



- 1 Introduction**
- 2 User Perspective**
- 3 Developer Perspective**

This document is not to be reproduced, modified, adapted, published, translated in any material form in whole or in part nor disclosed to any third party without the prior written permission of Thales. © THALES 2013 – All rights reserved.

Context

- Models contain a huge amount of information
- Information cannot be displayed to end-user in one shot

Need

- Bringing out complementary information to the ones provided by editors (e.g., Ecore Editor, Sirius Diagram, Sirius Table, etc.)
- Bringing out relevant relationships between model elements
- Providing mechanisms for defining complex relationships

Objective

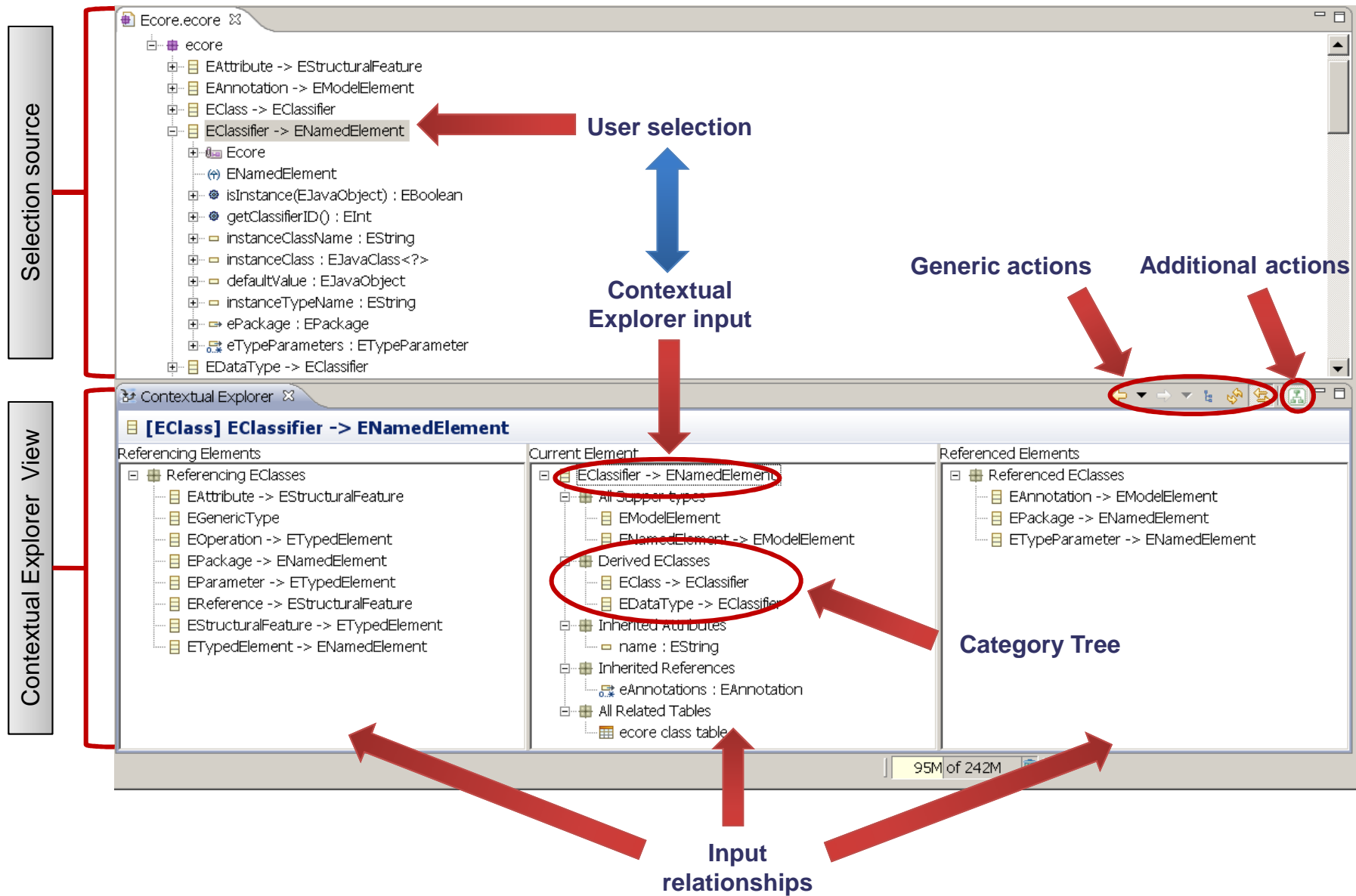
- The Contextual Explorer is a service which allows:
 - Providing universal three kind of relationships
 - Centering information display on one model element
 - Contextual browsing over model elements



