A first experimentation on high-level tooling support upon Fractal

Frédéric Loiret[1,2], David Servat[1], Lionel Seinturier[2]

[1]CEA-LIST - L-LSP/Accord
[2]INRIA Futurs - GOAL/Jacquard

5th Fractal Workshop

PhD context

- **INRIA Jacquard Team**
  - Software components and AOP
- **CEA LLSP Team**
  - Model-based approaches (UML) to design real time and embedded systems

- **Component paradigm for RT**
  - Fractal component model extensions
- **Modelisation**
  - « High-level » tooling support
- **Execution**
  - Think implementation as a destination platform

First experiments

- **Meta-modelisation for Fractal architectural concepts**
  - Eclipse Modeling Framework overview
  - Fractal / Think ADL meta-model subset
  - Think V2 tools front end

- **Reverse engineering tool**
  - Source code analysis for Think components
  - Behavior extraction

Eclipse Modeling Framework overview

- **A modeling and code generation framework for developing meta-model based applications and design tools**

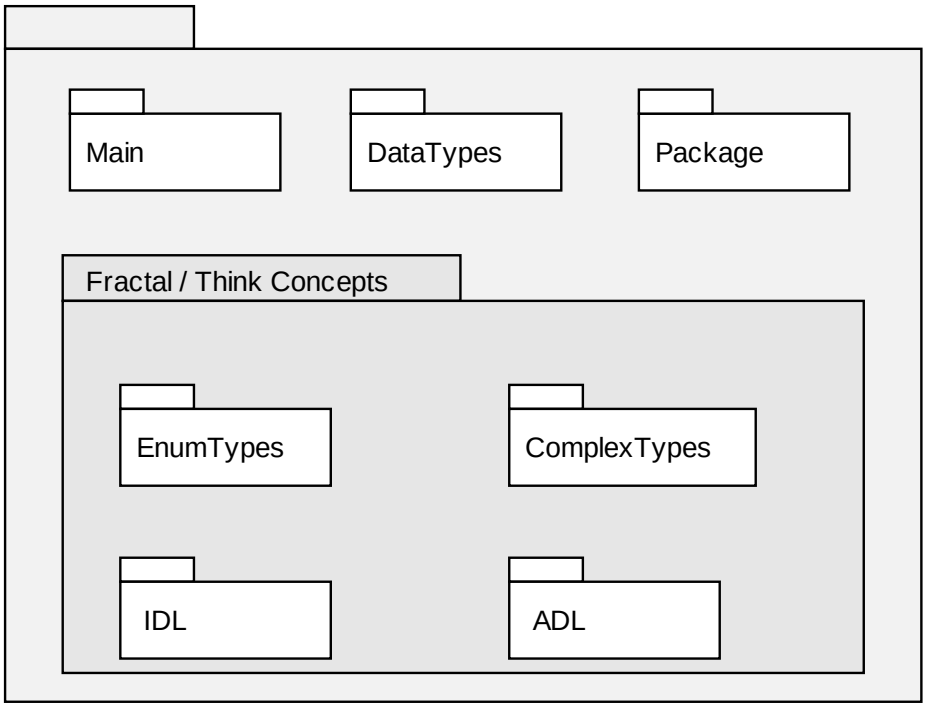
- **A java implementation of the MOF (OMG) : EMF Ecore**
  - class-relation constructs meta-model

- **Code generation :**
  - Java API (and classes) for model instance manipulations
  - Adaptater classes (generated EMF model editor as Eclipse plugin)

- **XMI serialization for persistence support**
  - (Model transformations)

