieldsCount*
public int getVisibleFieldsCount(java.util.ArrayList IFields)
 - **Usage**
 - * Calculates the number of visible fields of the entity.
 - **Returns** - int, visible fields number; 1 in case no visible fields have been defined.

31.2.2 CLASS BaseVO

DECLARATION

```
public abstract class BaseVO
extends java.lang.Object
```

CONSTRUCTORS

- *BaseVO*
public **BaseVO**()

METHODS

- *equals*
public abstract boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *format*
public String format(java.lang.String fieldId, java.util.ArrayList IFields)

- *formatValue*
public String formatValue(java.lang.String fieldId)
- *getEntityId*
public abstract String getEntityId()
- *getInternalId*
public abstract long getInternalId()
- *getInternalValue*
public abstract String getInternalValue()
- *getVisibleFieldsCount*
public int getVisibleFieldsCount(java.util.ArrayList lFields)
 - **Usage**
 - * Calculates the number of visible fields of the entity.
 - **Returns** - int, visible fields number; 1 in case no visible fields have been defined.

31.2.3 CLASS CountryOT

Store class to hold the data of a country.
Responsibilities:

Store of country data for exchanging data among presentation, business and data access layers.

.

DECLARATION

```
public class CountryOT
extends com.imagos.axilice.core.common.BaseVO
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idCountry
 - Internal id of the country.
- private String name
 -
- private FlagOT flag
 -
- private Integer version
 - Version of the country used by Hibernate for optimistic concurrency control.

CONSTRUCTORS

- *CountryOT*
`public CountryOT()`

METHODS

- *equals*
`public boolean equals(java.lang.Long id)`
- *equals*
`public boolean equals(java.lang.Object obj)`
- *getEntityId*
`public String getEntityId()`
- *getFlag*
`public FlagOT getFlag()`
- *getIdCountry*
`public long getIdCountry()`
- *getInternalId*
`public long getInternalId()`
- *getInternalValue*
`public String getInternalValue()`
- *getName*
`public String getName()`
- *getRevision*
`public static String getRevision()`
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
- *getVersion*
`public Integer getVersion()`
 - **Usage**
 - * Gets the version of the country used by Hibernate for optimistic concurrency control.
 - **Returns** - Integer, version of the group.
- *hashCode*
`public int hashCode()`
- *setFlag*
`public void setFlag(com.imagos.axilice.core.common.FlagOT flag)`
- *setIdCountry*
`public void setIdCountry(long idCountry)`

- *setName*
public void setName(java.lang.String name)
 - *setVersion*
public void setVersion(java.lang.Integer version)
 - **Usage**
 - * Sets up the version of the country used by Hibernate for optimistic concurrency control.
 - **Parameters**
 - * **version** - Integer, version of the group.
-
- *toString*
public String toString()
 - **Usage**
 - * Implementation of the toString() method to show the data of the CountryOT formatted.

METHODS INHERITED FROM CLASS com.imagos.axilice.core.common.BaseVO

(in 31.2.2, page 740)

- *equals*
public abstract boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *format*
public String format(java.lang.String fieldId, java.util.ArrayList lFields)
- *formatValue*
public String formatValue(java.lang.String fieldId)
- *getEntityId*
public abstract String getEntityId()
- *getInternalId*
public abstract long getInternalId()
- *getInternalValue*
public abstract String getInternalValue()
- *getVisibleFieldsCount*
public int getVisibleFieldsCount(java.util.ArrayList lFields)
 - **Usage**
 - * Calculates the number of visible fields of the entity.
 - **Returns** - int, visible fields number; 1 in case no visible fields have been defined.

31.2.4 CLASS CurrencyOT

DECLARATION

```
public class CurrencyOT
extends com.imagos.axilice.core.common.BaseVO
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idCurrency
—
- private String description
—
- private double conversionRate
—
- private Date lastUpdate
—
- private String codeISO
—
- private FlagOT flag
—
- private Integer version
—

CONSTRUCTORS

- *CurrencyOT*
public **CurrencyOT**()

METHODS

- *equals*
public boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *getCodeISO*
public String getCodeISO()
- *getConversionRate*
public double getConversionRate()
- *getDescription*
public String getDescription()
- *getEntityId*
public String getEntityId()
- *getFlag*
public FlagOT getFlag()

- *getIdCurrency*
public long getIdCurrency()
- *getInternalId*
public long getInternalId()
- *getInternalValue*
public String getInternalValue()
- *getLastUpdate*
public Date getLastUpdate()
- *getVersion*
public Integer getVersion()
- *hashCode*
public int hashCode()
- *setCodeISO*
public void setCodeISO(java.lang.String codeISO)
- *setConversionRate*
public void setConversionRate(double conversionRate)
- *setDescription*
public void setDescription(java.lang.String description)
- *setFlag*
public void setFlag(com.imagos.axilice.core.common.FlagOT flag)
- *setIdCurrency*
public void setIdCurrency(long idCurrency)
- *setLastUpdate*
public void setLastUpdate(java.util.Date lastUpdate)
- *setVersion*
public void setVersion(java.lang.Integer version)

METHODS INHERITED FROM CLASS com.imagos.axilice.core.common.BaseVO

(in 31.2.2, page 740)

- *equals*
public abstract boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *format*
public String format(java.lang.String fieldId, java.util.ArrayList lFields)
- *formatValue*
public String formatValue(java.lang.String fieldId)
- *getEntityId*
public abstract String getEntityId()
- *getInternalId*
public abstract long getInternalId()

- *getInternalValue*
`public abstract String getInternalValue()`
- *getVisibleFieldsCount*
`public int getVisibleFieldsCount(java.util.ArrayList IFields)`
 - **Usage**
 - * Calculates the number of visible fields of the entity.
 - **Returns** - int, visible fields number; 1 in case no visible fields have been defined.

