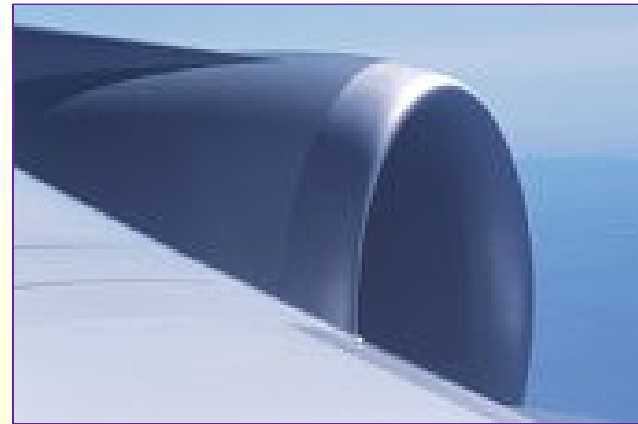




OpenEmbeDD as an MDE platform

A powerful set of tools
to improve your DSL work



Speaker: Vincent Mahé

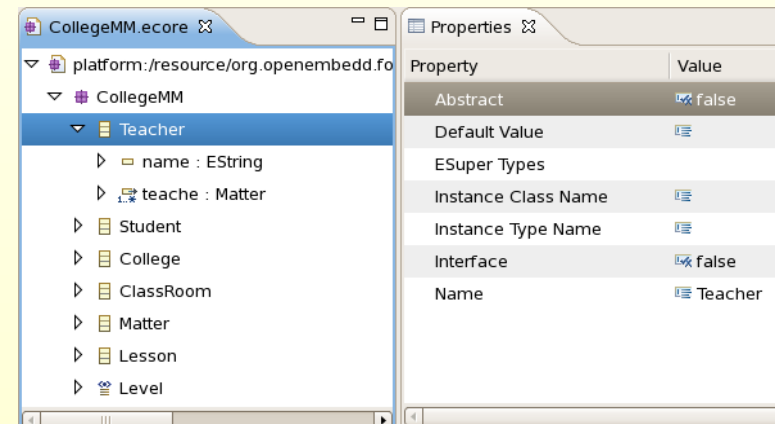
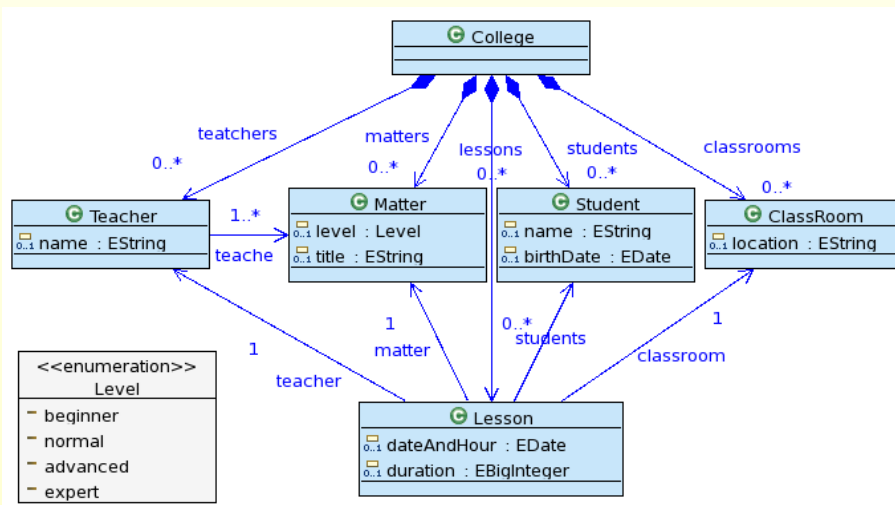


What is OpenEmbeDD?

- *A IDE targeting Real-Time and Embedded software*
- A set of powerful M.D.E. tools
- An easy-to-install platform
- The support of industrial standards
- An Open Source platform
- For any Domain Specific Language
- Now YOUR daily M.D.E. environment

