

TOPCASED

Toolkit In OPen source for Critical Applications & SystEms Development

- **General presentation of the project**
- A meta-modeling toolset
- The toolset architecture
- Services & Formats
- Demo / screenshots
- Conclusion

TOPCASED

Toolkit In OPen source for Critical Applications & SystEms Development

The creation of industrial systems relies on numerous tools on which it is essential to capitalize in order to optimize development costs.

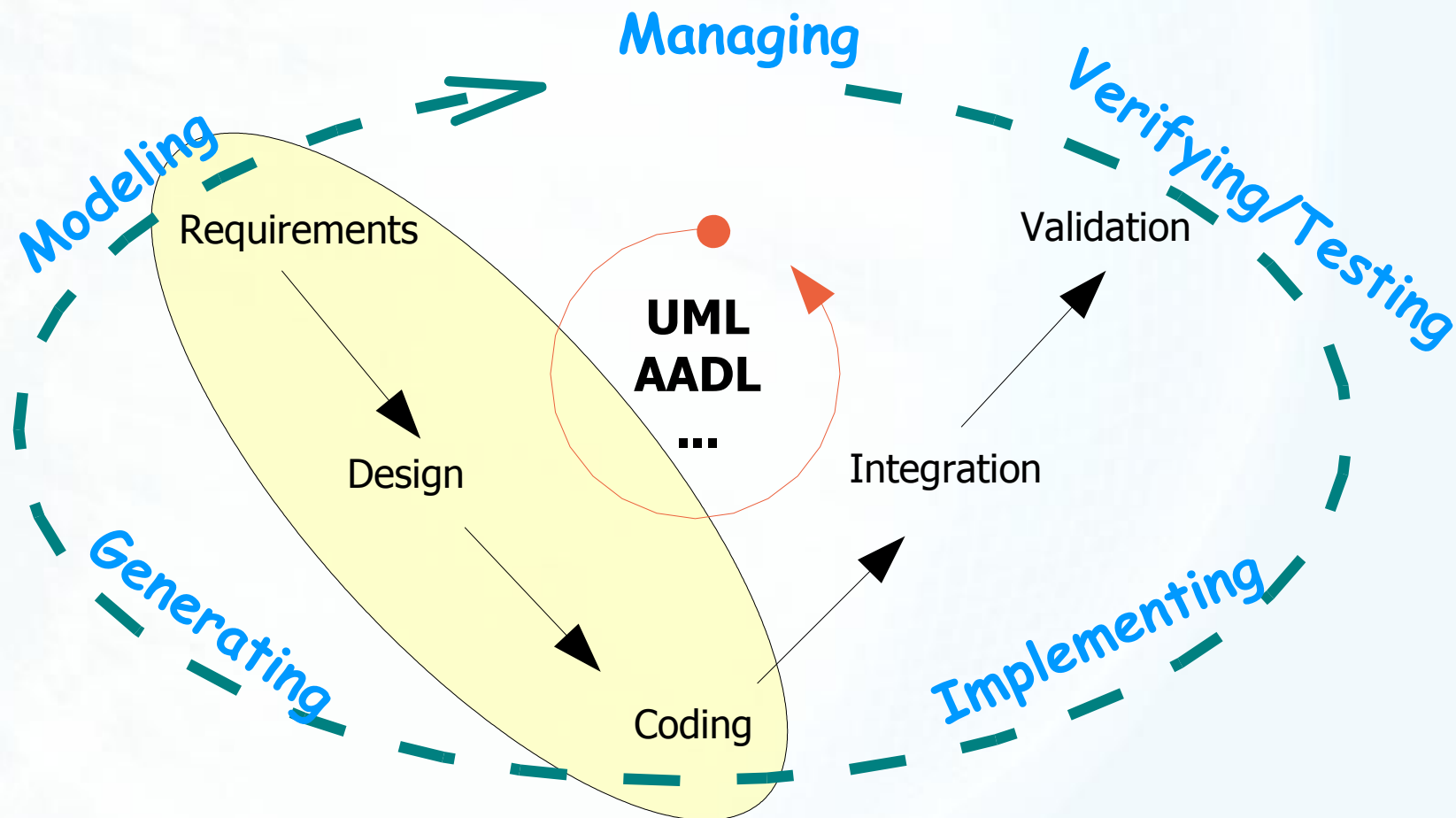
The lifetime of aerospace products is often as long as 10 to 30 years, and today, no software editor is able to commit for such long time at an acceptable cost.

Project Goals

- Perpetuate methods & tools for system and software development
- Minimize ownership costs
- Ensure the independence of development platforms
- Integrate as soon as possible, advances made in the academic world, and methodological changes
- Be able to adapt the tools to the process, and not the opposite
- Take into account qualification constraints

Project Perimeter

Requirements to code/implementation phases



Open Source ?

- Ensure the availability of the tools even on a long period
- It is possible to keep control on the tools life
- Costs are not directly dependent on the number of users, but on the development effort
- Development costs can be shared
- There is already a lot of valuable tools in open source : GCC, GDB, ...

But...

- Licensing plans have to be carefully studied
- The maintenance mechanism has to be defined to ensure efficiency
- The current business model has to evolve

implémenter

vérifier

TOPCASED

Atelier Développement Open Source

tester

gérer

générer

modéliser

réaliser

réaliser

Partnerships



UNIVERSITÉ
PAUL
SABATIER



TOULOUSE III



CENTRE NATIONAL D'ETUDES SPATIALES



Collaborations

- IST Modelware : Meta-modeling language & ModelBus (<http://www.modelware-ist.org>)
- Eclipse iMDD : a project supported by the Eclipse foundation aiming to distribute model oriented Eclipse features
- The TopModL initiative (<http://www.topmodl.org>) : generation of editors using meta models, semantic informations and meta modeling languages
- SAE AS-2C, SEI and IST ASSERT : AADL tools
- Use of finished projects NEPTUNE & COTRE : static & dynamic model verifications

Work packages

- WP1 – Processus
- WP2 – Modelling
- WP3 – Verification & simulation
- WP4 – Coding
- WP5 – Transformations
- WP6 – Interoperability
- WP7 – Infrastructure
- WP8 – Safety

Development & Maintenance Means

- A Gforge server is hosted by the ENSEEIHT (engineers school in Toulouse, France)
- The web site is hosted by the CNRS-LAAS in Toulouse
- A leading committee has been defined
- Fundings : private funds at this time, but public research projects are also targeted (Aerospace Valley french competitiveness pole, ITEA, IST, ...)