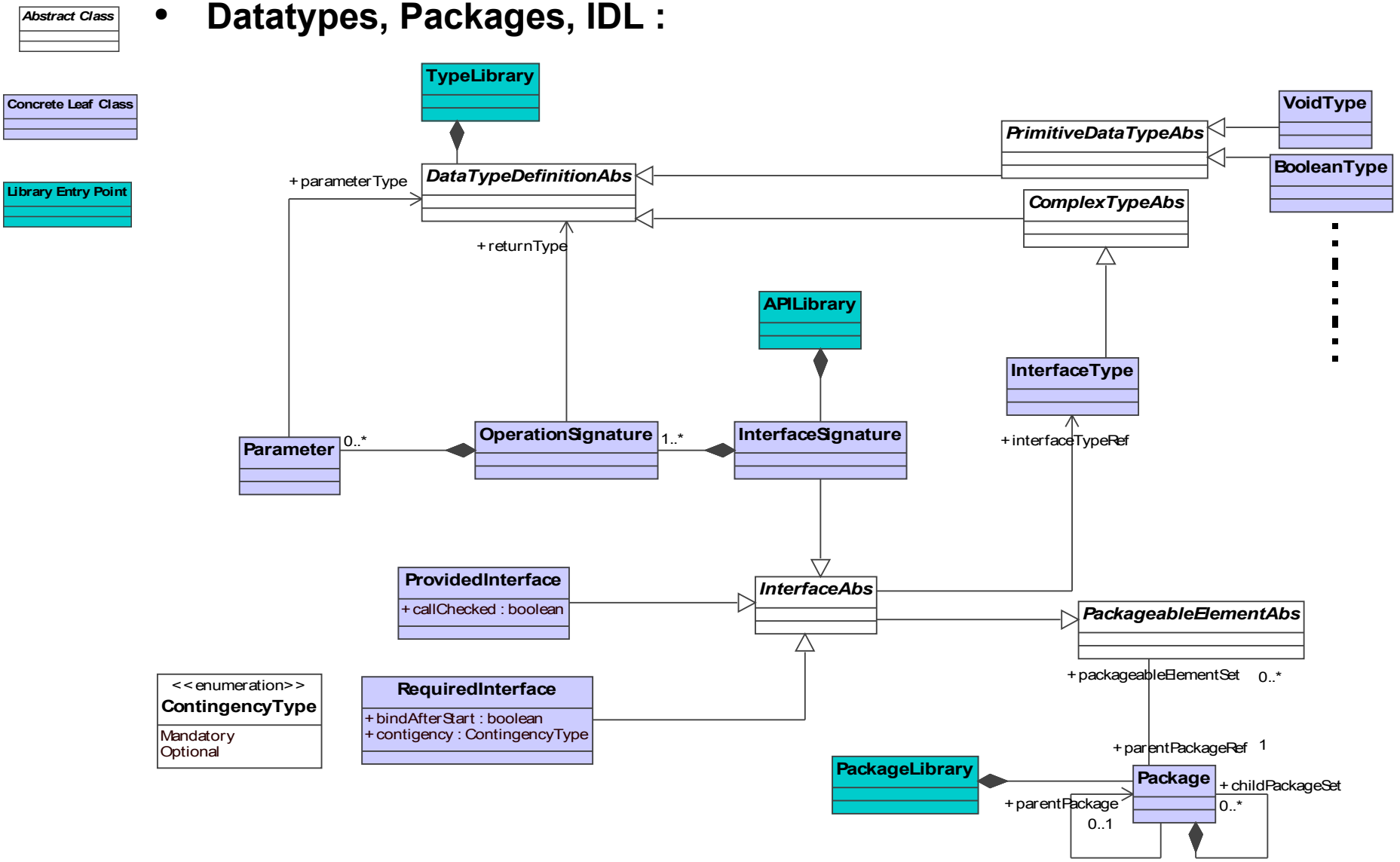
Fractal / Think ADL meta-model (1)

- **EMF meta-model definition « from scratch »**
  - ADL and IDL Abstract Syntax Trees as MM
  - Hierarchy of interrelated concrete and abstract Ecore classes
  - Two types of associations :
    - containments
    - references
- **Packages organization**



