



www.thalesgroup.com

Model Detachment

 **Kitalpha**



OPEN
Version 1.0.0

THALES



- 1 Introduction
- 2 User Perspective
- 3 Developer Perspective
- 4 Contract of Detachment Service

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 Introduction**
- 2 User Perspective**
- 3 Developer Perspective**
- 4 Contract of Detachment Service**

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

- A model is used by different stakeholders
- Each stakeholder has his own tooling environment (e.g., with installed and uninstalled viewpoints)

Need

- Removing data from a model
- Use cases: reasons of confidentiality or security between stakeholders / partners / sub-contractors; model cleanup; uninstallation of viewpoints

Objective

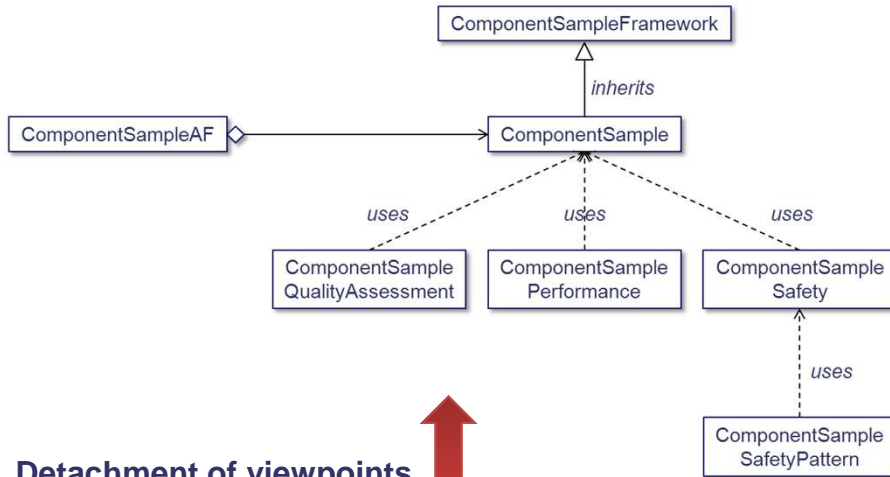
- The Model Detachment is a service which allows removing data (i.e., model and representations) thanks to detachment criteria



- 1 Introduction
- 2 User Perspective
- 3 Developer Perspective
- 4 Contract of Detachment Service

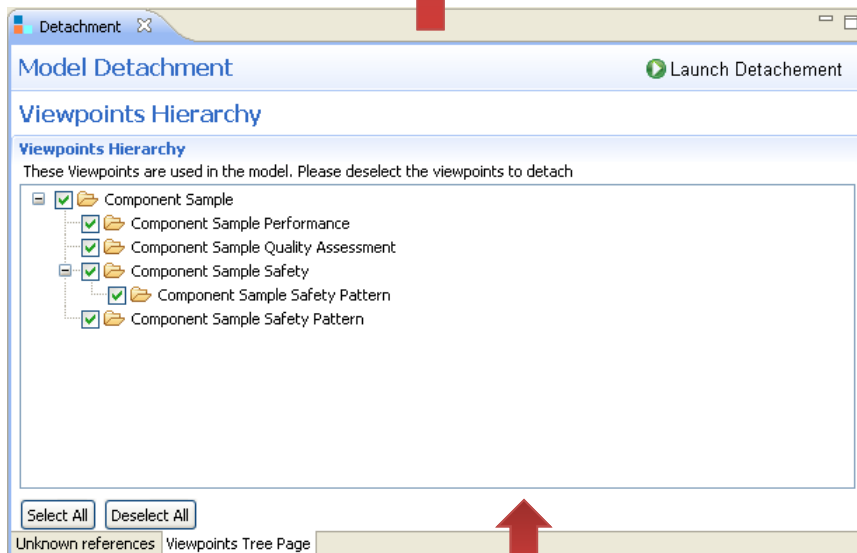
This document is not to be reproduced, modified, adapted, published, translated in any material form in whole or in part nor disclosed to any third party without the prior written permission of Thales. ©THALES 2013 – All rights reserved.