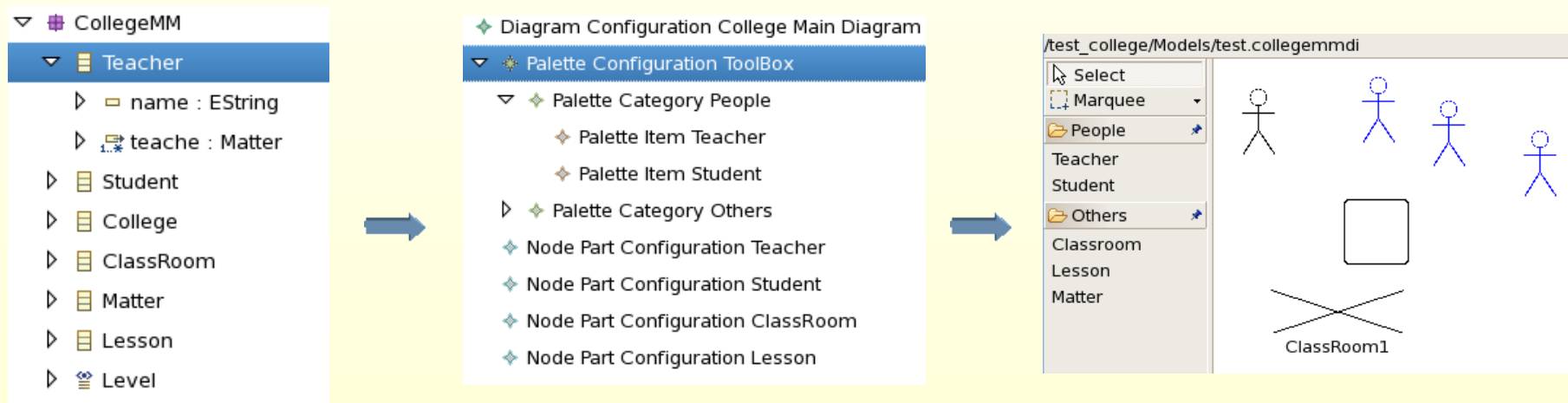
A set of powerful M.D.E. tools

- The Eclipse metamodeling tools
 - EMF modeling framework
 - Ecore modeler for easy metamodel definition
 - GMF graphical library to design your own modelers
 - ...



A set of powerful M.D.E. tools

- **The Topcased generator**
 - From a given EMF metamodel ...
 - Configure the generator
 - ... and get your own modeler

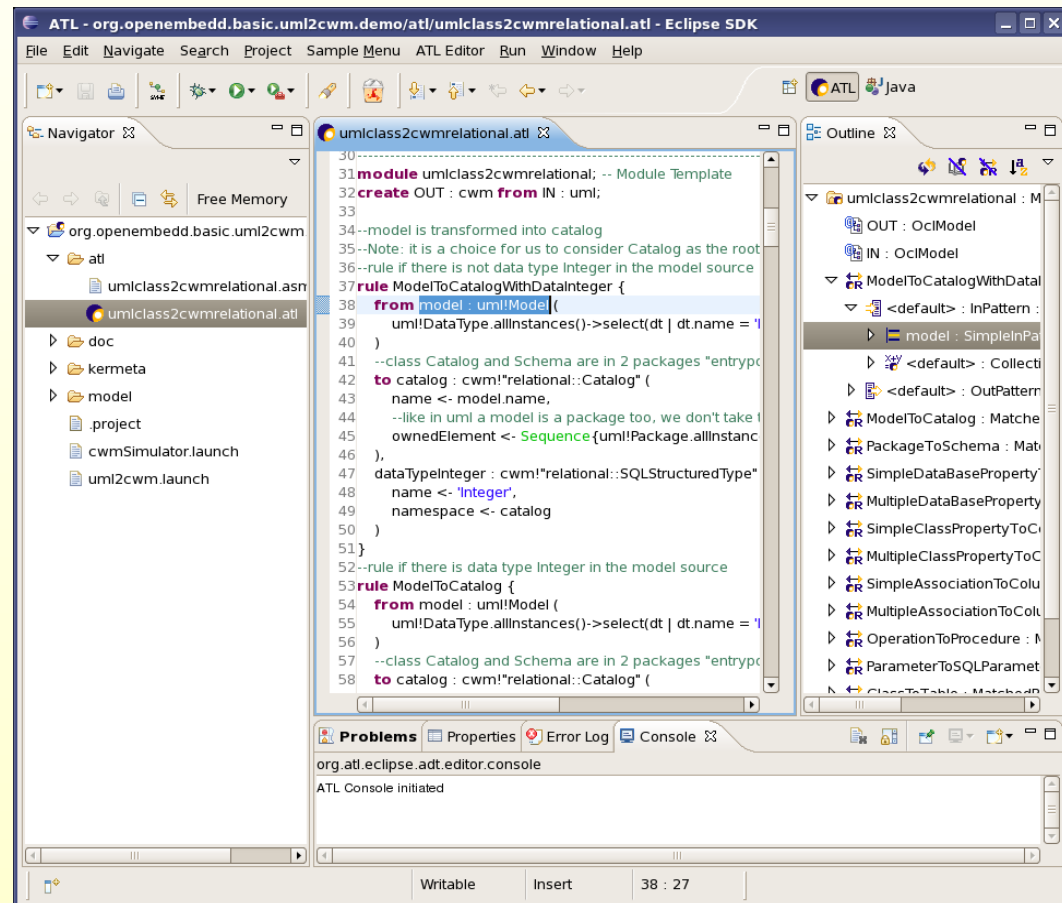


- **With powerful features**
 - Multiple diagrams for one model (use case & classes & states &)
 - Hierarchical diagrams (composite classes, state machine of a class)