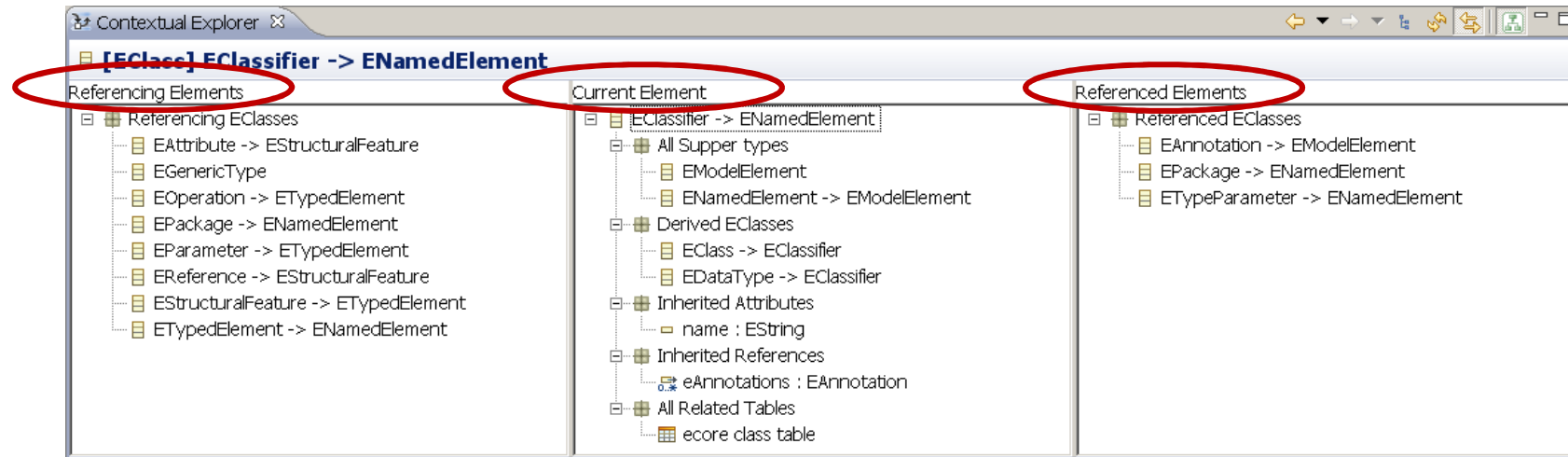
- 1 Introduction
- 2 User Perspective
- 3 Developer Perspective

This document is not to be reproduced, modified, adapted, published, translated in any material form in whole or in part nor disclosed to any third party without the prior written permission of Thales. © THALES 2013 – All rights reserved.

1. The User selects one model element
2. The Contextual Explorer analyzes the selection
3. The Contextual Explorer computes categories dealing with
 - a. Elements referencing the selected element
 - b. Element related to the selected element
 - c. Elements referenced by the selected element
4. The Contextual Explorer displays the result in three sections:
 - a. Referencing Elements
 - b. Current Element
 - c. Referenced Elements
5. The User applies actions on displayed elements
6. The User continues to navigate over the referencing or referenced elements



This document is not to be reproduced, modified, adapted, published, translated in any material form in whole or in part nor disclosed to any third party without the prior written permission of Thales. © THALES 2013 – All rights reserved.