- 1.The User launches the model detachment
- 2.The Model Detachment analyzes the model
- 3.The User selects the data to be detached
- 4.The Model Detachment backs up the model
- 5.The Model Detachment removes the data from the model

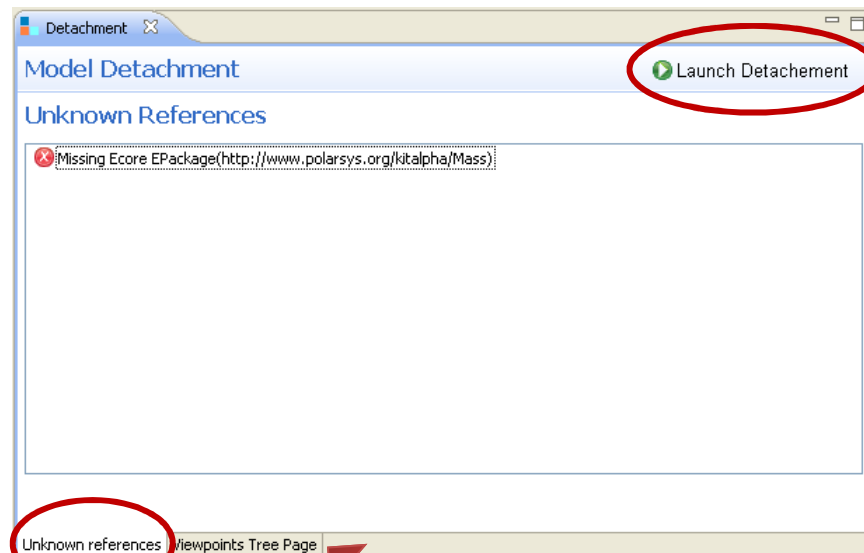


Detachment of viewpoints

Command to launch the detachment



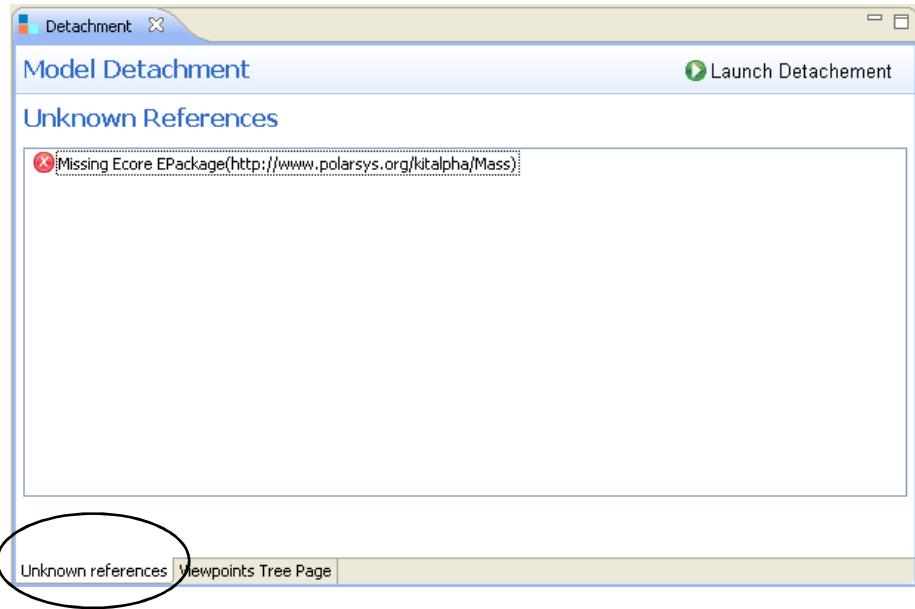
Selection area for a detachment topic



Topic of detachment

Extensible list of detachment topics

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.