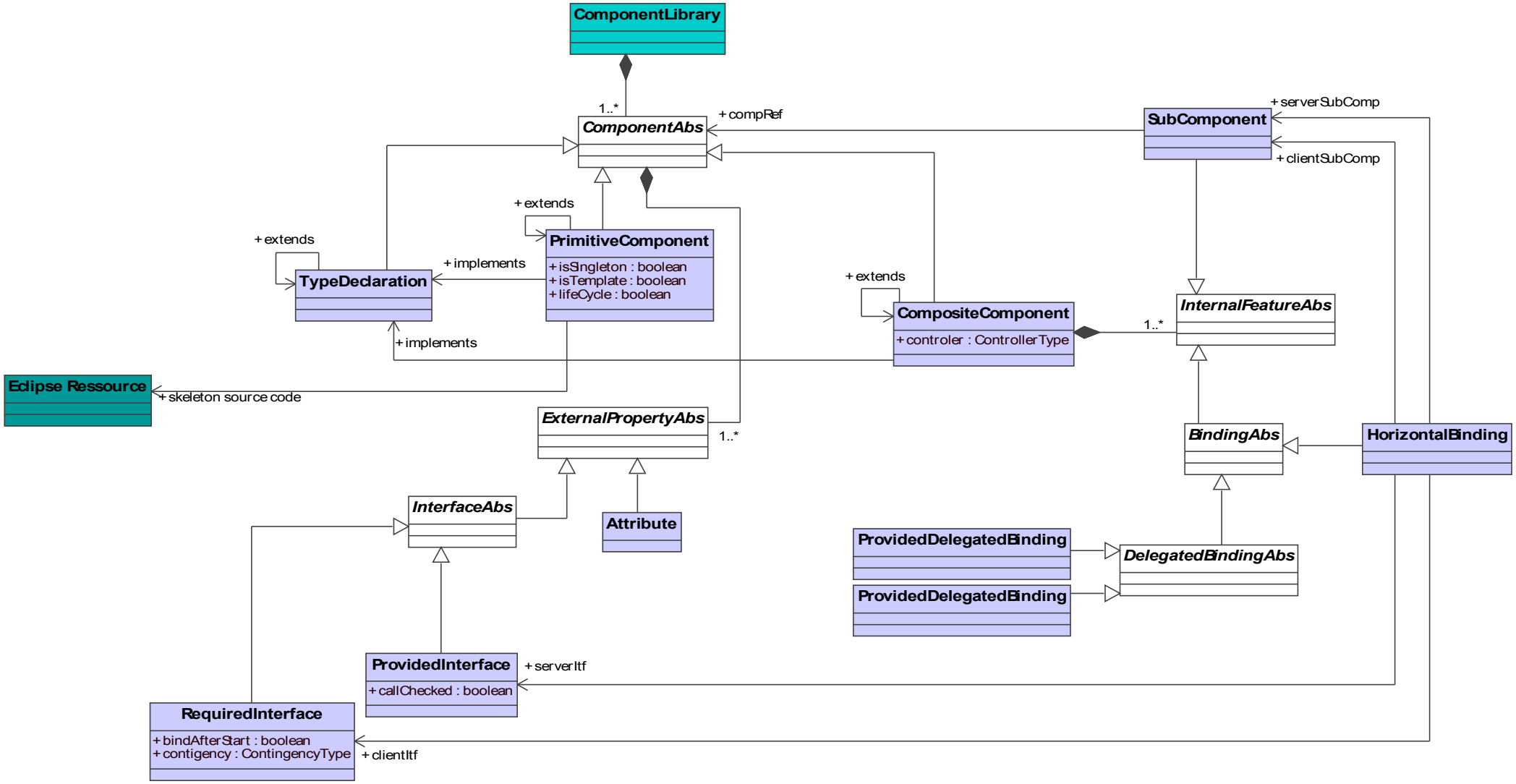
Fractal / Think ADL meta-model (2)