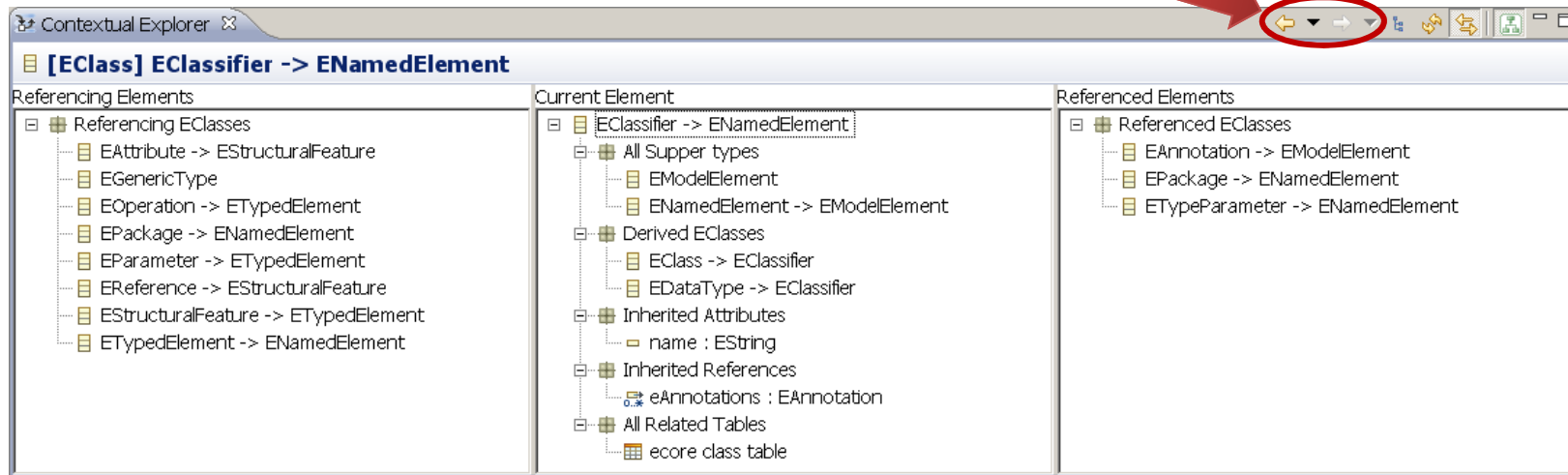
Objective

Displaying incoming and outgoing references and information on the selected model element

Actions

The user selects one element in it workbench (e.g., Open editor, project explorer, etc.)
The Contextual Explorer is, by default, updated after the user selection

Navigation actions



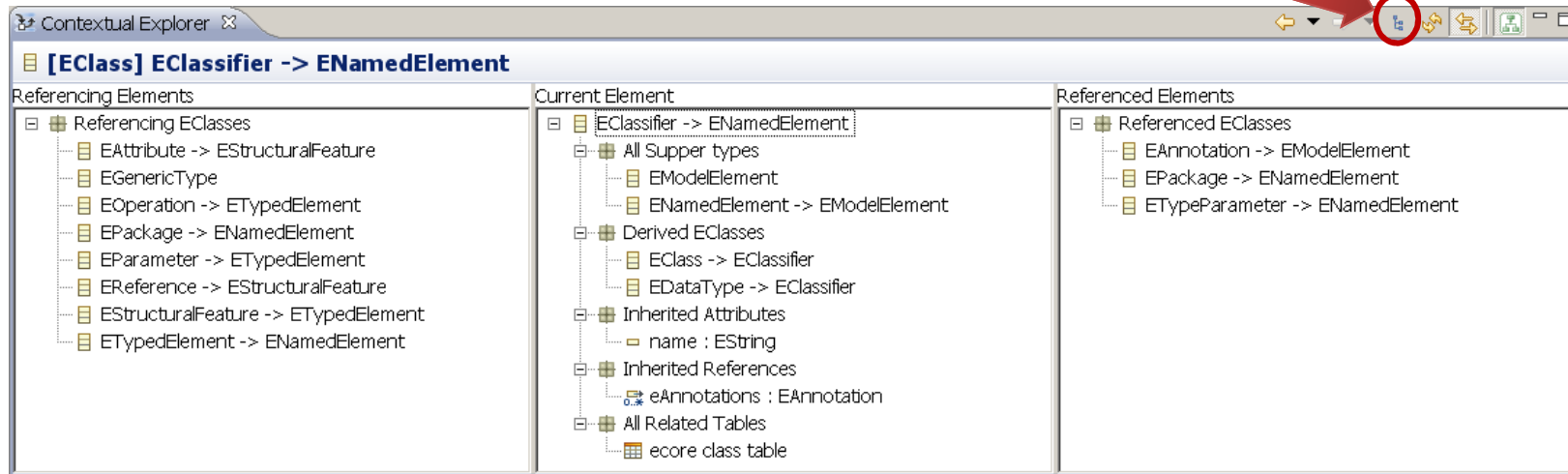
Objective

Keeping the traces on Contextual Explorer inputs and allowing the user to navigate in the navigation history