31.2.5 CLASS DocumentTotal

DECLARATION

```
public class DocumentTotal
extends java.lang.Object
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private BigDecimal baseImponible
 —
- private BigDecimal dtoComercial
 —
- private BigDecimal dtoPP
 —
- private BigDecimal importeDtoComercial
 —
- private BigDecimal importeDtoPP
 —
- private int numLineas
 —
- private int numTiposIVA
 —
- private BigDecimal total
 —
- private DocumentOT document
 —
- private int resultadoCalculo

-
- private ArrayList desgloseIVA
-
- private ArrayList desgloseApicable
-
- private BigDecimal sumaImportes
-

CONSTRUCTORS

- *DocumentTotal*
public **DocumentTotal**(com.imagos.axilice.core.management.DocumentOT doc)

METHODS

- *calcular*
public int **calcular**()
- *getBaseImponible*
public BigDecimal **getBaseImponible**()
- *getDesgloseApicable*
public ArrayList **getDesgloseApicable**()
- *getDesgloseIVA*
public ArrayList **getDesgloseIVA**()
- *getDocument*
public DocumentOT **getDocument**()
- *getDtoComercial*
public BigDecimal **getDtoComercial**()
- *getDtoPP*
public BigDecimal **getDtoPP**()
- *getDtoPPAplicable*
public BigDecimal **getDtoPPAplicable**()
- *getImporteDtoComercial*
public BigDecimal **getImporteDtoComercial**()
- *getImporteDtoPP*
public BigDecimal **getImporteDtoPP**()
- *getImporteIva*
public BigDecimal **getImporteIva**()
- *getImporteRE*
public BigDecimal **getImporteRE**()

- *getNumLineas*
public int **getNumLineas**()
- *getNumTiposIVA*
public int **getNumTiposIVA**()
- *getResultadoCalculo*
public int **getResultadoCalculo**()
- *getSumaImportes*
public BigDecimal **getSumaImportes**()
- *getTotal*
public BigDecimal **getTotal**()
- *setBaseImponible*
public void **setBaseImponible**(java.math.BigDecimal baseImponible)
- *setDesgloseApicable*
public void **setDesgloseApicable**(java.util.ArrayList desgloseApicable)
- *setDesgloseIVA*
public void **setDesgloseIVA**(java.util.ArrayList desgloseIVA)
- *setDocument*
public void **setDocument**(com.imagos.axilice.core.management.DocumentOT document)
- *setDtoComercial*
public void **setDtoComercial**(java.math.BigDecimal dtoComercial)
- *setDtoPP*
public void **setDtoPP**(java.math.BigDecimal dtoPP)
- *setImporteDtoComercial*
public void **setImporteDtoComercial**(java.math.BigDecimal importeDtoComercial)
- *setImporteDtoPP*
public void **setImporteDtoPP**(java.math.BigDecimal importeDtoPP)
- *setNumLineas*
public void **setNumLineas**(int numLineas)
- *setNumTiposIVA*
public void **setNumTiposIVA**(int numTiposIVA)
- *setResultadoCalculo*
public void **setResultadoCalculo**(int resultadoCalculo)
- *setSumaImportes*
public void **setSumaImportes**(java.math.BigDecimal sumaImportes)
- *setTotal*
public void **setTotal**(java.math.BigDecimal total)

31.2.6 CLASS FlagOT

Store class to hold the data of a flag.
Responsibilities:

Store of flag data for exchanging data among presentation, business and data access layers.
.

DECLARATION

```
public class FlagOT
extends com.imagos.axilice.core.common.BaseVO
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idFlag
 - Internal id of the flag.
- private String name
 - Name of the flag.
- private ResourceOT image
 -
- private Integer version
 - Version of the flag used by Hibernate for optimistic concurrency control.

CONSTRUCTORS

- *FlagOT*
public **FlagOT**()

METHODS

- *equals*
public boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *getEntityId*
public String getEntityId()
- *getIdFlag*
public long getIdFlag()

- *getImage*
public ResourceOT getImage()
- *getInternalId*
public long getInternalId()
- *getInternalValue*
public String getInternalValue()
- *getName*
public String getName()
- *getVersion*
public Integer getVersion()
 - **Usage**
 - * Gets the version of the flag used by Hibernate for optimistic concurrency control.
 - **Returns** - Integer, version of the group.
- *hashCode*
public int hashCode()
- *setIdFlag*
public void setIdFlag(long idFlag)
- *setImage*
public void setImage(com.imagos.axilice.core.installer.ResourceOT image)
- *setName*
public void setName(java.lang.String name)
- *setVersion*
public void setVersion(java.lang.Integer version)
 - **Usage**
 - * Sets up the version of the flag used by Hibernate for optimistic concurrency control.
 - **Parameters**
 - * **version** - Integer, version of the group.
- *toString*
public String toString()
 - **Usage**
 - * Implementation of the toString() method to show the data of the FlagOT formatted.

METHODS INHERITED FROM CLASS com.imagos.axilice.core.common.BaseVO

(in 31.2.2, page 740)

- *equals*
public abstract boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *format*
public String format(java.lang.String fieldId, java.util.ArrayList IFields)
- *formatValue*
public String formatValue(java.lang.String fieldId)
- *getEntityId*
public abstract String getEntityId()
- *getInternalId*
public abstract long getInternalId()
- *getInternalValue*
public abstract String getInternalValue()
- *getVisibleFieldsCount*
public int **getVisibleFieldsCount**(java.util.ArrayList IFields)
 - **Usage**
 - * Calculates the number of visible fields of the entity.
 - **Returns** - int, visible fields number; 1 in case no visible fields have been defined.

31.2.7 CLASS ImageComparator

DECLARATION

```
public class ImageComparator
extends java.lang.Object
implements java.util.Comparator
```

CONSTRUCTORS

- *ImageComparator*
public **ImageComparator**()

METHODS

- *compare*
public int **compare**(java.lang.Object arg0, java.lang.Object arg1)

31.2.8 CLASS ImageOT

Store class to hold the data of an application image resource.
Responsibilities:

- Store of image data for exchanging data among presentation, business and data access layers.

DECLARATION

```
public class ImageOT
extends com.imagos.axilice.core.common.BaseVO
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idImage
 - Internal id of the image.
- private String path
 - Path of the image file.
- private Integer version
 - Version of the image used by Hibernate for optimistic concurrency control.
- private Image renderedImage
 - Instance of image rendered from the image file.
- private byte contents
 - Contents of the image file.
- private ImageData imgData
 -

CONSTRUCTORS

- *ImageOT*

```
public ImageOT( )
```

 - **Usage**
 - * Default ImageOT constructor.

METHODS

- *equals*
public boolean equals(java.lang.Long id)
 - *equals*
public boolean equals(java.lang.Object other)
 - **Usage**
 - * Implementation of the equals method provided by the serializable interface.
 - *getContents*
public byte getContents()
 - *getEntityId*
public String getEntityId()
 - *getIdImage*
public long getIdImage()
 - **Usage**
 - * Gets the identifier of the image.
 - **Returns** - long, identifier of the image.
 - *getImgData*
public ImageData getImgData()
 - *getInternalId*
public long getInternalId()
 - *getInternalValue*
public String getInternalValue()
 - *getPath*
public String getPath()
 - **Usage**
 - * Gets the path of the image.
 - **Returns** - String, description of the image.
 - *getRenderizedImage*
public Image getRenderizedImage()
 - **Usage**
 - * Gets the renderized image loaded from the image file.
 - **Returns** - Image, renderized image.
 - *getRevision*
public static String getRevision()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
-

- *getVersion*
`public Integer getVersion()`
 - **Usage**
 - * Gets the version of the image used by Hibernate for optimistic concurrency control.
 - **Returns** - Integer, version of the image.

- *hashCode*
`public int hashCode()`
 - **Usage**
 - * Implementation of the hashCode method provided by the serializable interface.

- *setContents*
`public void setContents(byte [] contents)`

- *setIdImage*
`public void setIdImage(long idImage)`
 - **Usage**
 - * Sets up the identifier of the image.
 - **Parameters**
 - * idImage - long, identifier of the image.

- *setImgData*
`public void setImgData(ImageData imgData)`

- *setPath*
`public void setPath(java.lang.String path)`
 - **Usage**
 - * Sets up the path of the image.
 - **Parameters**
 - * path - String, path of the image.

- *setRenderizedImage*
`public void setRenderizedImage(Image renderizedImage)`
 - **Usage**
 - * Sets up the renderized image loaded from the image file
 - **Parameters**
 - * renderizedImage - Image, renderized image.

- *setVersion*
`public void setVersion(java.lang.Integer version)`
 - **Usage**
 - * Sets up the version of the image used by Hibernate for optimistic concurrency control.
 - **Parameters**
 - * version - Integer, version of the image.

- *toString*
`public String toString()`
 - **Usage**
 - * Implementation of the toString() method to show the data of the ExtFieldOT formatted.

METHODS INHERITED FROM CLASS `com.imagos.axilice.core.common.BaseVO`

(in 31.2.2, page 740)

- *equals*
`public abstract boolean equals(java.lang.Long id)`
- *equals*
`public boolean equals(java.lang.Object obj)`
- *format*
`public String format(java.lang.String fieldId, java.util.ArrayList IFields)`
- *formatValue*
`public String formatValue(java.lang.String fieldId)`
- *getEntityId*
`public abstract String getEntityId()`
- *getInternalId*
`public abstract long getInternalId()`
- *getInternalValue*
`public abstract String getInternalValue()`
- *getVisibleFieldsCount*
`public int getVisibleFieldsCount(java.util.ArrayList IFields)`
 - **Usage**
 - * Calculates the number of visible fields of the entity.
 - **Returns** - int, visible fields number; 1 in case no visible fields have been defined.

31.2.9 CLASS LineaDesgloseIVA

DECLARATION