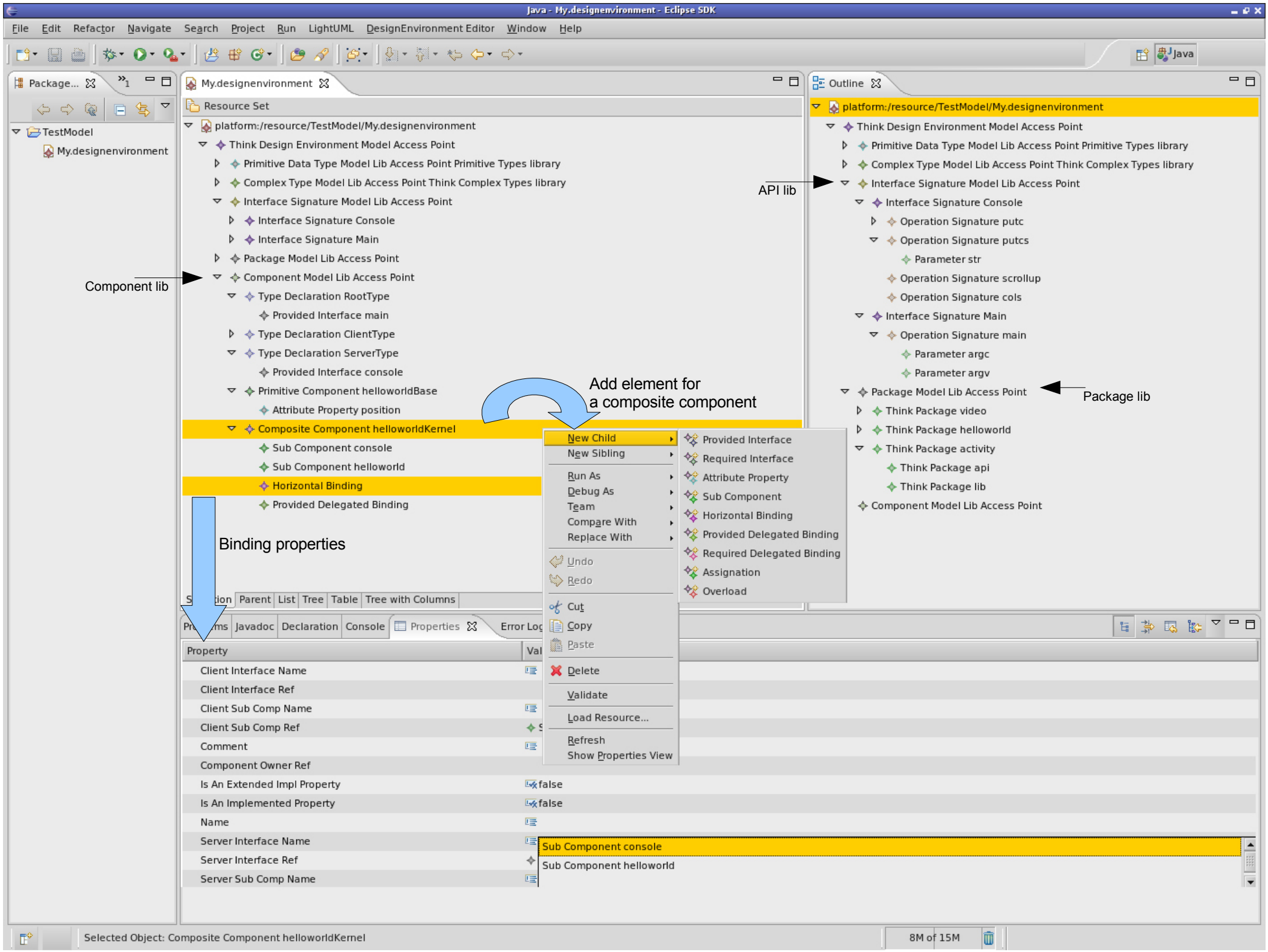
- **Datatypes, Packages, IDL :**