Actions

To navigate in the history, the user uses navigation toolbar actions

Tree limitation action



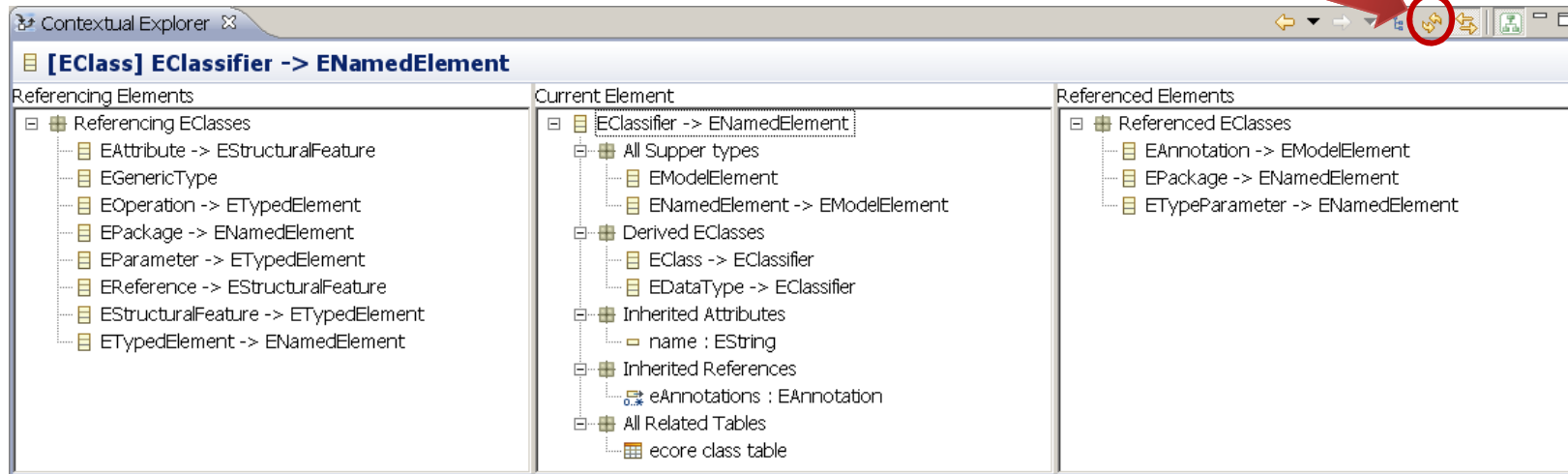
Objective

Collapsing all categories trees except the ones expanded by the user
 Categories tree will then keep their expansion state through navigation actions and future selections

Actions

- The user activates/deactivates the tree limitation by switching it on/off with the related toolbar action state
- If the tree limitation is activated, the user chooses the categories tree to expand