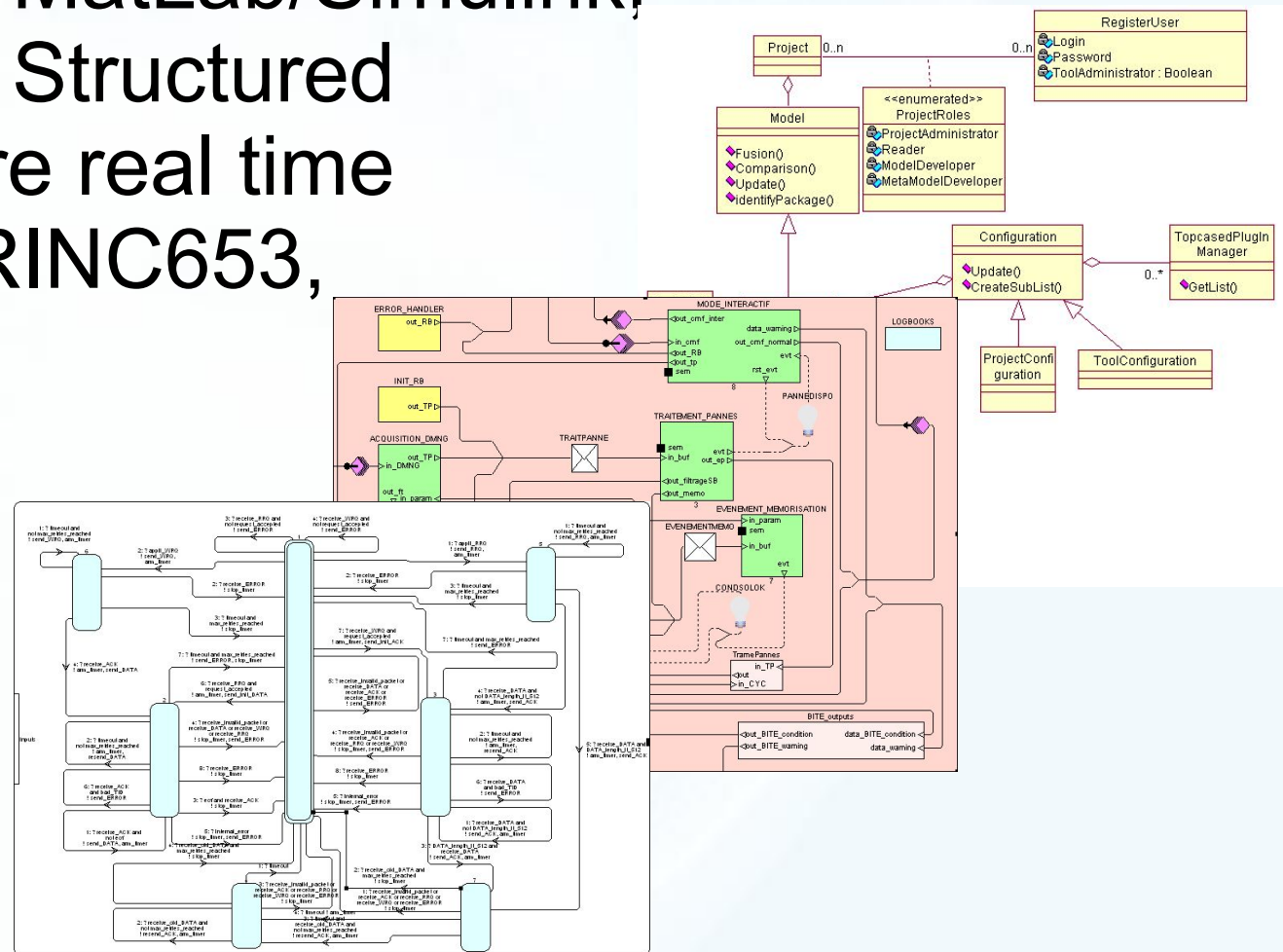
TOPCASED

Toolkit In OPen source for Critical Applications & SystEms Development

- General presentation of the project
- **A meta-modeling toolset**
- The toolset architecture
- Services & Formats
- Demo / screenshots
- Conclusion