```
public class LineaDesgloseIVA
extends java.lang.Object
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private BigDecimal baseImponible
—
- private BigDecimal iva
—
- private BigDecimal RE
—

CONSTRUCTORS

- *LineaDesgloseIVA*
public **LineaDesgloseIVA**()

METHODS

- *getBaseImponible*
public BigDecimal **getBaseImponible**()
- *getImporteIva*
public BigDecimal **getImporteIva**()
- *getImporteRE*
public BigDecimal **getImporteRE**()
- *getIva*
public BigDecimal **getIva**()
- *getRE*
public BigDecimal **getRE**()
- *setBaseImponible*
public void **setBaseImponible**(java.math.BigDecimal baseImponible)
- *setIva*
public void **setIva**(java.math.BigDecimal iva)
- *setRE*
public void **setRE**(java.math.BigDecimal rE)

31.2.10 CLASS MultiSelectionSubentityOT

DECLARATION

```
public class MultiSelectionSubentityOT
extends com.imagos.axilice.core.installer.SubentityOT
```

CONSTRUCTORS

- *MultiSelectionSubentityOT*
public **MultiSelectionSubentityOT**(java.lang.String entityId,
java.lang.String subentityWidgetName,
com.imagos.axilice.core.installer.EntityLauncher parentEntityLauncher,
SwtWidget widget, java.lang.String collectionName, java.lang.String
parentFieldId)

METHODS

- *getCollection*
public Collection getCollection()
- *getCollectionName*
public String getCollectionName()
- *getPanel*
public SwtWidget getPanel()
- *getPanelWidget*
public Composite getPanelWidget()
- *getSubentityLauncher*
public EntityLauncher getSubentityLauncher()
- *getSubentityWidgetName*
public String getSubentityWidgetName()
- *setCollection*
public void setCollection(java.util.Collection collection)
- *setCollectionName*
public void setCollectionName(java.lang.String collectionName)
- *setPanel*
public void setPanel(SwtWidget panel)
- *setSubentityLauncher*
public void setSubentityLauncher(com.imagos.axilice.core.installer.EntityLauncher subentityLauncher)
- *setSubentityWidgetName*
public void setSubentityWidgetName(java.lang.String subentityWidgetName)
- *toString*
public String toString()

METHODS INHERITED FROM CLASS com.imagos.axilice.core.installer.SubentityOT

(in 24.2.12, page 563)

- *getParentEntity*
public EntityLauncher getParentEntity()
 - **Usage**
 - * Gets the parent entity launcher that will manage the events of the subentity selection process.
 - **Returns** - EntityLauncher, parent entity launcher.
- *getParentFieldId*
public String getParentFieldId()
 - **Usage**
 - * Gets the parent entity field whose value is the selected subentity.

- **Returns** - String, parent entity field id.
- - *getRevision*
`public static String getRevision()`
 - **Usage**
 * Returns tag Id assigned to CVS source file.
- - *getSubentityData*
`public EntityOT getSubentityData()`
 - **Usage**
 * Gets the subentity data.
 - **Returns** - EntityOT, subentity data.
- - *setParentEntity*
`public void setParentEntity(com.imagos.axilice.core.installer.EntityLauncher parentEntity)`
 - **Usage**
 * Sets up the parent entity launcher that will manage the events of the subentity selection process.
 - **Parameters**
 * `parentEntity` - EntityLauncher, parent entity launcher.
- - *setParentFieldId*
`public void setParentFieldId(java.lang.String parentFieldId)`
 - **Usage**
 * Sets up the parent entity field whose value is the selected subentity.
 - **Parameters**
 * `parentFieldId` - String, parent entity field id.
- - *setSubentityData*
`public void setSubentityData(com.imagos.axilice.core.installer.EntityOT subentityData)`
 - **Usage**
 * Sets up the subentity data.
 - **Parameters**
 * `subentityData` - EntityOT, subentity data.

31.2.11 CLASS PriceListLineOT

DECLARATION

```
public class PriceListLineOT
extends com.imagos.axilice.core.common.BaseVO
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idPriceListLine
—
- private PriceListOT priceList
—
- private ProductOT product
—
- private BigDecimal price
—
- private double discount
—
- private Integer version
—

CONSTRUCTORS

- *PriceListLineOT*
public **PriceListLineOT**()

METHODS

- *equals*
public boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *getDiscount*
public double getDiscount()
- *getEntityId*
public String getEntityId()
- *getIdPriceListLine*
public long getIdPriceListLine()
- *getInternalId*
public long getInternalId()
- *getInternalValue*
public String getInternalValue()
- *getPrice*
public BigDecimal getPrice()

- *getPriceList*
public PriceListOT getPriceList()
- *getPriceListName*
public String getPriceListName()
- *getProduct*
public ProductOT getProduct()
- *getProductName*
public String getProductName()
- *getVersion*
public Integer getVersion()
- *hashCode*
public int hashCode()
- *setDisccount*
public void setDisccount(double disccount)
- *setIdPriceListLine*
public void setIdPriceListLine(long idPriceListLine)
- *setPrice*
public void setPrice(java.math.BigDecimal price)
- *setPriceList*
public void setPriceList(com.imagos.axilice.core.common.PriceListOT priceList)
- *setProduct*
public void setProduct(com.imagos.axilice.core.warehouse.ProductOT product)
- *setVersion*
public void setVersion(java.lang.Integer version)
- *toString*
public String toString()

METHODS INHERITED FROM CLASS com.imagos.axilice.core.common.BaseVO

(in 31.2.2, page 740)

- *equals*
public abstract boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *format*
public String format(java.lang.String fieldId, java.util.ArrayList lFields)
- *formatValue*
public String formatValue(java.lang.String fieldId)
- *getEntityId*
public abstract String getEntityId()

- *getInternalId*
public abstract long **getInternalId**()
- *getInternalValue*
public abstract String **getInternalValue**()
- *getVisibleFieldsCount*
public int **getVisibleFieldsCount**(java.util.ArrayList IFields)
 - **Usage**
 - * Calculates the number of visible fields of the entity.
 - **Returns** - int, visible fields number; 1 in case no visible fields have been defined.

31.2.12 CLASS PriceListOT

Store class to hold the data of a price list.
Responsibilities:

Store of flag data for exchanging data among presentation, business and data access layers.
.

DECLARATION

```
public class PriceListOT
extends com.imagos.axilice.core.common.BaseVO
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idPriceList
–
- private Integer version
–
- private String description
–
- private Date fromDate
–
- private Date toDate
–
- private Set lines
–
- private boolean taxesInc
–