Refresh action



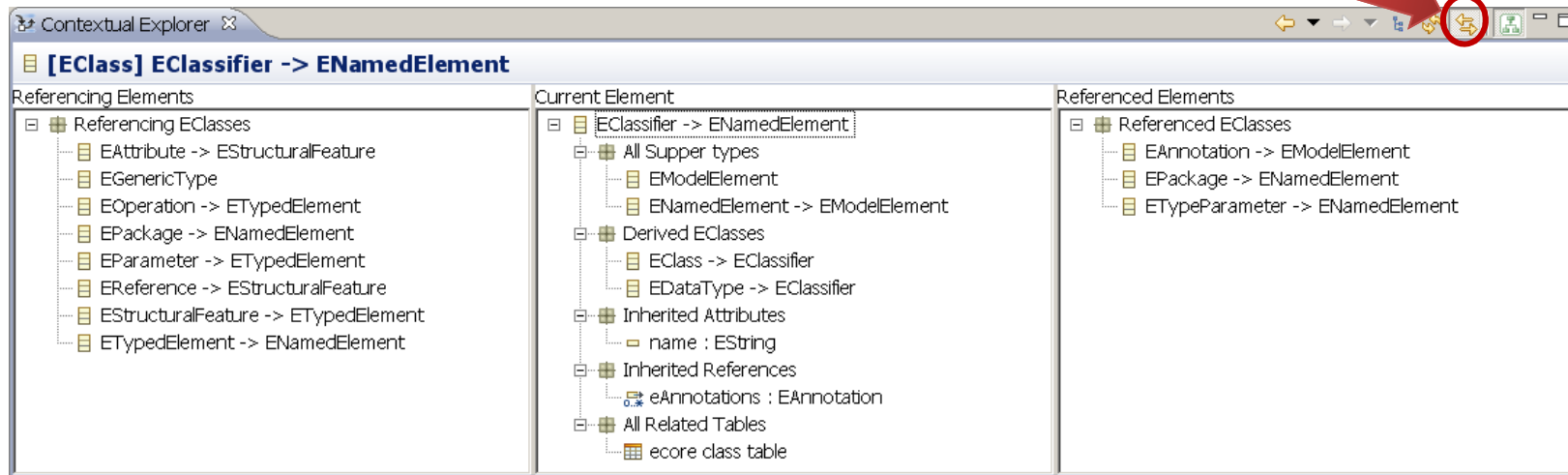
Objective

Refreshing the Contextual Explorer view

Actions

To refresh the view content, the user clicks on the Refresh toolbar action

Synchronize action



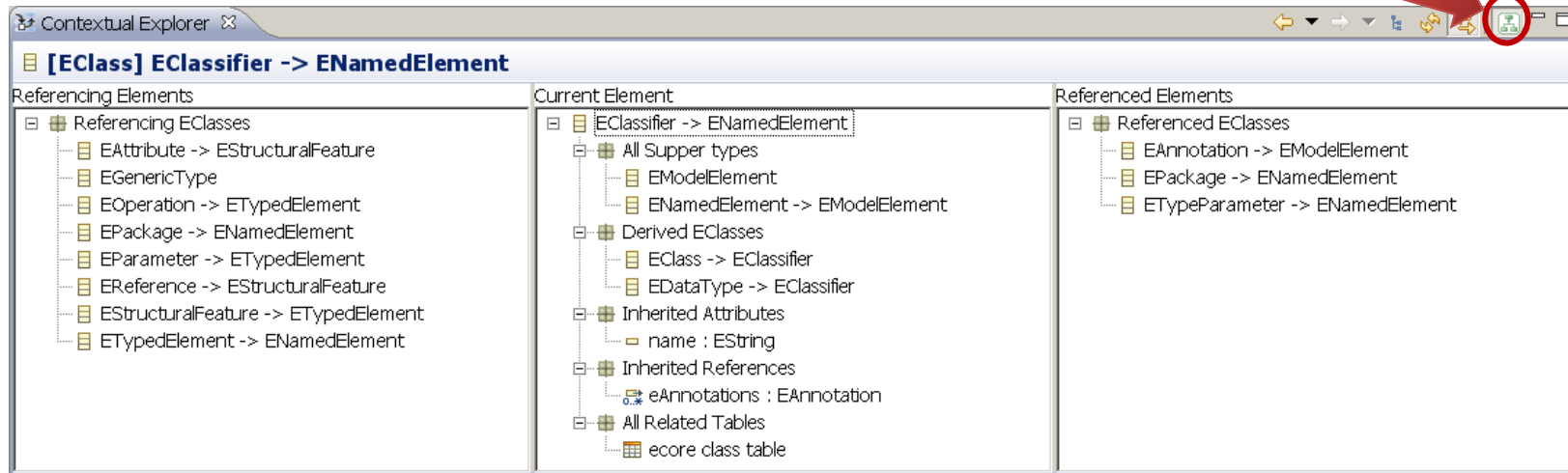
Objective

Keeping Contextual Explorer sensitive to workbench selection changes

Actions

The user activates/deactivates the selection synchronization by switching it on/off with the related toolbar action state

Additional action



This document is not to be reproduced, modified, adapted, published, translated in any material form in whole or in part nor disclosed to any third party without the prior written permission of Thales. © THALES 2013 – All rights reserved.

Objective

Providing additional actions

Actions

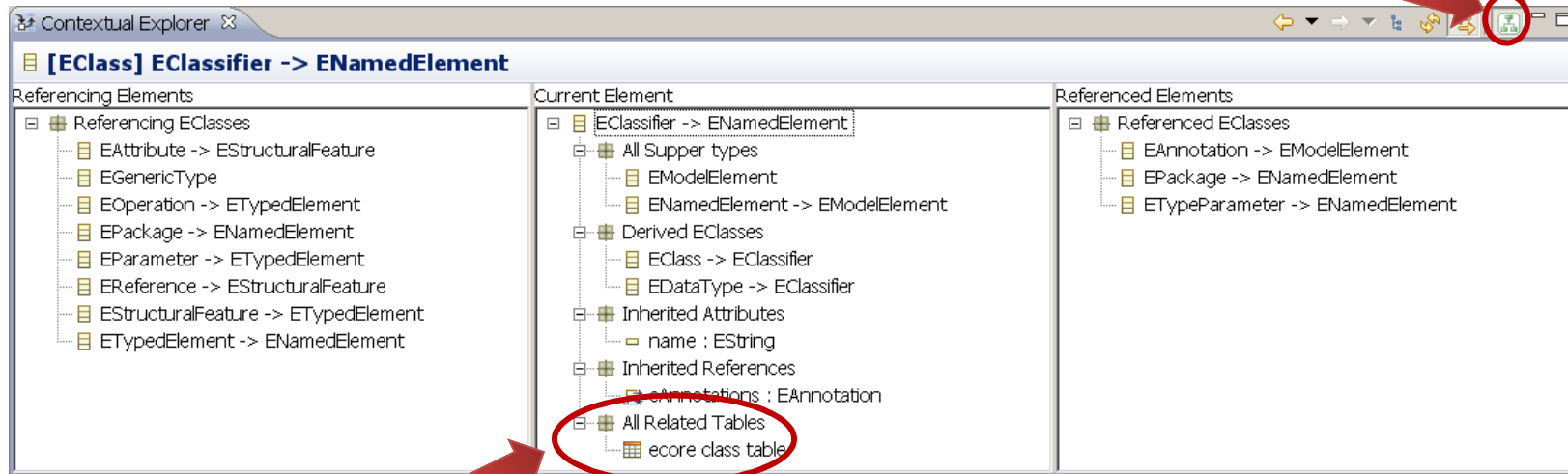
Execution of the action launched by the user