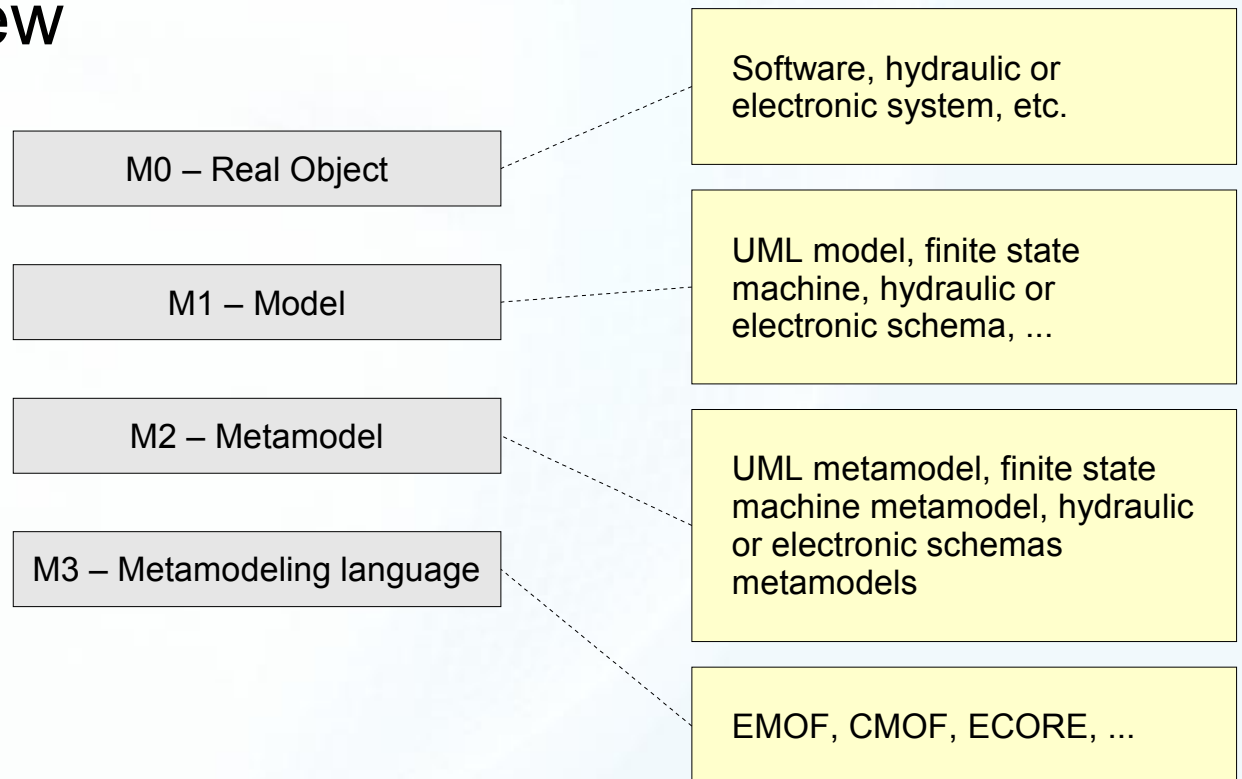
Numerous Formalisms

Scade/LUSTRE, MatLab/Simulink, ESTEREL, SDL, Structured analysis, Software real time architectures (ARINC653, bare hardware), Mealy machines, HOOD, UML, SysML, AADL, experimental languages...



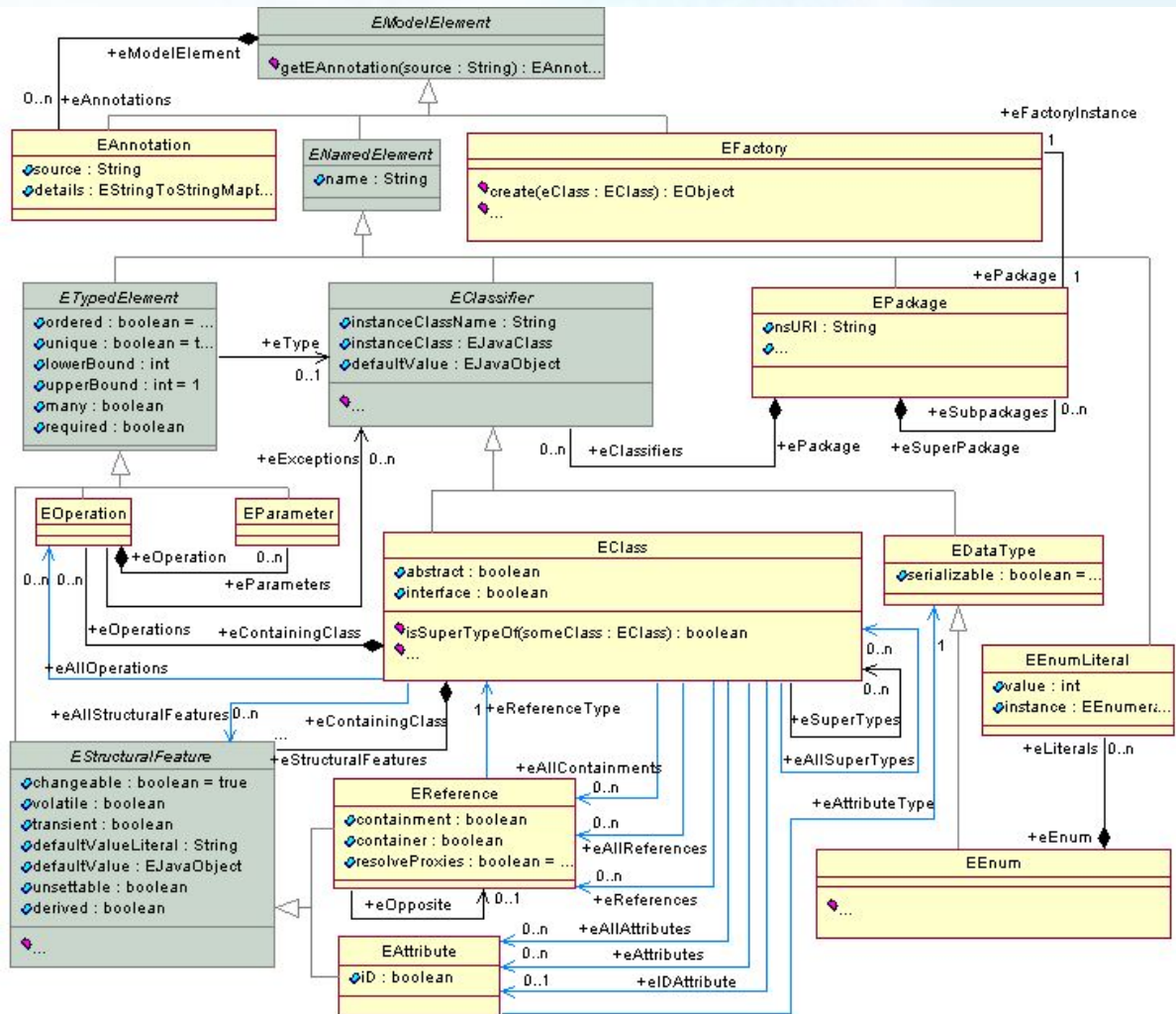
Meta modeling Technique

- A model type or a language can be described using a meta model or a meta language
- The Open Modeling Group (OMG) defined a 4-layers point of view