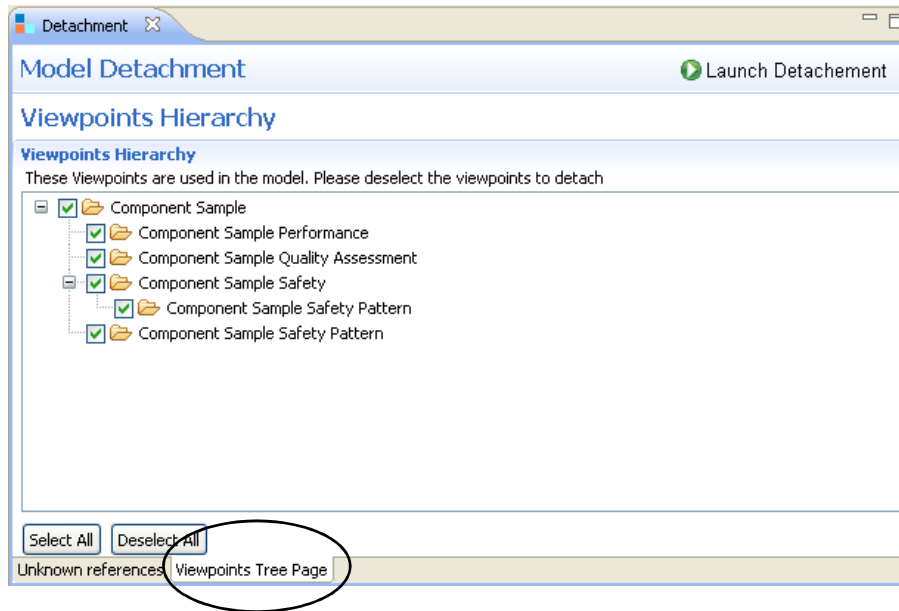
Objective

Removing all the unknown references which prevent to open or work with a model

Actions

No action required from the user

OPEN



Objective

Detaching data and representations of a viewpoint

Actions

The user unselects the viewpoint to be detached

Model Detachment Launch Detachement

Model Resources

Known Resources
List of resolved resources used by the model.
Deselect resources to delete

- platform:/resource/example/example.aird
- platform:/resource/example/fragments/OA.melodyfragment
- platform:/resource/example/fragments/PA.melodyfragment
- platform:/resource/example/example.melodymodeller

Unknown Resources
List of unresolved resources used by the model.
Deselect resources to delete

- platform:/resource/example/fragments/EPBSA.melodyfragment

Viewpoints Tree Page **Model Resource** Unknown references

Objective

Unreferencing resources (e.g., library) all over a model

Actions

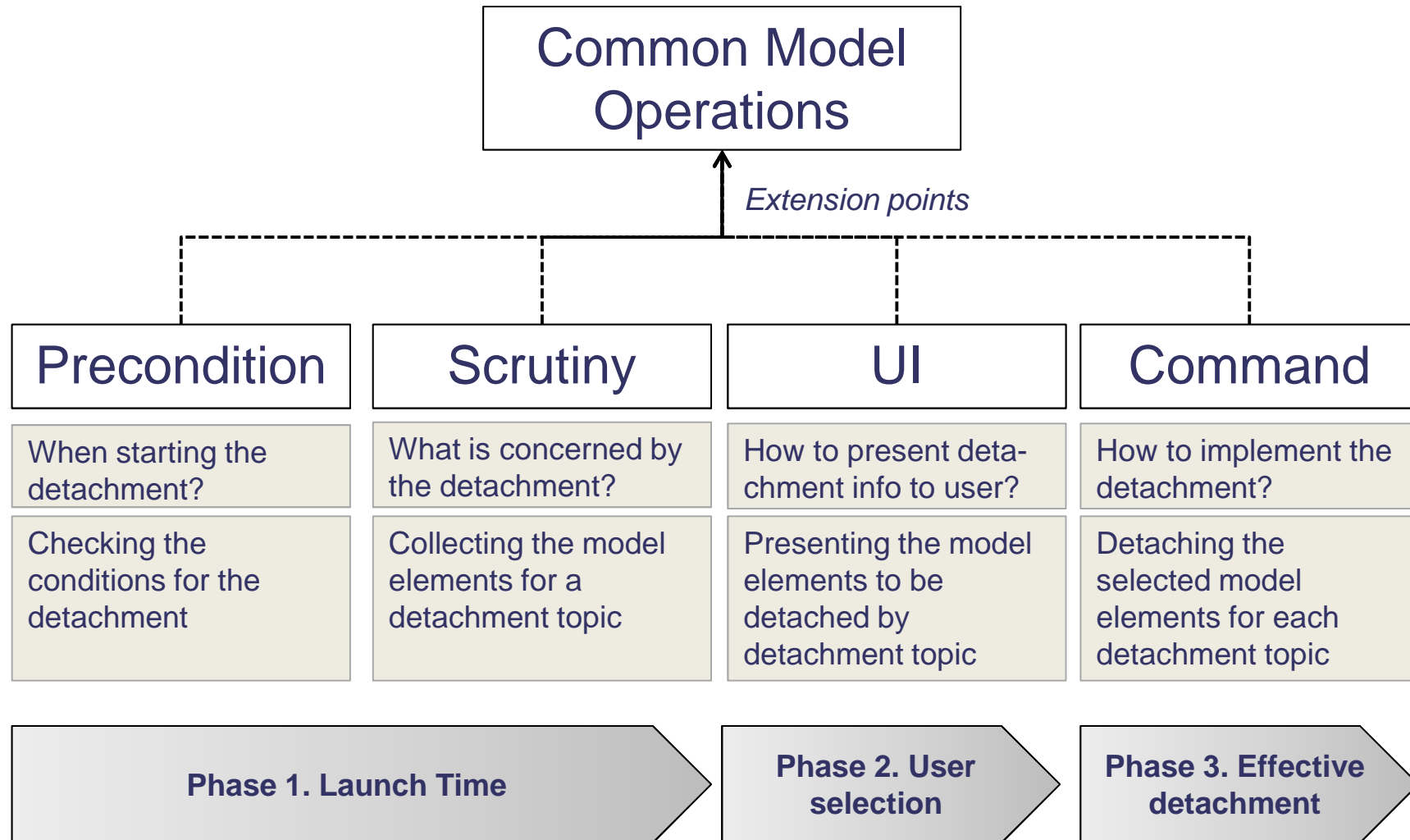
The user unselects the resources to be detached, known (installed) or not in the platform