OPEN



Sirius Integration

Representations Show/Hide action



Sirius representations categories

Objective

Showing/Hiding Sirius representations wherein the current element is present

Actions

To Show/Hide Sirius representations, the user switches it on/off with the related toolbar action state

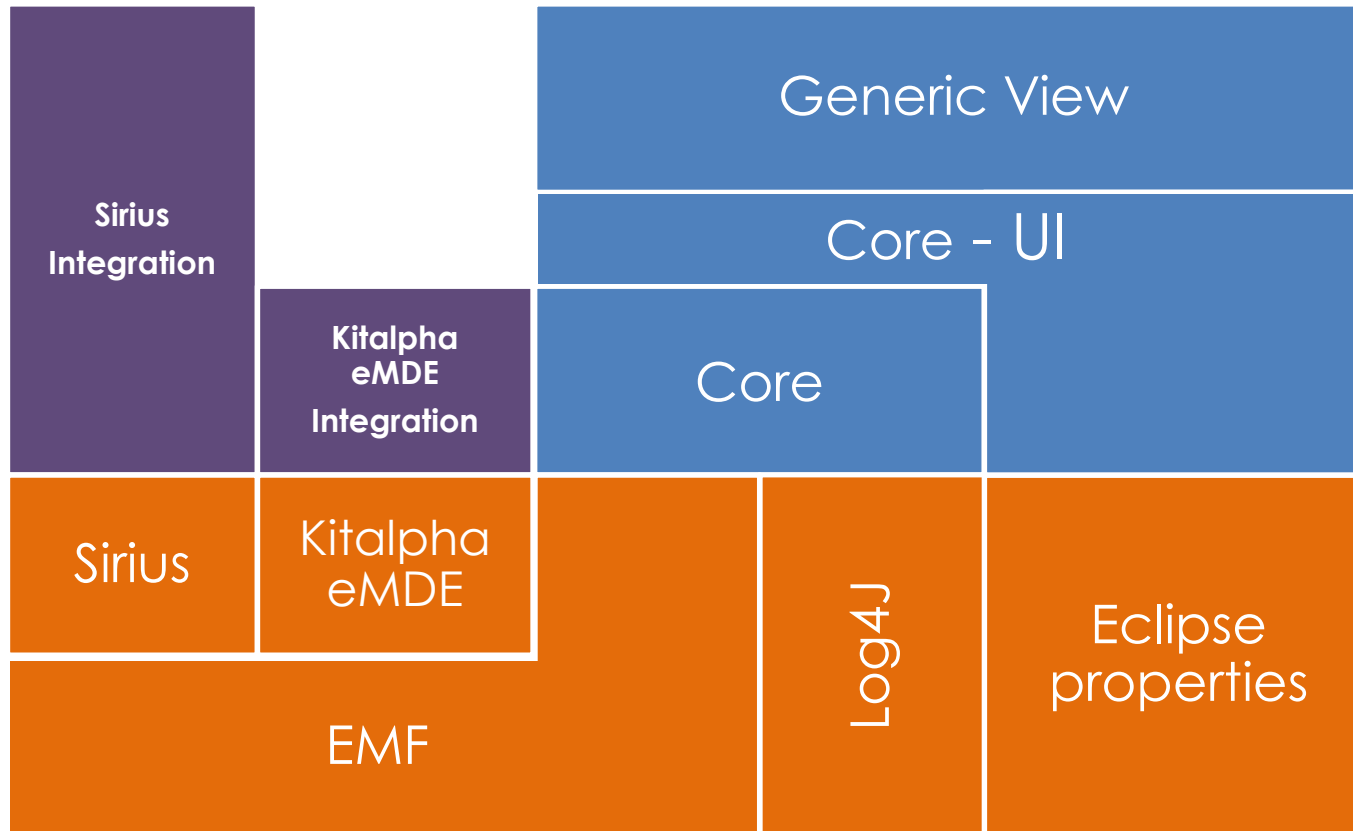


- 1 Introduction
- 2 User Perspective
- 3 Developer Perspective

This document is not to be reproduced, modified, adapted, published, translated in any material form in whole or in part nor disclosed to any third party without the prior written permission of Thales. © THALES 2013 – All rights reserved.