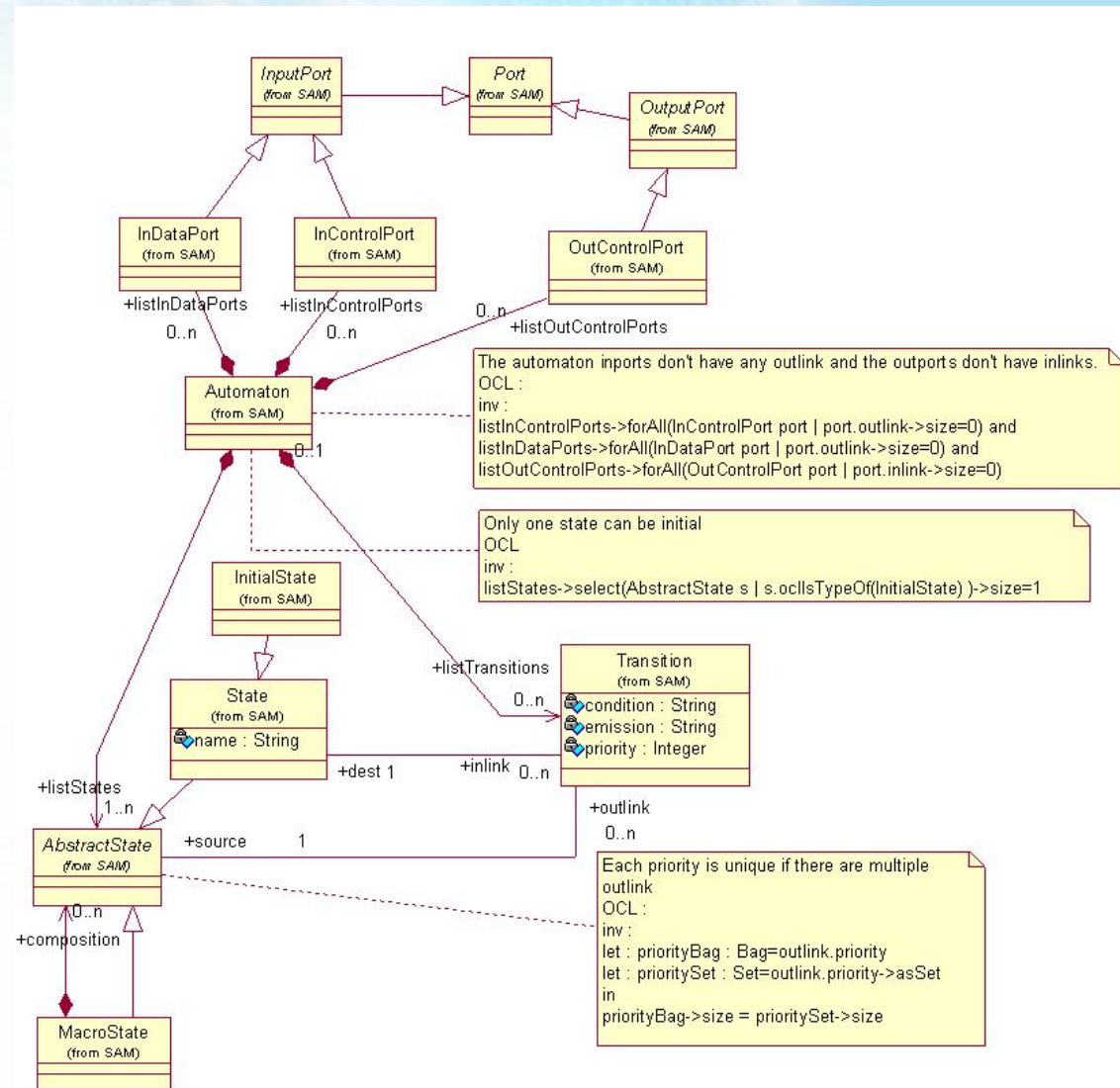
Meta Object Facility

- MOF is the meta modeling language standardized by the OMG.
- TOPCASED uses Ecore, which is the EMF variant of MOF.



Example of meta model

A home-made finite automaton meta model



Required Operators

- Comparison of models
- Merge of models
- Meta-model update for a model

The screenshot displays two Eclipse IDE windows. The top window, titled 'Java Structure Compare', shows a tree view of a 'Compilation Unit' containing 'Import Declarations', 'IPlatformConfiguration', and 'ConfiguratorUtils'. The bottom window, titled 'Java Source Compare', compares two Java source files: 'org.eclipse.update.configurator/src...onfigurator/ConfiguratorUtils.java' and 'org.eclipse.update.configurator/src...rator/IPlatformConfiguration.java'. The left pane shows the source code for ConfiguratorUtils.java, including package and import statements. The right pane shows the source code for IPlatformConfiguration.java, including package and import statements. The code is color-coded, with keywords like 'package' and 'import' in red.

```
org.eclipse.update.configurator/src...onfigurator/ConfiguratorUtils.java
*   IBM Corporation - initial API and
*****
package org.eclipse.update.configurator;

import java.io.*;
import java.net.*;

org.eclipse.update.configurator/src...rator/IPlatformConfiguration.java
*   IBM Corporation - initial API and
*****
package org.eclipse.update.configurator;

import java.io.IOException;
import java.net.URL;
```

Existing Meta-modeling Tools

- DoME from Honeywell
- OpenTool from TNI-Valiosys
- GME from the Vanderbilt University
- EMF from the Eclipse Foundation
- TopModL, a research project leaded by the Université de Haute Alsace
- ...