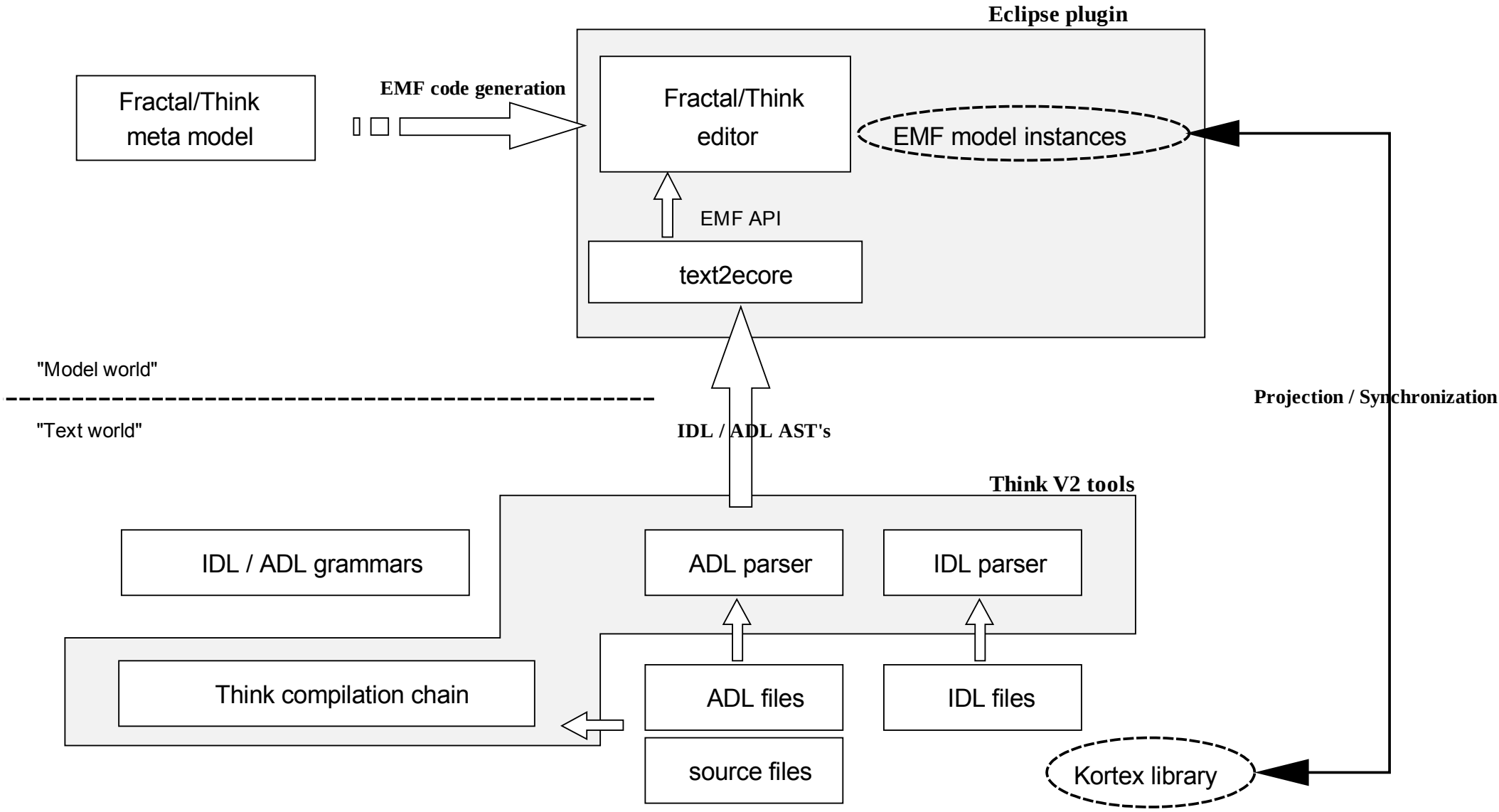
Fractal / Think ADL meta-model (3)