CONSTRUCTORS

- *PriceListOT*
public PriceListOT()

METHODS

- *equals*
public boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *getDescription*
public String getDescription()
- *getEntityId*
public String getEntityId()
- *getFromDate*
public Date getFromDate()
- *getIdPriceList*
public long getIdPriceList()
- *getInternalId*
public long getInternalId()
- *getInternalValue*
public String getInternalValue()
- *getLines*
public Set getLines()
- *getToDate*
public Date getToDate()
- *getVersion*
public Integer getVersion()
- *hashCode*
public int hashCode()
- *isTaxesInc*
public boolean isTaxesInc()
- *setDescription*
public void setDescription(java.lang.String description)
- *setFromDate*
public void setFromDate(java.util.Date fromDate)
- *setIdPriceList*
public void setIdPriceList(long idPriceList)

- *setLines*
public void setLines(java.util.Set lines)
- *setTaxesInc*
public void setTaxesInc(boolean taxesInc)
- *setToDate*
public void setToDate(java.util.Date toDate)
- *setVersion*
public void setVersion(java.lang.Integer version)
- *toString*
public String toString()

METHODS INHERITED FROM CLASS com.imagos.axilice.core.common.BaseVO

(in 31.2.2, page 740)

- *equals*
public abstract boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *format*
public String format(java.lang.String fieldId, java.util.ArrayList IFields)
- *formatValue*
public String formatValue(java.lang.String fieldId)
- *getEntityId*
public abstract String getEntityId()
- *getInternalId*
public abstract long getInternalId()
- *getInternalValue*
public abstract String getInternalValue()
- *getVisibleFieldsCount*
public int getVisibleFieldsCount(java.util.ArrayList IFields)

– **Usage**

* Calculates the number of visible fields of the entity.

– **Returns** - int, visible fields number; 1 in case no visible fields have been defined.

31.2.13 CLASS RateOT

DECLARATION

```
public class RateOT
extends com.imagos.axilice.core.common.BaseVO
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idRate
—
- private String name
—
- private float percentIncrease
—
- private float linearIncrease
—
- private Integer version
—

CONSTRUCTORS

- *RateOT*
public **RateOT**()

METHODS

- *equals*
public boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *getEntityId*
public String getEntityId()
- *getIdRate*
public long getIdRate()
- *getInternalId*
public long getInternalId()
- *getInternalValue*
public String getInternalValue()
- *getLinearIncrease*
public float getLinearIncrease()
- *getName*
public String getName()
- *getPercentIncrease*
public float getPercentIncrease()

- *getVersion*
public long getVersion()
- *hashCode*
public int hashCode()
- *setIdRate*
public void setIdRate(long idRate)
- *setLinearIncrease*
public void setLinearIncrease(float linearIncrease)
- *setName*
public void setName(java.lang.String name)
- *setPercentIncrease*
public void setPercentIncrease(float percentIncrease)
- *setVersion*
public void setVersion(java.lang.Integer version)

METHODS INHERITED FROM CLASS com.imagos.axilice.core.common.BaseVO

(in 31.2.2, page 740)

- *equals*
public abstract boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *format*
public String format(java.lang.String fieldId, java.util.ArrayList IFields)
- *formatValue*
public String formatValue(java.lang.String fieldId)
- *getEntityId*
public abstract String getEntityId()
- *getInternalId*
public abstract long getInternalId()
- *getInternalValue*
public abstract String getInternalValue()
- *getVisibleFieldsCount*
public int getVisibleFieldsCount(java.util.ArrayList IFields)
 - **Usage**
 - * Calculates the number of visible fields of the entity.
 - **Returns** - int, visible fields number; 1 in case no visible fields have been defined.

31.2.14 CLASS RegistryOT

DECLARATION

```
public class RegistryOT
extends java.lang.Object
```

CONSTRUCTORS

- *RegistryOT*
public RegistryOT()
- *RegistryOT*
public RegistryOT(long id, java.lang.String fieldId, java.lang.String [] lValues)

METHODS

- *getColumnCount*
public int getColumnCount()
- *getFieldId*
public String getFieldId()
- *getId*
public long getId()
- *getLValues*
public String getLValues()
- *getValue*
public String getValue(int col)
- *getValues*
public String getValues()
- *setFieldId*
public void setFieldId(java.lang.String fieldId)
- *setId*
public void setId(long id)
- *setLValues*
public void setLValues(java.lang.String [] values)

31.2.15 CLASS SelectionSubentityOT

DECLARATION

```
public class SelectionSubentityOT
extends com.imagos.axilice.core.installer.SubentityOT
```

CONSTRUCTORS

- *SelectionSubentityOT*
public SelectionSubentityOT(java.lang.String entityId, java.lang.String subentityWidgetName, com.imagos.axilice.core.installer.EntityLauncher parentEntityLauncher, Button widget)

METHODS

- *getButton*
public Button **getButton**()
- *getSubentityLauncher*
public EntityLauncher **getSubentityLauncher**()
- *setButton*
public void **setButton**(Button button)
- *setSubentityLauncher*
public void **setSubentityLauncher**(
com.imagos.axilice.core.installer.EntityLauncher subentityLauncher)
- *toString*
public String **toString**()

METHODS INHERITED FROM CLASS com.imagos.axilice.core.installer.SubentityOT

(in 24.2.12, page 563)

- *getParentEntity*
public EntityLauncher **getParentEntity**()
 - **Usage**
* Gets the parent entity launcher that will manage the events of the subentity selection process.
 - **Returns** - EntityLauncher, parent entity launcher.
- *getParentFieldId*
public String **getParentFieldId**()
 - **Usage**
* Gets the parent entity field whose value is the selected subentity.
 - **Returns** - String, parent entity field id.
- *getRevision*
public static String **getRevision**()
 - **Usage**
* Returns tag Id assigned to CVS source file.
- *getSubentityData*
public EntityOT **getSubentityData**()
 - **Usage**
* Gets the subentiy data.
 - **Returns** - EntityOT, subentity data.
- *setParentEntity*
public void **setParentEntity**(com.imagos.axilice.core.installer.EntityLauncher
parentEntity)
 - **Usage**
* Sets up the parent entity launcher that will manage the events of the subentity selection process.
 - **Parameters**

- * **parentEntity** - EntityLauncher, parent entity launcher.
- - *setParentFieldId*
public void setParentFieldId(java.lang.String parentFieldId)
 - **Usage**
 - * Sets up the parent entity field whose value is the selected subentity.
 - **Parameters**
 - * **parentFieldId** - String, parent entity field id.

- *setSubentityData*
public void setSubentityData(com.imagos.axilice.core.installer.EntityOT subentityData)
 - **Usage**
 - * Sets up the subentity data.
 - **Parameters**
 - * **subentityData** - EntityOT, subentity data.

31.2.16 CLASS StateOT

Store class to hold the data of a province.
Responsibilities:

Store of province data for exchanging data among presentation, business and data access layers.
 .

DECLARATION

```
public class StateOT
extends com.imagos.axilice.core.common.BaseVO
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idState
 - Internal id of the province.
- private String name
 - Name of the province.
- private CountryOT country
 -
- private Integer version
 - Version of the province used by Hibernate for optimistic concurrency control.