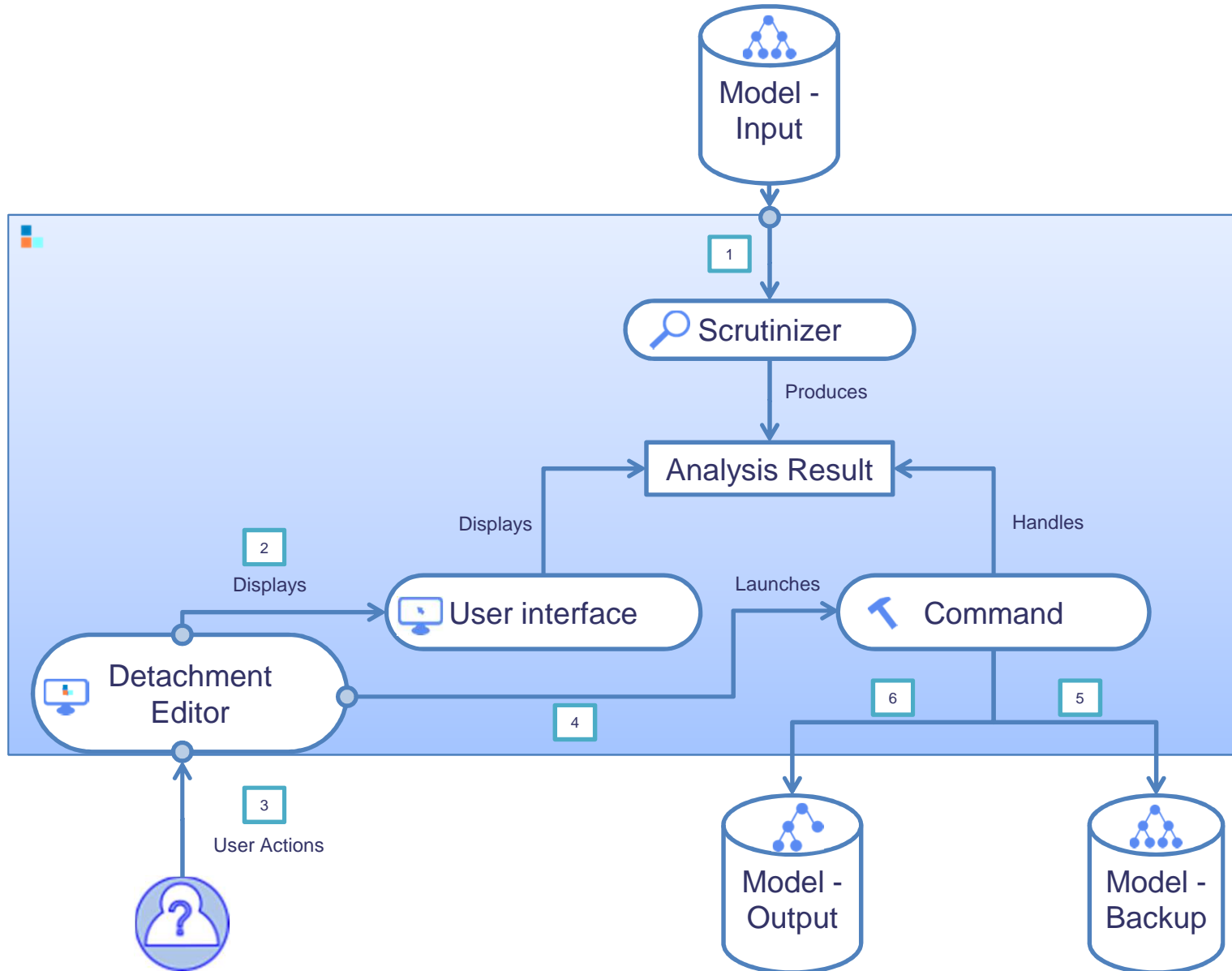
- 1 Introduction
- 2 User Perspective
- 3 Developer Perspective
- 4 Contract of Detachment Service