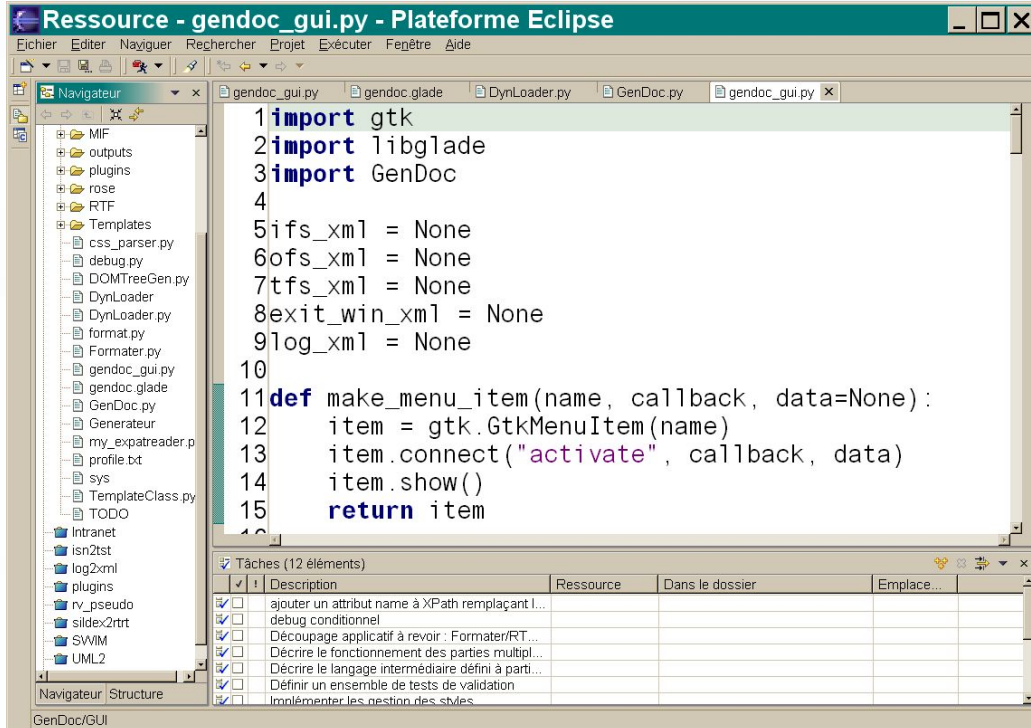
TOPCASED

Toolkit In OPen source for Critical Applications & SystEms Development

- General presentation of the project
- A meta-modeling toolset
- **The toolset architecture**
- Services & Formats
- Demo / screenshots
- Conclusion

Eclipse, EMF & GEF

- A Java based multi-platform IDE
- Fully open source (CPL/EPL)
- Several major industrial use and develop it (IBM, Thales, ...)
- GEF : Graphical Editing Framework
- EMF : Eclipse Modeling Framework

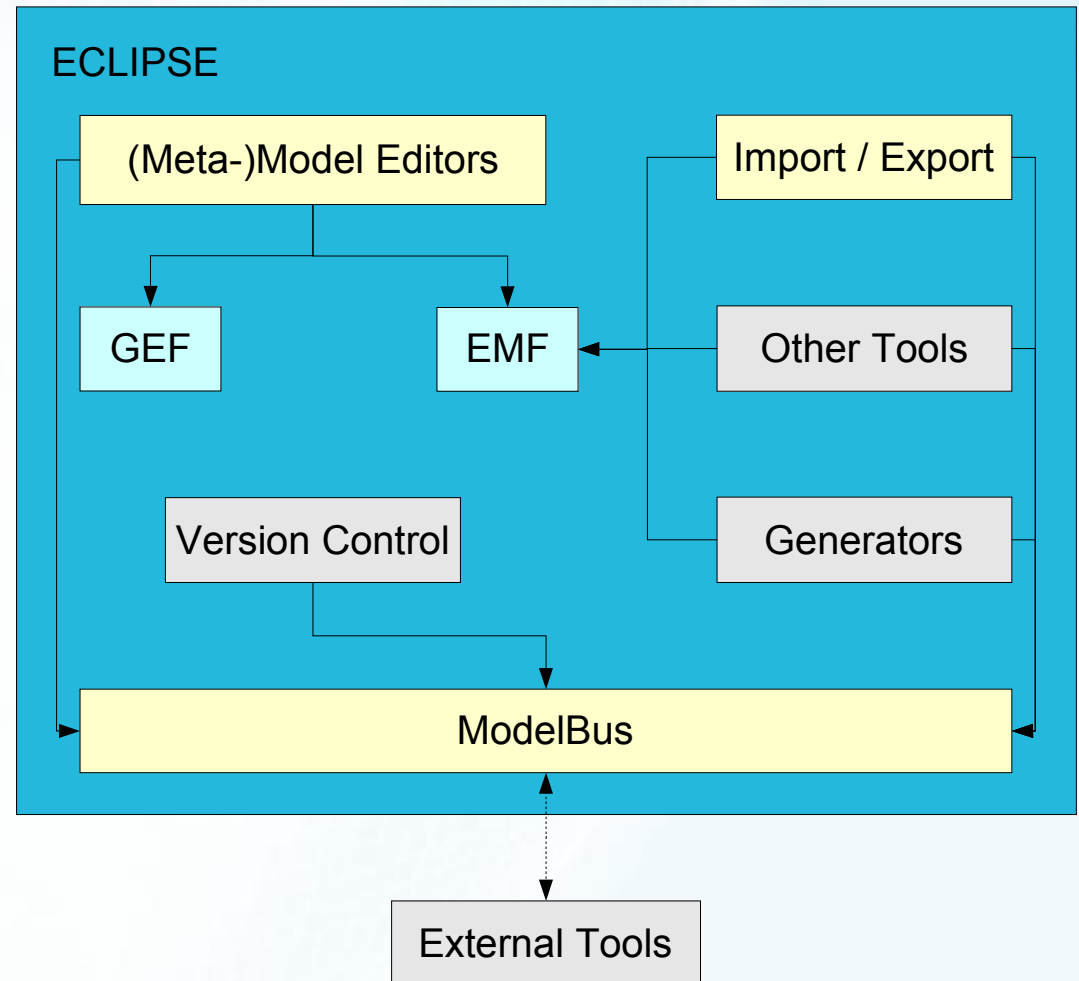


```
1 import gtk
2 import libglade
3 import GenDoc
4
5 ifs_xml = None
6 ofs_xml = None
7 tfs_xml = None
8 exit_win_xml = None
9 log_xml = None
10
11 def make_menu_item(name, callback, data=None):
12     item = gtk.GtkMenuItem(name)
13     item.connect("activate", callback, data)
14     item.show()
15     return item
```