CONSTRUCTORS

- *StateOT*
public **StateOT**()

METHODS

- *equals*
public boolean **equals**(java.lang.Long id)
 - *equals*
public boolean **equals**(java.lang.Object other)
 - **Usage**
 - * Implementation of the equals method provided by the serializable interface.
 - *getCountry*
public CountryOT **getCountry**()
 - *getEntityId*
public String **getEntityId**()
 - *getIdState*
public long **getIdState**()
 - *getInternalId*
public long **getInternalId**()
 - *getInternalValue*
public String **getInternalValue**()
 - *getName*
public String **getName**()
 - *getVersion*
public Integer **getVersion**()
 - **Usage**
 - * Gets the version of the province used by Hibernate for optimistic concurrency control.
 - **Returns** - Integer, version of the group.
 - *hashCode*
public int **hashCode**()
 - **Usage**
 - * Implementation of the hashCode method provided by the serializable interface.
 - *setCountry*
public void **setCountry**(com.imagos.axilice.core.common.CountryOT country)
-

- *setIdState*
public void setIdState(long idState)
 - *setName*
public void setName(java.lang.String name)
 - *setVersion*
public void setVersion(java.lang.Integer version)
 - **Usage**
 - * Sets up the version of the province used by Hibernate for optimistic concurrency control.
 - **Parameters**
 - * **version** - Integer, version of the group.
-
- *toString*
public String toString()
 - **Usage**
 - * Implementation of the toString() method to show the data of the StateOT formatted.

METHODS INHERITED FROM CLASS com.imagos.axilice.core.common.BaseVO

(in 31.2.2, page 740)

- *equals*
public abstract boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object obj)
- *format*
public String format(java.lang.String fieldId, java.util.ArrayList lFields)
- *formatValue*
public String formatValue(java.lang.String fieldId)
- *getEntityId*
public abstract String getEntityId()
- *getInternalId*
public abstract long getInternalId()
- *getInternalValue*
public abstract String getInternalValue()
- *getVisibleFieldsCount*
public int getVisibleFieldsCount(java.util.ArrayList lFields)
 - **Usage**
 - * Calculates the number of visible fields of the entity.
 - **Returns** - int, visible fields number; 1 in case no visible fields have been defined.

31.2.17 CLASS ValueOT

This class implements a generic container for storing the data of a pair key-value, suitable for filling a combobox. It implements the value object pattern.
Responsibilities:

- Store key-value pairs.

DECLARATION

```
public class ValueOT
extends java.lang.Object
```

CONSTRUCTORS

- *ValueOT*
public **ValueOT**()

METHODS

- *getKey*
public long **getKey**()
 - **Usage**
 - * Gets the internal identifier (key) of the pair key-value.
 - **Returns** - long, key of the element.
- *getKeyfieldName*
public String **getKeyfieldName**()
 - **Usage**
 - * Gets the name of the internal identifier (key) field.
 - **Returns** - String, name of the internal identifier field.
- *getRevision*
public static String **getRevision**()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.
- *getValfieldName*
public String **getValfieldName**()
 - **Usage**
 - * Gets the name of the value field.
 - **Returns** - String, name of the value field.
- *getValue*
public String **getValue**()
 - **Usage**
 - * Gets the value of the pair key-value.
 - **Returns** - String, value of the element.
- *setKey*
public void **setKey**(long key)

- **Usage**

- * Sets up the internal identifier (key) of the pair key-value.

- **Parameters**

- * **key** - long, key of the element.

- *setKeyfieldName*

```
public void setKeyfieldName( java.lang.String  keyfieldName )
```

- **Usage**

- * Sets up the name of the internal identifier field.

- **Parameters**

- * **keyfieldName** - String, name of the internal identifier field.

- *setValfieldName*

```
public void setValfieldName( java.lang.String  valfieldName )
```

- **Usage**

- * Sets up the name of the value field.

- **Parameters**

- * **valfieldName** - String, name of the value field.

- *setValue*

```
public void setValue( java.lang.String  value )
```

- **Usage**

- * Sets up the value of the pair key-value.

- **Parameters**

- * **value** - String, value of the element.

31.2.18 CLASS VatTypeOT

Store class to hold the data of a VAT type.
Responsibilities:

Store of VAT type data for exchanging data among presentation, business and data access layers.

.

DECLARATION

```
public class VatTypeOT
extends com.imagos.axilice.core.common.BaseVO
implements java.io.Serializable
```

SERIALIZABLE FIELDS

- private long idVat
 - Internal id of the VAT type.
- private String description
 - Description of the VAT type.
- private BigDecimal taxPercentage
 - Tax percentage of the VAT type.
- private BigDecimal recEquiPercentage
 - Rec equi percentage of the VAT type.
- private Integer version
 - Version of the VAT type used by Hibernate for optimistic concurrency control.

CONSTRUCTORS

- *VatTypeOT*
public VatTypeOT()
 - **Usage**
 - * Default VatTypeOT constructor.

METHODS

- *equals*
public boolean equals(java.lang.Long id)
- *equals*
public boolean equals(java.lang.Object other)
 - **Usage**
 - * Implementation of the equals method provided by the serializable interface.
- *getDescription*
public String getDescription()
 - **Usage**
 - * Gets the description of the VAT type.
 - **Returns** - String, description of the VAT type.
- *getEntityId*
public String getEntityId()
- *getIdVat*
public long getIdVat()

- **Usage**
 - * Gets the identifier of the VAT type.
 - **Returns** - String, identifier of the VAT type.

- *getInternalId*
public long getInternalId()

- *getInternalValue*
public String getInternalValue()

- *getRecEquiPercentage*
public BigDecimal getRecEquiPercentage()
 - **Usage**
 - * Gets the rec. equi. percentage of the VAT type.
 - **Returns** - BigDecimal, rec. equi. percentage of the VAT type.

- *getRevision*
public static String getRevision()
 - **Usage**
 - * Returns tag Id assigned to CVS source file.

- *getTaxPercentage*
public BigDecimal getTaxPercentage()
 - **Usage**
 - * Gets the tax percentage of the VAT type.
 - **Returns** - BigDecimal, tax percentage of the VAT type.

- *getVersion*
public Integer getVersion()
 - **Usage**
 - * Gets the version of the VAT type used by Hibernate for optimistic concurrency control.
 - **Returns** - Integer, version of the group.

- *hashCode*
public int hashCode()
 - **Usage**
 - * Implementation of the hashCode method provided by the serializable interface.

- *setDescription*
public void setDescription(java.lang.String description)
 - **Usage**
 - * Sets up the description of the VAT type.
 - **Parameters**
 - * **description** - String, description of the VAT type.

- *setIdVat*
 public void **setIdVat**(long idVat)
 – **Usage**
 * Sets up the identifier of the VAT type.
 – **Parameters**
 * idGroup - String, identifier of the VAT type.

- *setRecEquiPercentage*
 public void **setRecEquiPercentage**(java.math.BigDecimal recEquiPercentage)
 – **Usage**
 * Sets up the rec. equi. percentage of the VAT type.
 – **Parameters**
 * recEquiPercentage - BigDecimal, rec. equi. percentage of the VAT type.

- *setTaxPercentage*
 public void **setTaxPercentage**(java.math.BigDecimal taxPercentage)
 – **Usage**
 * Sets up the tax percentage of the VAT type.
 – **Parameters**
 * taxPercentage - BigDecimal, tax percentage of the VAT type.

- *setVersion*
 public void **setVersion**(java.lang.Integer version)
 – **Usage**
 * Sets up the version of the VAT type used by Hibernate for optimistic concurrency control.
 – **Parameters**
 * version - Integer, version of the VAT type.