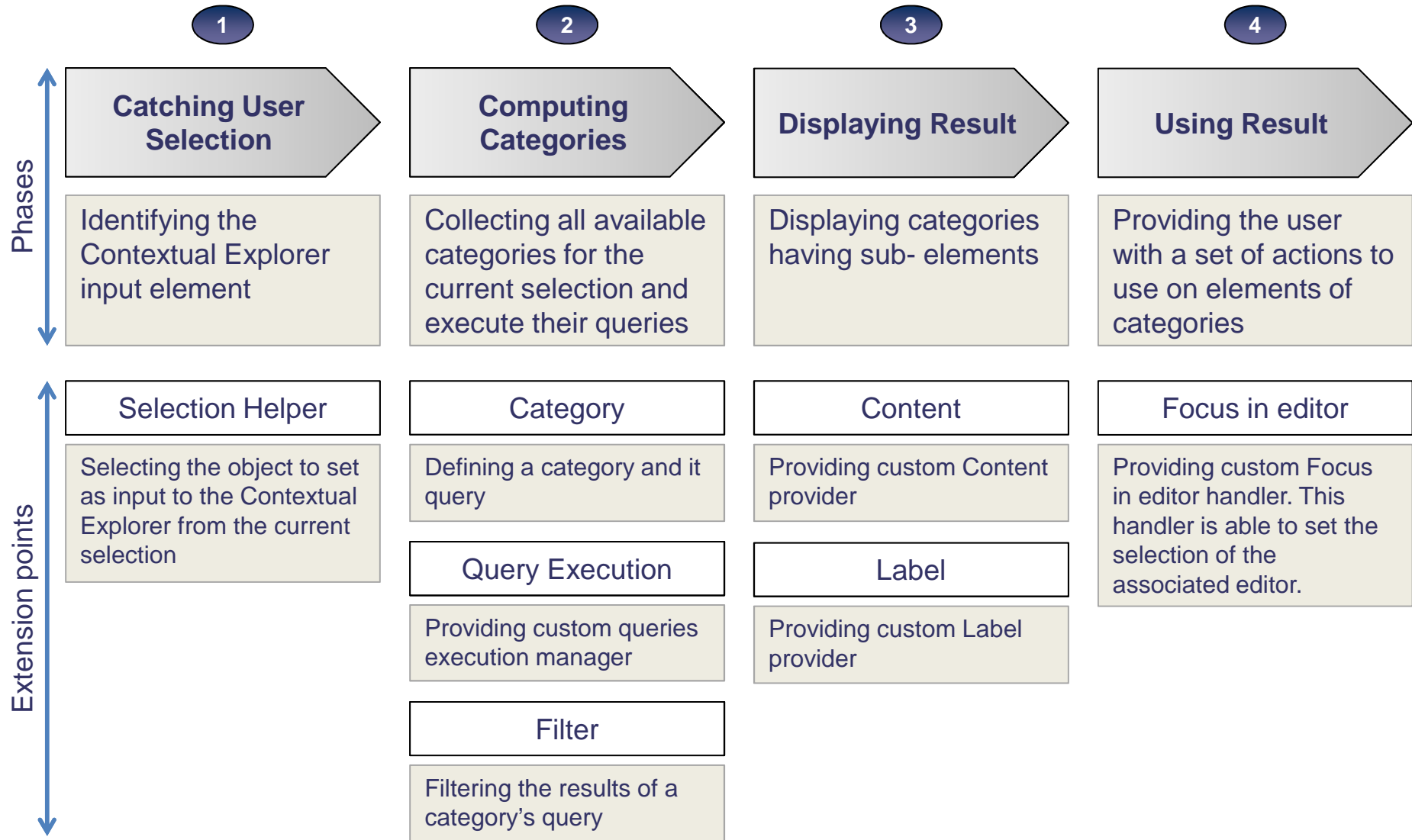
Foundations

- Contributions to Contextual Explorer
- Contextual Explorer component
- Used components



This document is not to be reproduced, modified, adapted, published, translated in any material form in whole or in part nor disclosed to any third party without the prior written permission of Thales. © THALES 2013 – All rights reserved.

Contextual Explorer workflow and extension points



This document is not to be reproduced, modified, adapted, published, translated in any material form in whole or in part nor disclosed to any third party without the prior written permission of Thales. ©THALES 2013 – All rights reserved.

Step 1 Listening to workbench selection changes

Implementation

Plugin name	org.eclipse.amalgam.explorer.contextual.core.ui
Java Package	org.eclipse.amalgam.explorer.contextual.core.ui.view
Class name	ExplorerContextualView

Extension point(s)

Name	Plugin	Schema
selectionHelper	org.eclipse.amalgam.explorer.contextual.core.ui	selectionHelper.exsd

Available extension(s)

Description	Sirius Selection Helper. This helper returns the target of a DSemanticDecorator
Plugin name	org.eclipse.amalgam.explorer.contextual.sirius
Java Package	org.eclipse.amalgam.explorer.contextual.sirius.selection
Class name	SiriusSelectionHelper