Tâches (12 éléments)			
	Description	Ressource	Dans le dossier
<input checked="" type="checkbox"/>	ajouter un attribut name à XPath remplaçant l...		
<input checked="" type="checkbox"/>	debug conditionnel		
<input checked="" type="checkbox"/>	Découpage applicatif à revoir : Formater/RT...		
<input checked="" type="checkbox"/>	Décrire le fonctionnement des parties multipl...		
<input checked="" type="checkbox"/>	Décrire le langage intermédiaire défini à parti...		
<input checked="" type="checkbox"/>	Définir un ensemble de tests de validation		
<input checked="" type="checkbox"/>	Implémenter la gestion des styles		

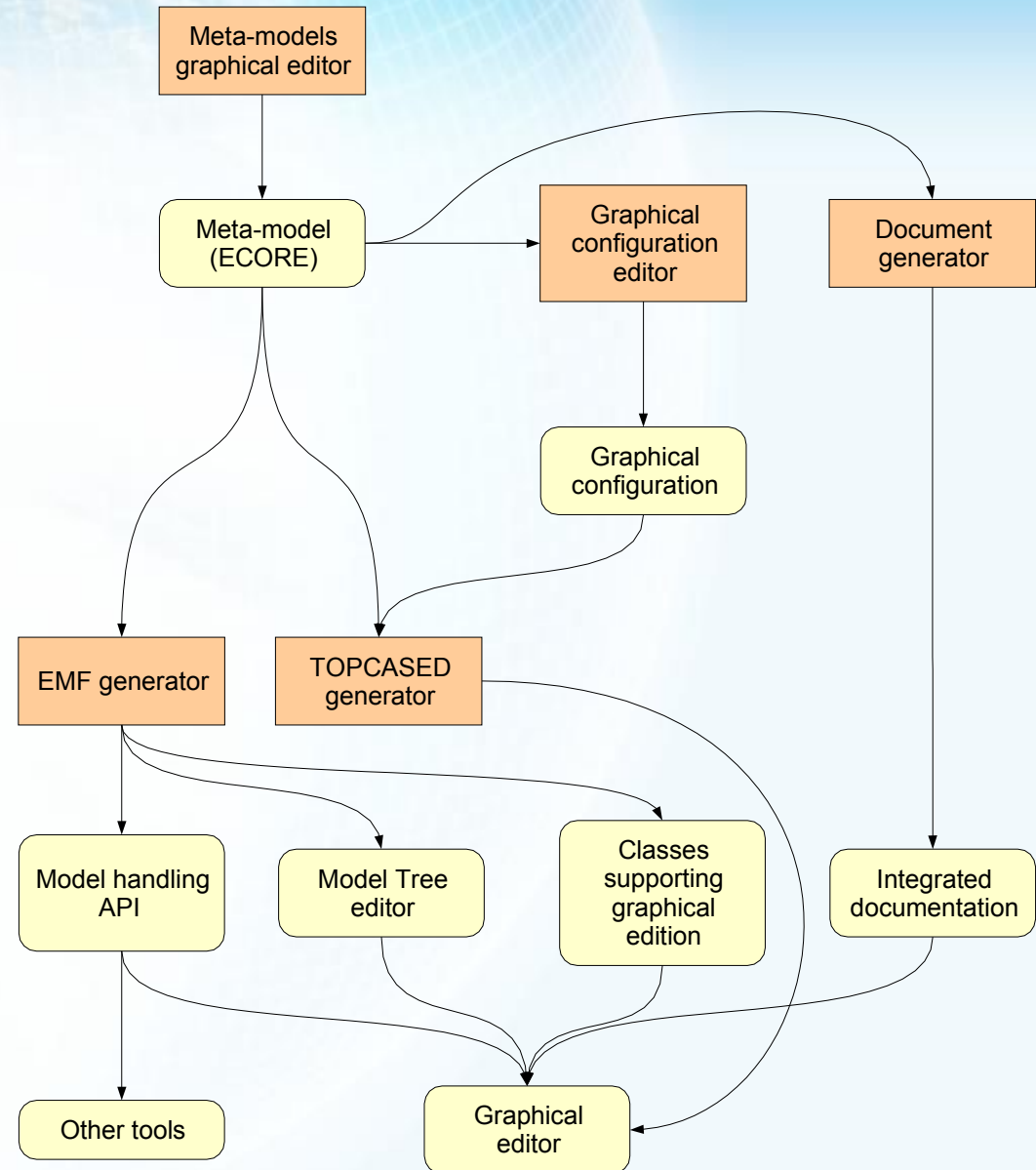
Workbench Architecture

- Services distributed as Eclipse plug-ins and features
- Services can be found and used through the bus
- External tools can also be plugged (useful to avoid licenses conflicts)



Development of graphical editors

- Partially generated
- From meta-models and a specific graphical configuration model



TOPCASED

Toolkit In OPen source for Critical Applications & SystEms Development

- General presentation of the project
- A meta-modeling toolset
- The toolset architecture
- **Services & Formats**
- Demo / screenshots
- Conclusion

Transformations & Formats

- through EMF, models are read and written in the XMI 2.0 format
- Others formats are possible (e.g. aaxl from AADL/OSATE)
- ATL (INRIA AMMA project, released in Eclipse/GMT) is used for import/export
- It is of course always possible to use more general languages to implement transformations (Java, C, C++, etc.)