A set of powerful M.D.E. tools

- The **ATL transformation language**

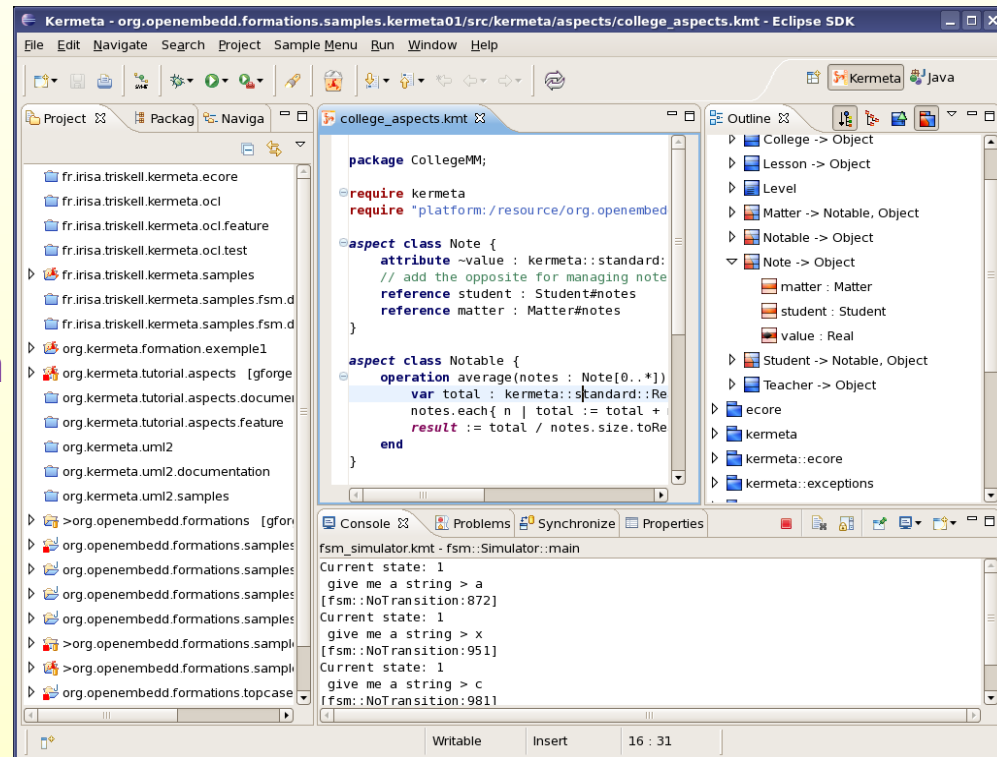
- A functional language
- Helpers system to write libraries
- Main part of the M2M Eclipse project