- *toString*
 public String **toString**()
 – **Usage**
 * Implementation of the toString() method to show the data of the ExtFieldOT formatted.

METHODS INHERITED FROM CLASS com.imagos.axilice.core.common.BaseVO

(in 31.2.2, page 740)

- *equals*
 public abstract boolean **equals**(java.lang.Long id)

- *equals*
 public boolean **equals**(java.lang.Object obj)

- *format*
 public String **format**(java.lang.String fieldId, java.util.ArrayList lFields)

- *formatValue*
public String formatValue(java.lang.String fieldId)
- *getEntityId*
public abstract String getEntityId()
- *getInternalId*
public abstract long getInternalId()
- *getInternalValue*
public abstract String getInternalValue()
- *getVisibleFieldsCount*
public int **getVisibleFieldsCount**(java.util.ArrayList IFields)
 - **Usage**
 - * Calculates the number of visible fields of the entity.
 - **Returns** - int, visible fields number; 1 in case no visible fields have been defined.

Chapter 32

Package

com.imagos.axilice.server.warehouse.dao

Package Contents

Page

Classes

MeasureUnitDAO	778
<i>This class implements the measure unit data access object.</i>	
ProductDAO	780
<i>This class implements the product data access object.</i>	
ProductGroupDAO	782
<i>This class implements the product group data access object.</i>	

32.1 Classes

32.1.1 CLASS MeasureUnitDAO

This class implements the measure unit data access object. It's a part of the data access layer and gives access to the database data. It knows the database structure and implementation.
This class is implemented to access a PostgreSQL database implementation.
Responsibilities:

Provide statements for inserting new measure units into the database.

Provide statements for updating measure units into the database.

Provide statements for deleting measure units from the database.

Provide the database queries needed for measure units operations.

DECLARATION

```
public class MeasureUnitDAO
extends java.lang.Object
```

CONSTRUCTORS

- *MeasureUnitDAO*
public MeasureUnitDAO()
 - **Usage**
 - * Default constructor of the ProductDAO class.

METHODS

- *delMeasureUnit*
public OperationResult delMeasureUnit(java.lang.Long idMeasureUnit)

- *delMeasureUnits*
public OperationResult delMeasureUnits(java.util.ArrayList lSelGroups)

- *getAllMeasureUnits*
public List getAllMeasureUnits()
 - **Usage**
 - * Query the database to get all the available measure units.
 - **Returns** - ArrayList, list of all the measure units.
 - **Exceptions**
 - * **Exception**, - in case any error take place during the query operation.

- *getMeasureUnit*

```
public MeasureUnitOT getMeasureUnit( long idMeasureUnit )
```

- **Usage**

- * Query (by primary key) the database to get a measure unit data.

- **Parameters**

- * idMeasureUnit - long, Internal id of the measure unit.

- **Returns** - MeasureUnitOT, measure unit if it exists, null in any other case.

- **Exceptions**

- * AxiliceException, - in case any error take place during the query operation.

- *getMeasureUnit*

```
public MeasureUnitOT getMeasureUnit( java.lang.String description )
```

- **Usage**

- * Query the database to get a measure unit data.

- **Parameters**

- * description - String, measure unit description.

- **Returns** - MeasureUnitOT, measure unit if it exists, null in any other case.

- **Exceptions**

- * AxiliceException, - in case any error take place during the query operation.

- *getMeasureUnits*

```
public ArrayList getMeasureUnits( java.util.ArrayList lCriteria )
```

- **Usage**

- * Query the database to get a list of measure units that math the query criteria specified.

- **Parameters**

- * lCriteria - ArrayList<CriteriaBean>, list of query criteria.

- **Returns** - ArrayList<MeasureUnitOT>, list of measure units matching the searching criteria.

- **Exceptions**

- * AxiliceException, - in case any error take place during the query operation.

- *getRevision*

```
public static String getRevision( )
```

- **Usage**

- * Returns tag Id assigned to CVS source file.

- *saveMeasureUnit*

```
public OperationResult saveMeasureUnit(
com.imagos.axilice.core.warehouse.MeasureUnitOT group )
```

32.1.2 CLASS ProductDAO

This class implements the product data access object. It's a part of the data access layer and gives access to the database data. It knows the database structure and implementation.
This class is implemented to access a PostgreSQL database implementation.
Responsibilities:

Provide statements for inserting new products into the database.

Provide statements for updating products into the database.

Provide statements for deleting products from the database.

Provide the database queries needed for products operations.

DECLARATION

```
public class ProductDAO
extends java.lang.Object
```

CONSTRUCTORS

- *ProductDAO*
public ProductDAO()
 - **Usage**
 - * Default constructor of the ProductDAO class.

METHODS

- *delProduct*
public OperationResult delProduct(java.lang.Long idProduct)
 - **Usage**
 - * Updates the database removing the selected product.
 - **Parameters**
 - * **idProduct** - Long, internal id of the selected product.
 - **Returns** - OperationResult, result of the operation and list of messages generated through the operation.
 - **Exceptions**
 - * **AxiliceException**, - in case any error take place during the query operation.
- *delProducts*
public OperationResult delProducts(java.util.ArrayList lSelProducts)
- *getAllProducts*
public List getAllProducts()

- **Usage**
 - * Query the database to get all the available products.
 - **Returns** - ArrayList, list of all the products.
 - **Exceptions**
 - * **Exception**, - in case any error take place during the query operation.
-
- *getProduct*

```
public ProductOT getProduct( long idProduct )
```

 - **Usage**
 - * Query (by primary key) the database to get a product data.
 - **Parameters**
 - * **idProduct** - long, Internal id of the product.
 - **Returns** - ProductOT, product if it exists, null in any other case.
 - **Exceptions**
 - * **AxiliceException**, - in case any error take place during the query operation.
-
- *getProduct*

```
public ProductOT getProduct( java.lang.String reference )
```

 - **Usage**
 - * Query the database to get a product data.
 - **Parameters**
 - * **reference** - String, product reference.
 - **Returns** - ProductOT, product if it exists, null in any other case.
 - **Exceptions**
 - * **AxiliceException**, - in case any error take place during the query operation.
-
- *getProducts*

```
public ArrayList getProducts( java.util.ArrayList lCriteria )
```

 - **Usage**
 - * Query the database to get a list of products that math the query criteria specified.
 - **Parameters**
 - * **lCriteria** - ArrayList<CriteriaBean>, list of query criteria.
 - **Returns** - ArrayList<ProductOT>, list of products matching the searching criteria.
 - **Exceptions**
 - * **AxiliceException**, - in case any error take place during the query operation.
-
- *getRevision*

```
public static String getRevision( )
```

 - **Usage**
 - * Returns tag Id assigned to CVS source file.
-
- *saveProduct*

```
public OperationResult saveProduct(
com.imagos.axilice.core.warehouse.ProductOT product )
```

 - **Usage**

- * Updates the database with the data of the edited entity registry. It's responsible of saving new entity registries (add a new registry) as well as of updating the contents of existing entity registries.
- **Parameters**
 - * **product** - ProductOT, product instance with the new data.
- **Returns** - OperationResult, result of the operation and list of messages generated through the operation.
- **Exceptions**
 - * **AxiliceException**, - in case any error take place during the query operation.