- **Simplified ADL meta-model :**




Java - My.designenvironment - Eclipse SDK
File Edit Refactor Navigate Search Project Run LightUML DesignEnvironment Editor Window Help
Package...
TestModel
My.designenvironment
My.designenvironment
Resource Set
platform:/resource/TestModel/My.designenvironment
Think Design Environment Model Access Point
Primitive Data Type Model Lib Access Point Primitive Types library
Complex Type Model Lib Access Point Think Complex Types library
Interface Signature Model Lib Access Point
Interface Signature Console
Interface Signature Main
Package Model Lib Access Point
Component Model Lib Access Point
Type Declaration RootType
Provided Interface main
Type Declaration ClientType
Type Declaration ServerType
Provided Interface console
Primitive Component helloworldBase
Attribute Property position
Composite Component helloworldKernel
Sub Component console
Sub Component helloworld
Horizontal Binding
Provided Delegated Binding
Component lib
Binding properties
Add element for
a composite component
New Child
New Sibling
Run As
Debug As
Team
Compare With
Replace With
Undo
Redo
Cut
Copy
Paste
Delete
Validate
Load Resource...
Refresh
Show Properties View
Provided Interface
Required Interface
Attribute Property
Sub Component
Horizontal Binding
Provided Delegated Binding
Required Delegated Binding
Assignation
Overload
Parent List Tree Table Tree with Columns
Javadoc Declaration Console Properties
Property
Client Interface Name
Client Interface Ref
Client Sub Comp Name
Client Sub Comp Ref
Comment
Component Owner Ref
Is An Extended Impl Property
Is An Implemented Property
Name
Server Interface Name
Server Interface Ref
Server Sub Comp Name
false
false
Sub Component console
Sub Component helloworld
Outline
platform:/resource/TestModel/My.designenvironment
Think Design Environment Model Access Point
Primitive Data Type Model Lib Access Point Primitive Types library
Complex Type Model Lib Access Point Think Complex Types library
Interface Signature Model Lib Access Point
API lib
Interface Signature Console
Operation Signature putc
Operation Signature putcs
Parameter str
Operation Signature scrollup
Operation Signature cols
Interface Signature Main
Operation Signature main
Parameter argc
Parameter argv
Package Model Lib Access Point
Package lib
Think Package video
Think Package helloworld
Think Package activity
Think Package api
Think Package lib
Component Model Lib Access Point
Selected Object: Composite Component helloworldKernel
8M of 15M

Think V2 tools front-end

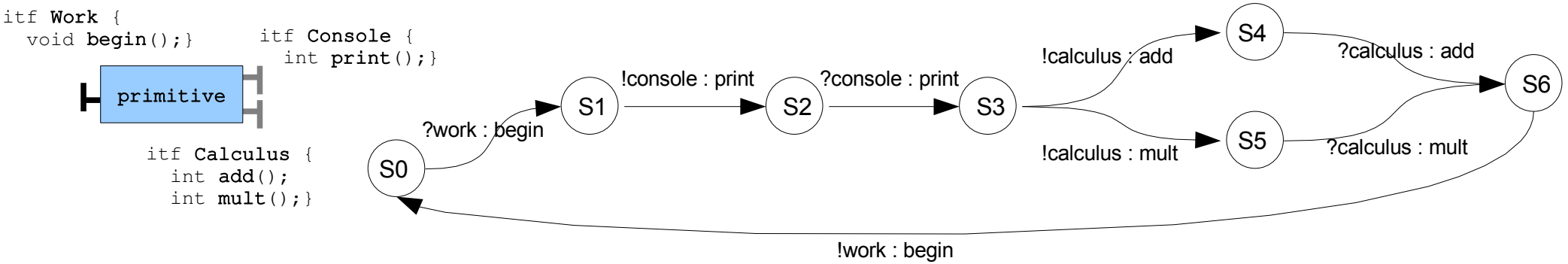


Discussions

- **Homogeneous design environment for the user to specify Fractal applications**
  - « Eclipse-based »
- **Simpler for the user**
  - Intuitive constructs using the editor
  - All meta-model concepts are « typed »
  - Semantical constraints based on meta-model structure
  - Resolv references during application specification
- **Extensions**
  - Graphical layer using GEF above EMF
  - Fractal EMF meta-model to UML2 profile
  - Expand the meta-model (≡ ADL extension)
    - Provide annotations mechanism for non-functional properties on model entities
    - Component behaviour