A set of powerful M.D.E. tools

- **The Kermeta environment**

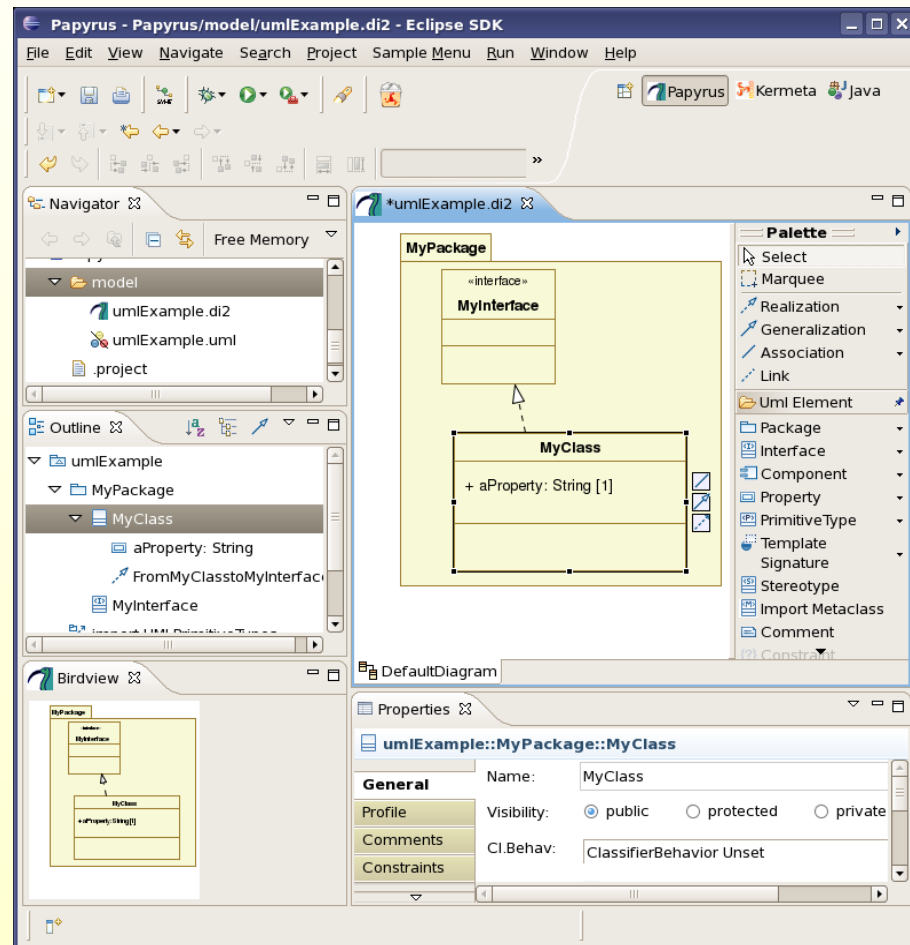
- An Object Oriented language with its IDE (completion, debugging,...)
- Aspects to weave behaviour into metamodels
- A graphical editor to design DSL metamodels
- Ability to enhance, transform and simulate models



A set of powerful M.D.E. tools

The Papyrus UML profile editor

- Full respect of the UML2 OMG standard
- Full respect of the DI2 (Diagram Interchange) standard
- Extendable architecture that allows to add new diagrams, etc.
- Profile development support facilities for UML2 profiles
- Nested profiles support



An easy to install platform

- **Bundles for Linux, Windows and Mac OS X**
 - Include the Eclipse SDK
- **Full plug-ins pack for your already installed Eclipse**
 - Generic modelling tools
 - Real Time & Embedded tools
 - Useful examples
- **An update site to get a finely tuned platform, with only selected tools you need**
- **A selection of useful tools for developers**
 - Subclipse plug-ins
 - OpenOffice integration to Eclipse
 - OpenEmbeDD development tools

An easy to install platform

- **And all those packages are stored for each version**
 - You always can re-open old models in 10 years
 - Transformations written on a version can always be ran
=> your earlier efforts and attempts are not loose

All versions

	Bundles			Plugins			Update sites		DEV plugins	Sources
										
Experimental	tar.gz	zip	tar.gz	zip	zip	zip	site	-	zip	-
Japet (0.4.0)	tar.gz	zip	tar.gz	zip	zip	zip	site	zip	zip	site
0.3.0	tar.gz			zip			site	zip	zip	site
0.2.0	zip			zip			site	zip	-	site
0.1.0	zip			-			site	zip	-	site

Support of industrial standards

- Complete UML 2.1 metamodel & diagrams
- OMG Marte UML profile for Real-Time design
- Industrial standards with their modelers
 - AADL for aeronautic design
 - SysML models
 - SDL models
- Papyrus modeler for UML profiles

An Open Source platform

- **Based on the Eclipse development platform**
 - A well-known Integrated Development Environment
 - World-wide industrial use & support
- **All packaged tools are Open Source**
- **Availability of sources**
 - Eclipse source plug-ins for libraries
 - Zipped sources for OpenEmbeDD tools
- **Mainly Eclipse Public License**
- **Public source forge for OpenEmbeDD developments**
 - <http://gforge.inria.fr/openembedd>
- **A repository of large sample models**
 - <http://gforge.inria.fr/openembedd-rep>

For any Domain Specific Language

- Design your metamodel(s) with the Ecore modeler
- Generate the editor(s) with EMF
- Set diagram modelers with Topcased
 - Multiple diagrams in the same model
- Write transformations in ATL
- Enhance metamodel with behaviour in Kermeta
- Run simulations of models with Kermeta



Now YOUR daily M.D.E. environment

- **Easily install the platform**
 - <http://openembedd.org/Download>
- **Welcome into Eclipse EMF world**
 - Enter your DSL metamodels with Ecore
 - Generate corresponding editors
- **Easily define tools for your DSLs**
 - Pass models from one metamodel to another one
 - Simulate those models
 - Compute models to extract informations
 -

and then, forget what was your work before OpenEmbeDD :-)

See OpenEmbeDD in action in the hall