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



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.



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 1.1. Start up – Verification of the detachment preconditions

Implementation

Plugin name	org.polarsys.kitalpha.model.detachment.ui
Java Package	org.polarsys.kitalpha.model.detachment.ui.action

Extension point(s)

Name	Plugin	Schema
Precondition	org.polarsys.kitalpha.model.common.precondition	precondition.exsd

Step 1.2. Scrutiny

Extension point(s)

Name	Plugin	Schema
Scrutiny	org.polarsys.kitalpha.model.common.scrutiny	scrutinize.exsd

Extension point element	Description
id	Scrutinizer id
name	Scrutinizer name
class	Implementation of the scrutinizer interface org.polarsys.kitalpha.model.common.scrutiny.interfaces.IScrutinize

Example

```
public class ScrutinizerContribution implements IScrutinize<Object, Object> {
    // First argument: result of the analysis
    // Second argument: for the getFeedbackAnalysisMessages method

    @Override
    public void findIn(EObject arg0) {
        // Applies selection criteria to select or not an object
        // The selected objects are saved by a means defined by the developer
    }

    @Override
    public void findIn(Resource arg0) {
        // Same as findIn(EObject arg0) but for a resource
    }

    @Override
    public Object getAnalysisResult() {
        // Returns the result of the findIn method
        return null;
    }

    @Override
    public Object getFeedbackAnalysisMessages() {
        // Deprecated
        // Displays messages in the user interface
        return null;
    }
}
```

Contribution for viewpoint extension

Name	Plugin
Scrutiny	org.polarsys.kitalpha.model.common.scrutiny.contrib.viewpoints

User Interface

Extension point(s)

Name	Plugin	Schema
UI	org.polarsys.kitalpha.model.detachment.ui	page.exsd

Extension point element	Description
id	Id of the user interface contribution
scrutinizeId	Id of the scrutinizer (Cf. Step 1.2)
title	Title of page in the user interface
class	Implementation class of the user interface page

Example

```

public class UIPageImplementation extends AbstractDetachmentFormPage {

    public UIPageImplementation(FormEditor editor, String id, String title) {
        super(editor, id, title);
        // TODO Auto-generated constructor stub
    }

    @Override
    protected void createFormContent(IManagedForm managedForm) {
        // Page implementation with Forms
        super.createFormContent(managedForm);
    }
}

```

OPEN

THALES

Extension point(s)

Name	Plugin	Schema
Command	org.polarsys.kitalpha.model.common.commands	command.exsd

Extension point element	Description
Sequence	
id	Id of the command contribution
scrutinizeId	Id of the scrutinizer (Cf. Step 1.2)
priority	Order of the command in the sequence of commands
workflow	Indicates whether the command is applied on the detachment, attachment or both
class	Implementation class of the command

Example

```

public class CommandImplementation extends ModelCommand {

    @Override
    public void exec(Resource arg0, IProgressMonitor arg1) throws
    ModelCommandException {
        // Implementation of the command
    }
}

```

Extension point(s)

Name	Plugin	Schema
Command	org.polarsys.kitalpha.model.common.commands	command.exsd

Step 3.1. Additional resource checking

Default extension(s)

Name	Plugin
Command	org.polarsys.kitalpha.model.common.commands.contrib.resourcesChecker

Step 3.2. Detachment for each detachment topic

Step 3.3. Resource refresh (e.g., diagram)

Default extension(s)