Verifying Models

- Simple semantic rules : static checkers (OCL, ...)
- a posteriori : when the designer decides to do so
- a priori : during edition
- Behavioral checks : requires powerful tools
- Display the check result at the graphical level

Simulation

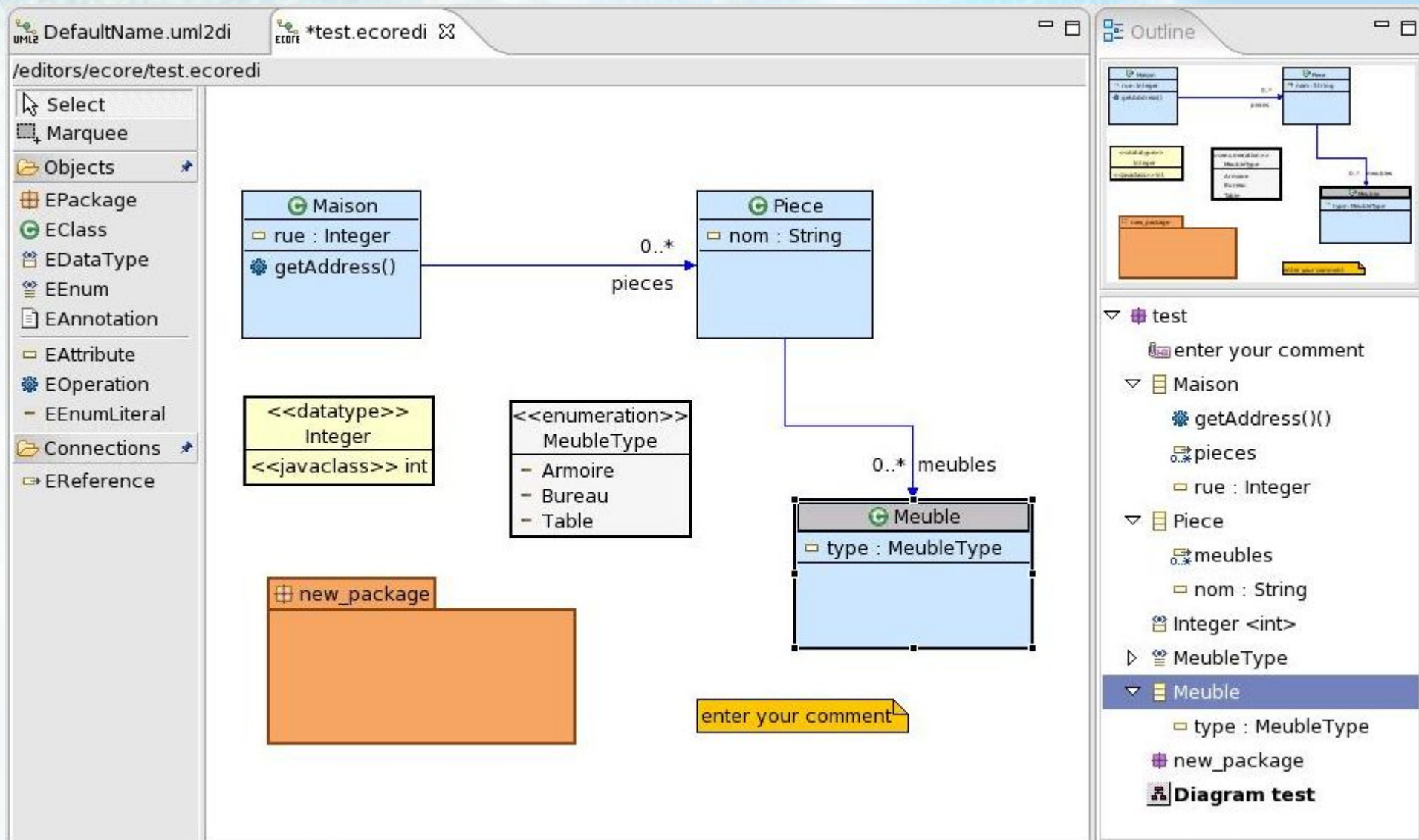
- A good tool to ensure the understanding of projects particularities, and to validate models behaviors
- Can be used to implement non-regression tests on models
- First studies are going to start

TOPCASED

Toolkit In OPen source for Critical Applications & SystEms Development

- General presentation of the project
- A meta-modeling toolset
- The toolset architecture
- Services & Formats
- **Demo / screenshots**
- Conclusion

ECORE



Functional split & Automata

The screenshot displays the Eclipse Platform interface for Topcased Modeling. The main window shows a state machine diagram for the project 'spec-appli.samdi'. The diagram consists of several states and transitions:

- repos**: A state with a multiplicity of 1. It has a transition labeled 'init finished' leading to the 'attente' state.
- attente**: A state that receives the 'init finished' transition from 'repos' and has a transition leading to the 'réception' state.
- réception**: A composite state containing two sub-states: 'wait ack' and 'wait data'. It receives a transition from 'attente' and has a transition leading to the 'fin de transmission' state.
- fin de transmission**: A state that receives a transition from 'réception' and has a transition leading back to the 'repos' state.