32.1.3 CLASS ProductGroupDAO

This class implements the product group data access object. It's a part of the data access layer and gives access to the database data. It knows the database structure and implementation.
This class is implemented to access a PostgreSQL database implementation.
Responsibilities:

Provide statements for inserting new product groups into the database.

Provide statements for updating product groups into the database.

Provide statements for deleting product groups from the database.

Provide the database queries needed for product groups operations.

DECLARATION

```
public class ProductGroupDAO
extends java.lang.Object
```

CONSTRUCTORS

- *ProductGroupDAO*
public ProductGroupDAO()
 - **Usage**
 - * Default constructor of the ProductDAO class.

METHODS

- *delProductGroup*
public OperationResult delProductGroup(java.lang.Long idGroup)
 - *delProductGroups*
public OperationResult delProductGroups(java.util.ArrayList lSelGroups)
-

- *getAllProductGroups*

```
public List getAllProductGroups( )
```

- **Usage**

- * Query the database to get all the available product groups.

- **Returns** - ArrayList, list of all the product groups.

- **Exceptions**

- * **Exception**, - in case any error take place during the query operation.

- *getNextProductId*

```
public long getNextProductId( )
```

- *getProductGroup*

```
public ProductGroupOT getProductGroup( long idGroup )
```

- **Usage**

- * Query (by primary key) the database to get a product group data.

- **Parameters**

- * **idGroup** - long, Internal id of the product group.

- **Returns** - ProductGroupOT, product group if it exists, null in any other case.

- **Exceptions**

- * **AxiliceException**, - in case any error take place during the query operation.

- *getProductGroup*

```
public ProductGroupOT getProductGroup( java.lang.String description )
```

- **Usage**

- * Query the database to get a product group data.

- **Parameters**

- * **description** - String, product group description.

- **Returns** - ProductGroupOT, product group if it exists, null in any other case.

- **Exceptions**

- * **AxiliceException**, - in case any error take place during the query operation.

- *getProductGroups*

```
public ArrayList getProductGroups( java.util.ArrayList lCriteria )
```

- **Usage**

- * Query the database to get a list of product groups that math the query criteria specified.

- **Parameters**

- * **lCriteria** - ArrayList<CriteriaBean>, list of query criteria.

- **Returns** - ArrayList<ProductGroupOT>, list of product groups matching the searching criteria.

- **Exceptions**

- * **AxiliceException**, - in case any error take place during the query operation.

- *getRevision*

```
public static String getRevision( )
```

- **Usage**

* Returns tag Id assigned to CVS source file.

- *resetNextProductId*

```
public void resetNextProductId( long   idGroup )
```

- *saveProductGroup*

```
public OperationResult saveProductGroup(  
com.imagos.axilice.core.warehouse.ProductGroupOT  group )
```


Chapter 33

Package com.imagos.axilice.client.traceability

<i>Package Contents</i>	<i>Page</i>
<hr/>	
Classes	
TraceabilityTreeViewno description...	786
<hr/>	

33.1 Classes

33.1.1 CLASS TraceabilityTreeView

DECLARATION

```
public class TraceabilityTreeView
extends java.lang.Object
```

CONSTRUCTORS

- *TraceabilityTreeView*
public **TraceabilityTreeView**()

METHODS

- *parseTree*
public TraceabilityTree **parseTree**(java.lang.String filename)

Chapter 34

Package com.imagos.axilice.core.traceability

<i>Package Contents</i>	<i>Page</i>
<hr/>	
Interfaces	
Tracer <i>...no description...</i>	788
Classes	
TracerException <i>This class encapsulates and holds all the errors that might have taken place during any of the application tracer operations.</i>	788
<hr/>	

34.1 Interfaces

34.1.1 INTERFACE Tracer

DECLARATION

```
public interface Tracer
implements java.rmi.Remote
```

METHODS

- *buildTraceabilityTree*
public TraceabilityTree buildTraceabilityTree(byte [] xmlContent)

34.2 Classes

34.2.1 CLASS TracerException

This class encapsulates and holds all the errors that might have taken place during any of the application tracer operations.
Responsibilities:

- Encapsulate and pass on the error messages to the upper layers.

DECLARATION

```
public class TracerException
extends java.lang.Exception
```

SERIALIZABLE FIELDS

- private int errCode
 - Internal error code to support error localization and classification.
- private String errorMessage
 - Error message string that is showed to the end user.

CONSTRUCTORS

- *TracerException*
public TracerException(int errCode)
 - Usage

- * Default TracerException constructor.
 - **Parameters**
 - * **errCode** - int, error code.
-
- *TracerException*

```
public TracerException( int    errCode, java.lang.Throwable  cause )
```

 - **Usage**
 - * TracerException constructor for assigning the exception that raised the error.
 - **Parameters**
 - * **errCode** - int, error code.
 - * **cause** - Throwable, exception that raised the error.

METHODS

- *getErrCode*

```
public int getErrCode( )
```

 - **Usage**
 - * Gets the error code returned to the user.
 - **Returns** - int, error code.
-
- *getErrorMessage*

```
public String getErrorMessage( )
```

 - **Usage**
 - * Gets the error label that will be used for constructing the formatted error message. This label will be the base of the message, constructed also with the parameters specified and using the internationalization mechanism to translate the message to the user's language if it's supported.
 - **Returns** - String, error label.
-
- *getMessage*

```
public String getMessage( )
```

 - **Usage**
 - * Gets the formatted error message text as it's showed on screen to the user.
 - **Returns** - int, error code.
-
- *getRevision*

```
public static String getRevision( )
```

 - **Usage**
 - * Returns tag Id assigned to CVS source file.
-
- *setErrCode*

```
public void setErrCode( int    errCode )
```

 - **Usage**
 - * Sets up the error code returned to the user.

– **Parameters**

* **errCode** - int, error code.

• *setErrorMessage*

public void setErrorMessage(java.lang.String **errorMessage)**

– **Usage**

* Sets up the error label that will be used for constructing the formatted error message. This label will be the base of the message, constructed also with the parameters specified and using the internationalization mechanism to translate the message to the user's language if it's supported.

– **Parameters**

* **errorMessage** - String, error label.

• *toString*

public String toString()

METHODS INHERITED FROM CLASS `java.lang.Exception`

METHODS INHERITED FROM CLASS `java.lang.Throwable`

• *fillInStackTrace*

public synchronized native Throwable fillInStackTrace()

• *getCause*

public Throwable getCause()

• *getLocalizedMessage*

public String getLocalizedMessage()

• *getMessage*

public String getMessage()

• *getStackTrace*

public StackTraceElement getStackTrace()

• *initCause*

public synchronized Throwable initCause(java.lang.Throwable **arg0)**

• *printStackTrace*

public void printStackTrace()

• *printStackTrace*

public void printStackTrace(java.io.PrintStream **arg0)**

• *printStackTrace*

public void printStackTrace(java.io.PrintWriter **arg0)**

• *setStackTrace*

public void setStackTrace(java.lang.StackTraceElement [] **arg0)**

• *toString*

public String toString()

Chapter 35

Package com.imagos.axilice.client.installer

<i>Package Contents</i>	<i>Page</i>
<hr/>	
Classes	
ClientClassLoader 792	
<i>...no description...</i>	
ClientClasspathUpdater 794	
<i>This class extends the local classpath of the Java virtual machine to hold the new or updated installed modules.</i>	
ClientInstaller 796	
<i>This class implements the client manager of the installing process.</i>	
<hr/>	

35.1 Classes

35.1.1 CLASS ClientClassLoader

DECLARATION

```
public class ClientClassLoader
extends java.net.URLClassLoader
```

CONSTRUCTORS

- *ClientClassLoader*
 public **ClientClassLoader**(java.lang.ClassLoader parent)
- *ClientClassLoader*
 public **ClientClassLoader**(java.lang.String classpath, java.lang.ClassLoader parent)
- *ClientClassLoader*
 public **ClientClassLoader**(java.net.URL [] lUrls)
 - **Usage**
 - * Constructor
 - **Parameters**
 - * lUrls - URL[], list of urls of the jars that must be
 - * parent - ClassLoader, parent ClassLoader in which all the classes' resolutions are delegated.
- *ClientClassLoader*
 public **ClientClassLoader**(java.net.URL [] lUrls, java.lang.ClassLoader parent)
 - **Usage**
 - * Constructor
 - **Parameters**
 - * lUrls - URL[], list of urls of the jars that must be
 - * parent - ClassLoader, parent ClassLoader in which all the classes' resolutions are delegated.