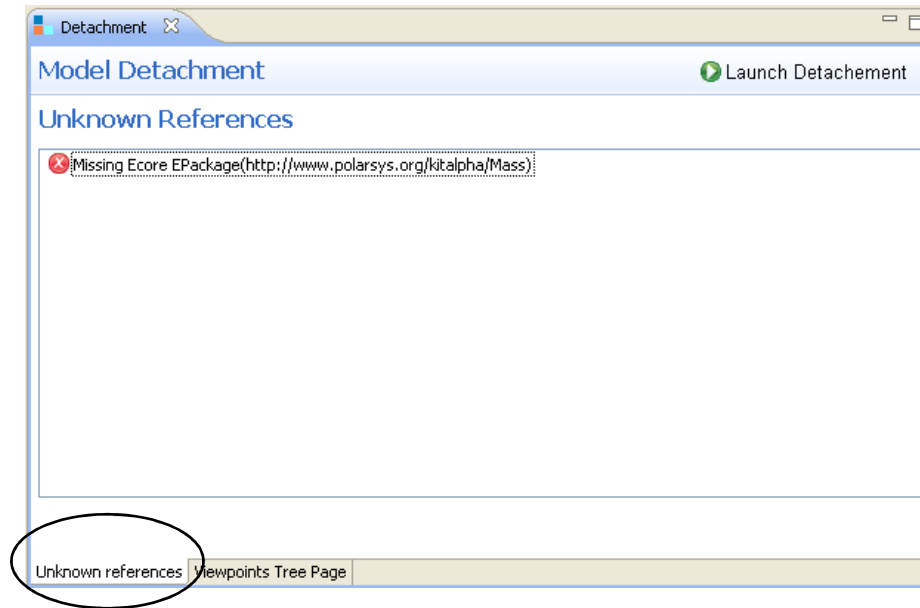
Name	Plugin
Command	org.polarsys.kitalpha.model.common.commands.contrib.refresh

Step 3.4. Back up and save

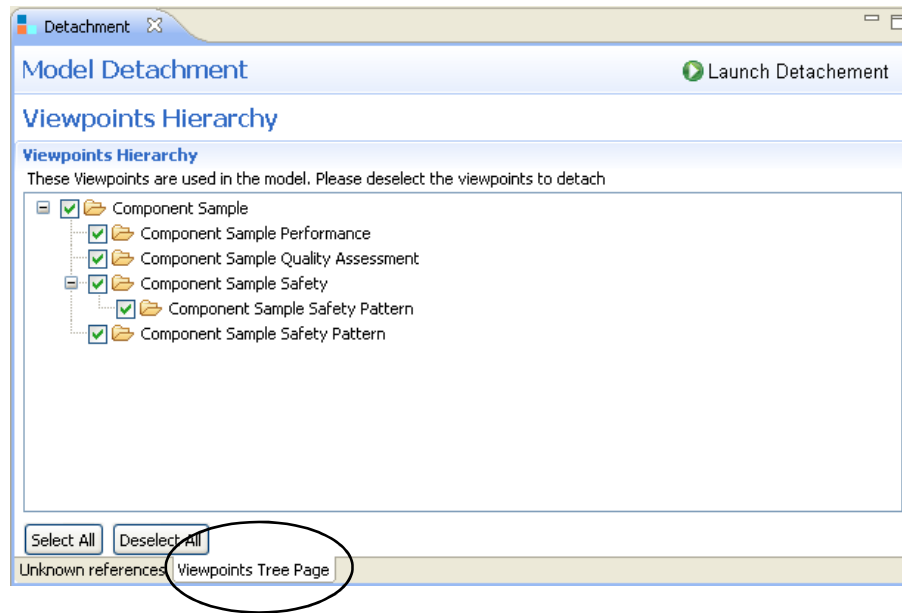
Default extension(s)

Name	Plugin
Command	org.polarsys.kitalpha.model.common.commands.contrib.backup

Detachment Topics



Extension	Plugin name
Scrutiny	org.polarsys.kitalpha.model.common.scrutiny.contrib.unknownreferences
UI	org.polarsys.kitalpha.model.detachement.ui.contrib.unknownreferences
Command	org.polarsys.kitalpha.model.common.commands.contrib.unknownreferences