Toward a behaviour extraction and analysis

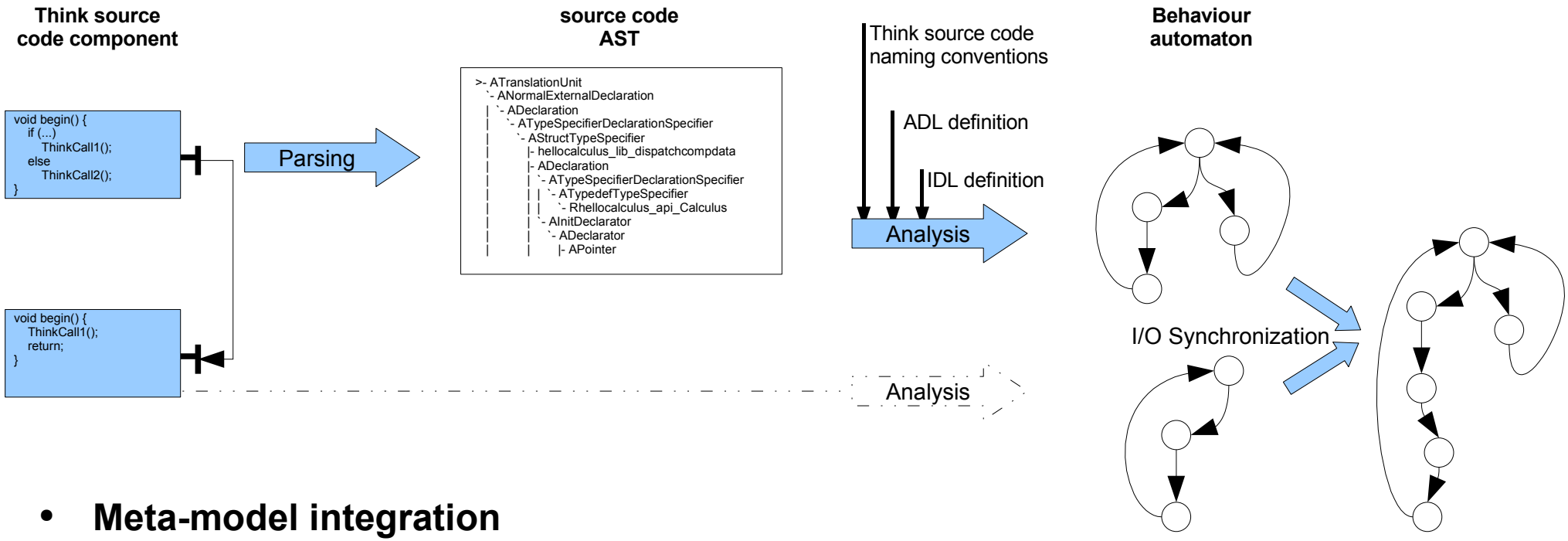
- **Inspired from SafeArchie [1]**
  - Behaviour contracts specified by I/O automata (messages traces exchanged by components with their environment).



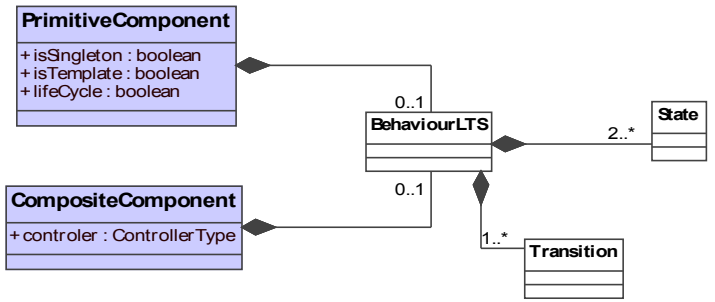
- **Bottom-up approach :**
  - Extract behaviour from Think source code components
  - Integrate this behaviour to the meta-model

[1] **O. Barais**. *Construire et Maitriser l'Evolution d'une Architecture Logicielle à base de composants*. PhD Thesis. 2005.

• Reverse engineering tool chain



• Meta-model integration



automata2ecore

– Behaviour properties verification during user specification