The interface includes a Navigator on the left showing the project structure, an Outline window, and a Properties window at the bottom right. The Properties window shows the following details:

Property	Value
Comment	
Name	
Requirements	
Uid	1117525762771

AADL/COTRE

Topcased Modeling - deadlock.aadlcotre - Eclipse Platform

File Edit Navigate Search Project Run Window Help

Update Site Action

Topcased Mo...

spec-appli.samdi deadlock.aadlcotre

/models/deadlock.aadlcotre

Select
Marquee
Objects
System Instance
Property Set
Property
Types
Implementations

S System
P Process
D Data
T Thread
G ThreadGroup

S s
S s.S
P Proc
P Proc.pp
T TT
T TT.tt

P Proc
P taille
P poids
A
A
G
G

Properties Problems Error Log

Property	Value
Identifier List	processor Proc.pp
Refined	false
Sub Component Access Classifier	

models

- configurator.ecore
- deadlock.aadlcotre
- deadlock.aadlcotre
- fsm.ecore
- fsm.ecoredi
- models.services
- mon_appli.uml2
- mon_appli.uml2di
- mon_beau_modele.aadlcotre
- mon_beau_modele.aadlcotre
- spec-appli.sam
- spec-appli.samdi
- org.topcased.aadlcotre

Outline

AADL Specification

AADL 1.0

Topcased Modeling - my_architecture.aaxldi - Eclipse Platform

File Edit Refactor Navigate Search Project Run Window Help

Java Topcased Mo... C/C++ Resource

100%

my_architecture.aaxldi

/Demo/AADL/my_architecture.aaxldi

AADLSpec Diagram : null / my_architecture

Diagram elements:

- AadPackage1 (Package): PRIVATE, PUBLIC
- PropertySet1 (Property Set): PropertyType1, PropertyDefinition1, PropertyConstant1
- SystemType1 (System)
- SystemType1.Sys... (System)
- DeviceType1 (Device)
- DataType1 (Data)
- DataType1.DataI... (Data)

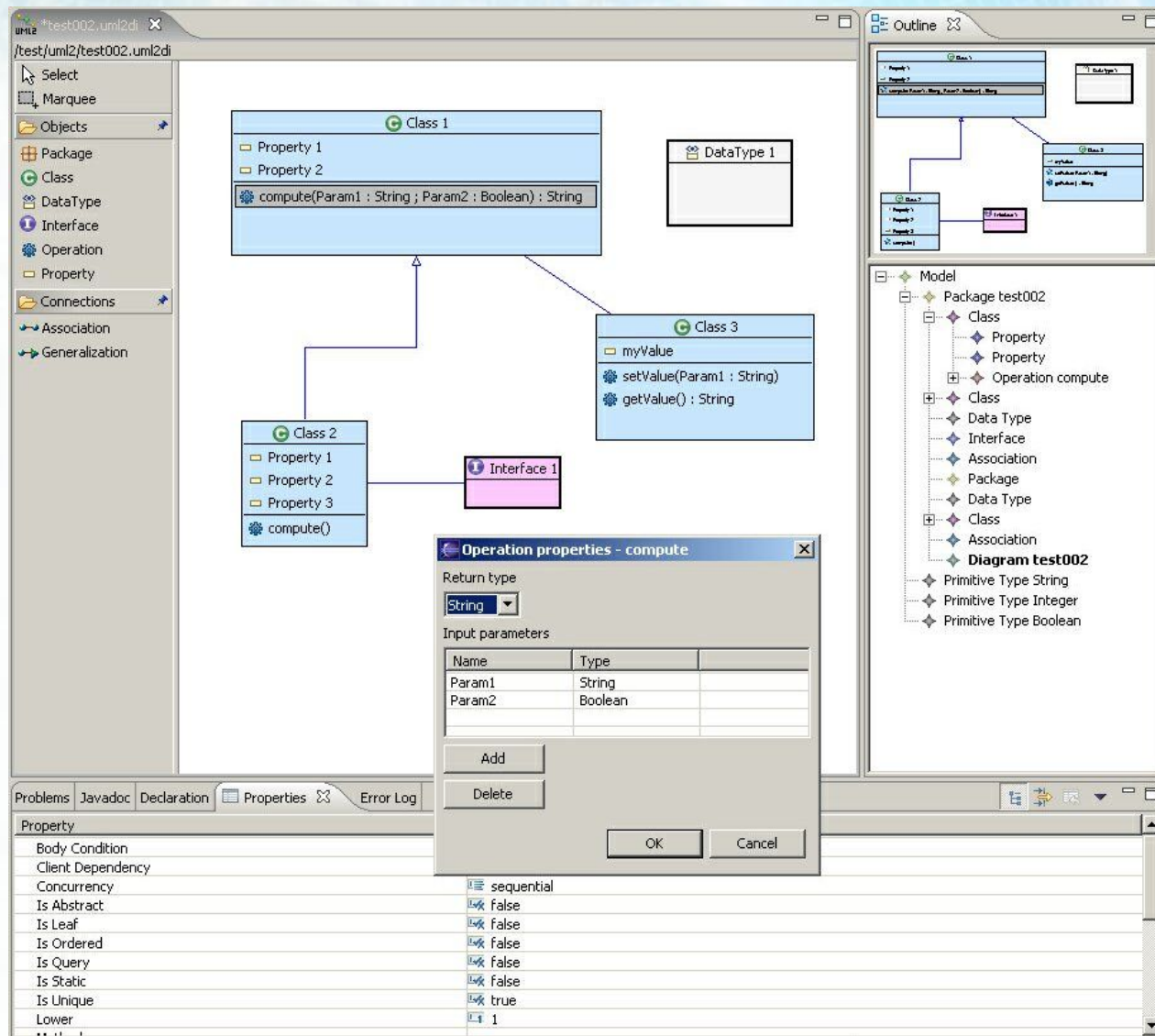
Relationships:

- SystemType1 <<Realizes>> SystemType1.Sys...
- DeviceType1 <<Realizes>> SystemType1.Sys...
- DataType1 <<Realizes>> DataType1.DataI...
- SystemType1 <<Extends>> DataType1

Properties:

Property	Value
Comment	
Name	

UML 2.0



To Conclude

- The software developed is of course freely available (current release : 0.5.0)
- First experiments will begin shortly
- The partnership is open to every goodwills
- Keep in touch with TOPCASED at

<http://www.topcased.org/>