Extension	Plugin name
Scrutiny	org.polarsys.kitalpha.model.common.scrutiny.contrib.viewpoints
UI	org.polarsys.kitalpha.model.detachment.ui.contrib.viewpoints
Command	org.polarsys.kitalpha.model.common.commands.contrib.viewpoints

Model Detachment ▶ Launch Detachement

Model Resources

Known Resources

List of resolved resources used by the model.
Deselect resources to delete

- platform:/resource/example/example.aird
- platform:/resource/example/fragments/OA.melodyfragment
- platform:/resource/example/fragments/PA.melodyfragment
- platform:/resource/example/example.melodymodeller

Unknown Resources

List of unresolved resources used by the model.
Deselect resources to delete

- platform:/resource/example/fragments/EPBSA.melodyfragment

Viewpoints Tree Page **Model Resource** Unknown references

Extension	Plugin name
Scrutiny	org.polarsys.kitalpha.model.common.scrutiny.contrib.modelresources
UI	org.polarsys.kitalpha.model.common.ui.contrib.modelresource
Command	org.polarsys.kitalpha.model.common.commands.contrib.modelresources



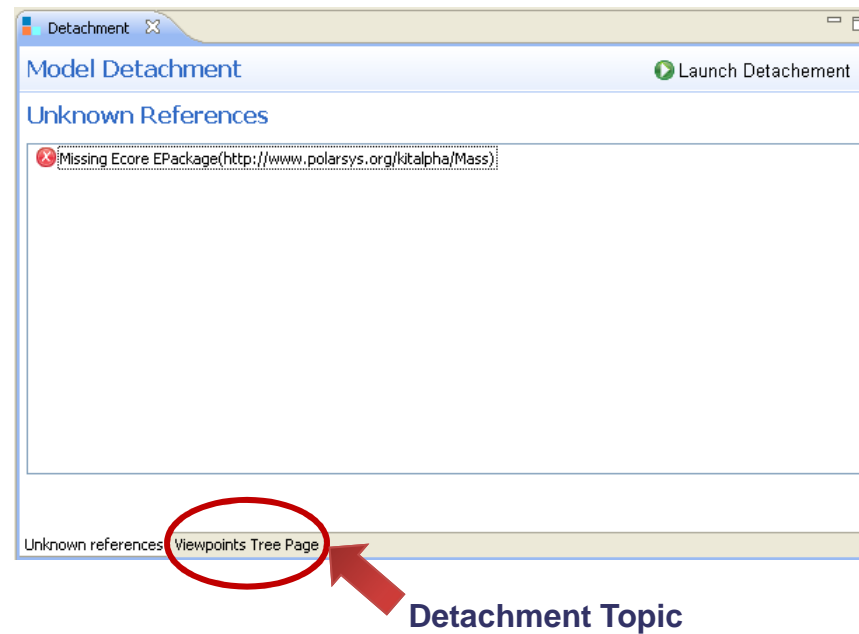
- 1 Introduction
- 2 User Perspective
- 3 Developer Perspective
- 4 Contract of Detachment Service

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.

Conditions to respect for the detachment service

1. Conditions for a new detachment topic
2. Conditions for viewpoint detachment

Objective: Implementation of a new topic of model detachment.



1. Contributing to the Scrutinizer

- Purpose: for selection of the model elements eligible to be detached
- Contribution: *org.polarsys.kitalpha.model.common.scrutiny* extension point
- Understanding: Cf. Phase 1-Step 1.2

2. Contributing to the User Interface

- Purpose: addition of a specific page in the detachment user interface
- Contribution: *org.polarsys.kitalpha.model.detachment.ui* extension point
- Understanding: Cf. Phase 2-User Interface