Step 2.1 Collecting categories

Implementation

Plugin name	org.eclipse.amalgam.explorer.contextual.core
Java Package	org.eclipse.amalgam.explorer.contextual.core.category
Class name	All package classes

Extension point(s)

Name	Plugin	Schema
contentProviderCategory	org.eclipse.amalgam.explorer.contextual.core	contentProviderCategory.exsd

Step 2.2 Category's query execution

Implementation

Plugin name	org.eclipse.amalgam.explorer.contextual.core
Java Package	org.eclipse.amalgam.explorer.contextual.core.query.impl
Class name	QueryExecutionManager

Extension point(s)

Name	Plugin	Schema
customQueryExecutionManager	org.eclipse.amalgam.explorer.contextual.core	customQueryExecutionManager.exsd

Default implementation

Description	An execution manager for IQuery derived classes
Plugin name	org.eclipse.amalgam.explorer.contextual.core
Java Package	org.eclipse.amalgam.explorer.contextual.core.query.impl
Class name	QueryExecutionManager

Step 2.3 Filtering Category's elements

Implementation

Plugin name	org.eclipse.amalgam.explorer.contextual.core
Java Package	org.eclipse.amalgam.explorer.contextual.core.filter
Class name	All package classes

Extension point(s)

Name	Plugin	Schema
candidateElementFilter	org.eclipse.amalgam.explorer.contextual.core	candidateElementFilter.exsd

Available extension(s)

Description	eMDE filter. This filter removes elements belonging to inactive extensions
Plugin name	org.eclipse.amalgam.explorer.contextual.emde
Java Package	org.eclipse.amalgam.explorer.contextual.emde.filter
Class name	ExplorerContextualEmdeFilter

Step 3.1 Content provider factory

Implementation

Plugin name	org.eclipse.amalgam.explorer.contextual.core
Java Package	org.eclipse.amalgam.explorer.contextual.core.provider
Class name	AbstractContentProviderFactory

Extension point(s)

Name	Plugin	Schema
contentProviderFactory	org.eclipse.amalgam.explorer.contextual.core	contentProviderFactory.exsd

Default implementation

Description	Creating default ITreeContentProvider for the current, referenced and referencing sections
Plugin name	org.eclipse.amalgam.explorer.contextual.core
Java Package	org.eclipse.amalgam.explorer.contextual.core.provider
Class name	DefaultContentProviderFactory

Step 3.2 Label provider factory

Implementation

Plugin name	org.eclipse.amalgam.explorer.contextual.core
Java Package	org.eclipse.amalgam.explorer.contextual.core.provider
Class name	AbstractLabelProviderFactory

Extension point(s)

Name	Plugin	Schema
labelProviderFactory	org.eclipse.amalgam.explorer.contextual.core	labelProviderFactory.exsd

Default implementation

Description	Creating a unique ILabelProvider based on the generated IItemLabelProvider to get element label and image. This label provider is used for current, referenced and referencing sections.
Plugin name	org.eclipse.amalgam.explorer.contextual.core
Java Package	org.eclipse.amalgam.explorer.contextual.core.provider
Class name	DefaultLabelProviderFactory

Step 4.1 Show/Hide categories toolbar actions

Implementation

Plugin name	org.eclipse.amalgam.explorer.contextual.core.ui
Java Package	org.eclipse.amalgam.explorer.contextual.core.ui.action
Class name	AbstractShowHideAction

Extension point(s)

Name	Plugin	Schema
viewActions	org.eclipse.ui	viewActions.exsd

Extension(s)

Description	Actions must be defined by using eclipse mechanisms. The action class can extend AbstractShowHideAction. This class facilitate defining initial state, showing and hiding the associated category/categories	
View Contribution	Target ID	org.eclipse.amalgam.explorer.contextualview.ExplorerContextualID
Action	Style	Toggle
	State	True if the associated category is active by default, false otherwise

Step 4.2 Popup menu

Extension point(s)

Name	Plugin	Schema
Menus	org.eclipse.ui	menus.exsd

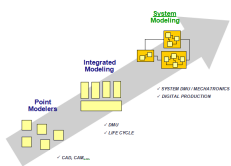
Extension(s)

Description	Menu contributions must be defined by using eclipse mechanisms. No APIs are provided.	
Menu contribution	locationURI	popup:org.eclipse.amalgam.explorer.contextualview.ExplorerContextualID ?after=additions

The Contextual Explorer has been developed in the context of PolarSys by Capella and Kitalpha

 **Capella**
<http://polarsys.org/capella/>

 **Kitalpha**
<http://polarsys.org/kitalpha/>



Capella has been supported by **Clarity**, a French collaborative project
Kitalpha has been supported by **Sys2Soft**, **Crystal**, and **Clarity**, French and European collaborative projects