METHODS

- *getClasspathUrls*
 public static final URL **getClasspathUrls**(java.lang.String classpath)
 - **Usage**
 - * Build the list of the URLs referred by the classpath. It must include all the directories, all the files, and all the jars contained in the classpath.

– **Parameters**

* **classpath** - String, classpath of the application. List of paths separated by the path.separator used by the system.

– **Returns** - URL[], list of individual URLs contained in the classpath.

– **Exceptions**

* **com.imagos.axilice.core.installer.ResourceException** - if any error take place during the building of the list of URLs enclosed by the classpath.

• *loadClass*

public Class loadClass(java.lang.String name)

METHODS INHERITED FROM CLASS `java.net.URLClassLoader`

• *findResource*

public URL findResource(java.lang.String arg0)

• *findResources*

public Enumeration findResources(java.lang.String arg0)

• *getURLs*

public URL getURLs()

• *newInstance*

public static URLClassLoader newInstance(java.net.URL [] arg0)

• *newInstance*

**public static URLClassLoader newInstance(java.net.URL [] arg0,
java.lang.ClassLoader arg1)**

METHODS INHERITED FROM CLASS `java.security.SecureClassLoader`

METHODS INHERITED FROM CLASS `java.lang.ClassLoader`

• *clearAssertionStatus*

public synchronized void clearAssertionStatus()

• *getParent*

public final ClassLoader getParent()

• *getResource*

public URL getResource(java.lang.String arg0)

• *getResourceAsStream*

public InputStream getResourceAsStream(java.lang.String arg0)

• *getResources*

public Enumeration getResources(java.lang.String arg0)

• *getSystemClassLoader*

public static ClassLoader getSystemClassLoader()

• *getSystemResource*

public static URL getSystemResource(java.lang.String arg0)

• *getSystemResourceAsStream*

public static InputStream getSystemResourceAsStream(java.lang.String arg0)

• *getSystemResources*

public static Enumeration getSystemResources(java.lang.String arg0)

- *loadClass*
public Class loadClass(java.lang.String arg0)
- *setClassAssertionStatus*
public synchronized void setClassAssertionStatus(java.lang.String arg0, boolean arg1)
- *setDefaultAssertionStatus*
public synchronized void setDefaultAssertionStatus(boolean arg0)
- *setPackageAssertionStatus*
public synchronized void setPackageAssertionStatus(java.lang.String arg0, boolean arg1)

35.1.2 CLASS ClientClasspathUpdater

This class extends the local classpath of the Java virtual machine to hold the new or updated installed modules. It makes the new modules available to the Java virtual machine.
Responsibilities:

- Classpath updating for join the new available modules.

DECLARATION

```
public class ClientClasspathUpdater
extends java.lang.Object
```

CONSTRUCTORS

- *ClientClasspathUpdater*
public ClientClasspathUpdater()

– **Usage**
* Default constructor.
- *ClientClasspathUpdater*
public ClientClasspathUpdater(java.lang.String classpath)

– **Usage**
* Constructor for a provided classpath (as String).
– **Parameters**
* **classpath** - String, classpath that must be added to the Java virtual machine.
- *ClientClasspathUpdater*
public ClientClasspathUpdater(java.net.URL [] lUrls)

– **Usage**
* Constructor for a provided classpath (as URL list).
– **Parameters**
* **lUrls** - URL[], list of urls of the jars that must be added to the Java virtual machine classpath.

METHODS

• *addClasspath*

```
public static void addClasspath( java.lang.String classpath )
```

– **Usage**

- * Add the directories and jar files included in the local classpath, to the System's classloader classpath.

– **Parameters**

- * *classpath* - String, local classpath to add.

– **Exceptions**

- * *com.imagos.axilice.core.installer.ResourceException* - if any error takes place during the update of the System's classpath.
-

• *addURL*

```
public static void addURL( java.net.URL u )
```

– **Usage**

- * Add the URL to the System's classloader classpath

– **Parameters**

- * *u* - URL, URL to add to the classpath.

– **Exceptions**

- * *ResourceException*, - if any error takes place during the update of the classpath (url not found, malformed, etc.)
-

• *getAbsolutePath*

```
public static String getAbsolutePath( java.lang.String path )
```

• *getClasspathUrls*

```
public static final URL getClasspathUrls( java.lang.String classpath )
```

– **Usage**

- * Build the list of the URLs referred by the classpath. It must include all the directories, all the files, and all the jars contained in the classpath.

– **Parameters**

- * *classpath* - String, classpath of the application. List of paths separated by the path.separator used by the system.

– **Returns** - URL[], list of individual URLs contained in the classpath.– **Exceptions**

- * *com.imagos.axilice.core.installer.ResourceException* - if any error take place during the building of the list of URLs enclosed by the classpath.
-

• *getRevision*

```
public static String getRevision( )
```

– **Usage**

- * Returns tag Id assigned to CVS source file.

35.1.3 CLASS ClientInstaller

This class implements the client manager of the installing process. It requests the resources list to the server of the application, writes the resources to the local path and updates the local catalog of installed resources, when required.
Responsibilities:

- Communication with the remote Installer server object.
- Install in the client machine the new or updated resources.
- Update the client catalog of installed resources.
- Configure the client application for using the new resources.

DECLARATION

```
public class ClientInstaller
extends java.lang.Object
```

METHODS

- *configClient*
 public void **configClient**()
 – **Usage**
 * Searches for new or updated modules and install them in the local client of the application.

- *configClient*
 public void **configClient**(Display **display**, ProgressBar **progressBar**,
 StatusBar **statusBar**, Shell **initWindow**)
 – **Usage**
 * Searches for new or updated modules and install them in the local client of the application.

- *getAbsolutePath*
 public static String **getAbsolutePath**(java.lang.String **path**)

- *getInstance*
 public static ClientInstaller **getInstance**()
 – **Usage**
 * Gets an instance of the ClientInstaller class.
 – **Returns** - ClientInstaller, the unique instance of the ClientInstaller class (Singleton pattern).

- *getModules*
 public Hashtable **getModules**()

- *getRevision*
 public static String **getRevision**()

– **Usage**

* Returns tag Id assigned to CVS source file.

• *setModules*

public void **setModules**(java.util.Hashtable **modules**)

• *updateProgress*

public void **updateProgress**(Display **display**, ProgressBar **progressBar**, int **progressInc**, StatusBar **statusBar**, java.lang.String **statusText**)