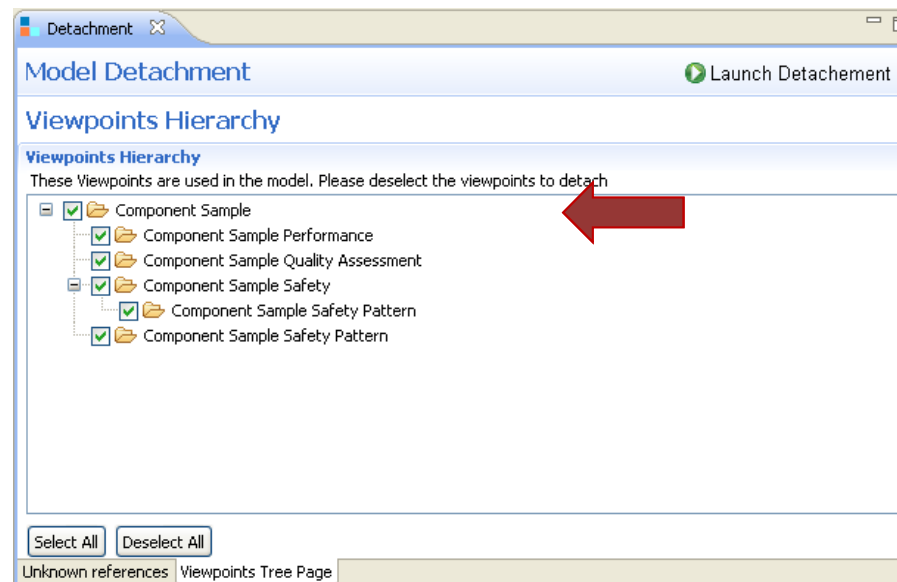
3. Command

- Purpose: Implementation of the detachment actions
- Contribution:
org.polarsys.kitalpha.model.common.commands.action.ModelCommand ext. point
- Understanding: Cf. Phase 3-Effective Detachment

Objective: Declaration of viewpoint metamodel to be detached.

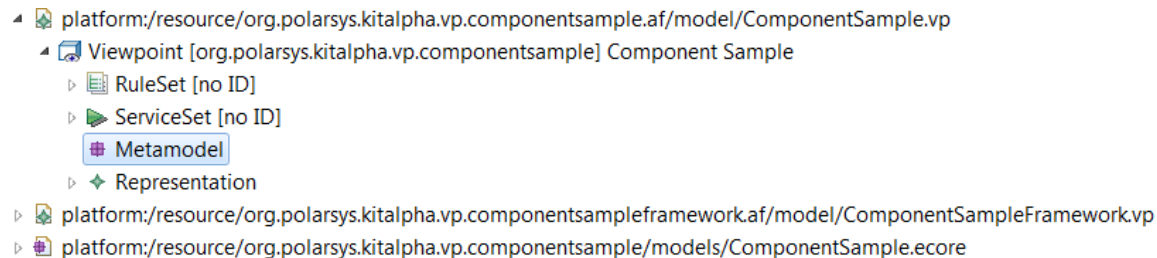
This enables that:

1. In the detachment user interface, a viewpoint to be detached is presented in the list of viewpoints,
2. The viewpoint is effectively detached.



1. Declaration of the viewpoint metamodel in a ".vp" viewpoint model

- Purpose: declaration that an ecore model contributes to a viewpoint
- Contribution:
 1. Creation of a ".vp" model
 2. Declaration of the ecore path in the metamodel element



2. Declaration of the viewpoint model to ResourceReuse

- Purpose: declaration of a viewpoint in the list of all the viewpoints
- Contribution: *org.polarsys.kitalpha.resourcereuse.resources* extension point.
Elements values: 1) id, name, path of the model, 2) domain = "AF", 3) tags = "vp"

All Extensions
Define extensions for this plug-in in the following section.

type filter text

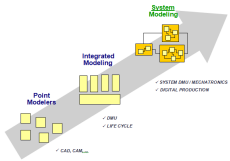
- org.eclipse.ui.views
- org.polarsys.kitalpha.resourcereuse.resources
 - Component Sample (resource)
- org.eclipse.ui.propertiesView
- org.polarsys.kitalpha.ad.viewpoint.javarules
- org.polarsys.kitalpha.ad.viewpoint.javarules
- org.polarsys.kitalpha.ad.viewpoint.javarules
- org.polarsys.kitalpha.ad.viewpoint.javarules

Buttons: Add..., Remove, Up, Down

Extension Element Details
Set the properties of 'resource' Required fields are denoted by '*'.

id*: org.polarsys.kitalpha.vp.componentsample
 name*: Component Sample
 path*: /org.polarsys.kitalpha.vp.componentsample.af/model/ComponentSam
 version:
 domain: AF
 tags: vp
 weight:

Overview Dependencies Runtime Extensions Extension Points Build MANIFEST.MF plugin.xml build.properties



Kitalpha is supported by
Sys2Soft, Crystal, and Clarity,
 French and European projects



Thank You!

<https://polarsys.org/kitalpha/>

benoit.langlois@thalesgroup.com

[#LangloisBenoit](#)