	AM3 USE CASE EXAMPLE "PET STORE"	Guillaume Doux Guillaume.doux@inria.fr
ATLANMOD	Use Case Description	Date : 2009/03/31

1. General Description

"Pet Store" is a complex and complete use case [1], part of the AM3 component [4] use cases, that has been designed and developed in order to manage traceability and navigability between models during a development process. This problem can be divided into two complementary sub-problems corresponding to different abstraction levels:

- The traceability/navigability between models (higher level);
- The traceability/navigability between model elements (lower level).

The approach uses a combination of megamodeling [4] and model weaving [5] to manage these two abstraction levels. In the developed approach we consider that the navigability between model elements can be viewed as a refinement of the navigability between models. The model level navigation is managed using links contained in the megamodel. The navigability between model elements uses links stored in weaving models corresponding to the megamodel links.

Several artifacts have been developed for this use case. They represent a snapshot in a simplified development of a "Pet Store" application [2]. Nine metamodels and the corresponding models have been developed to cover all the development process from the requirements to the deployment of the application. There are also several weaving models that provide navigable model element link. All these artifacts are managed using a megamodel in AM3.

The following figure presents an excerpt of the contents of the megamodel containing all these artifacts and some links between these artifacts:

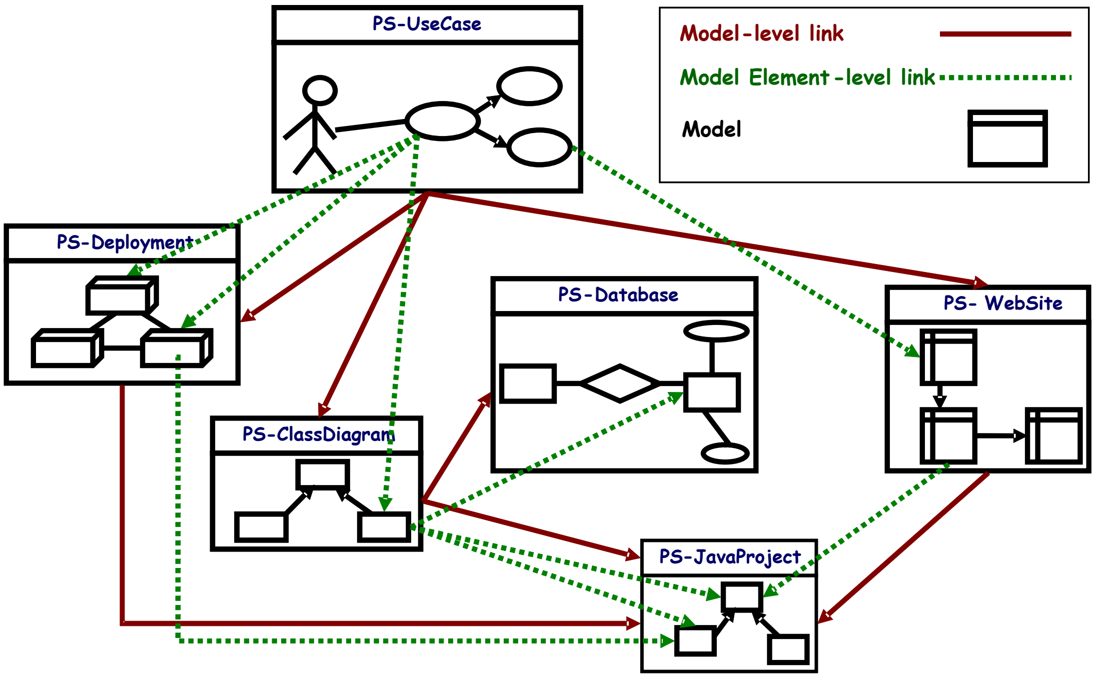
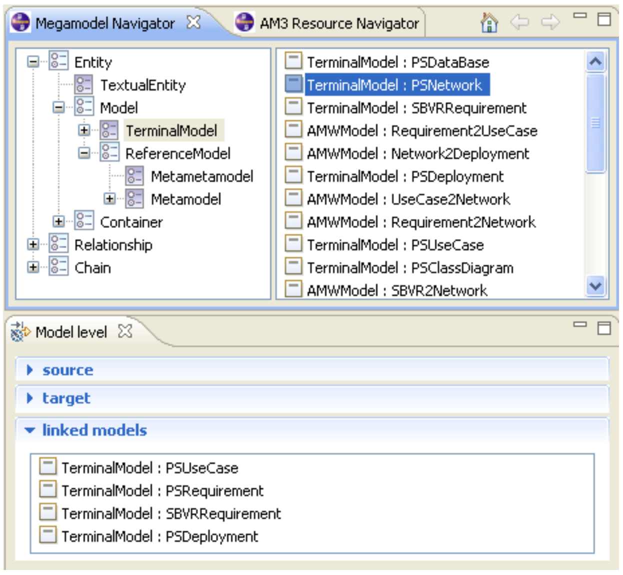


Figure 1 - Excerpt of the megamodel contents

In Figure 1, we see six models from the use case and some links between them. The red links are model level links, there are stored in the megamodel. The green ones are model-elements level links. There are stored in weaving models and are refinements of the corresponding model level links.

The two kinds of navigation presented here are implemented in the AM3 tool. The following screenshots present the user interface for the navigation.



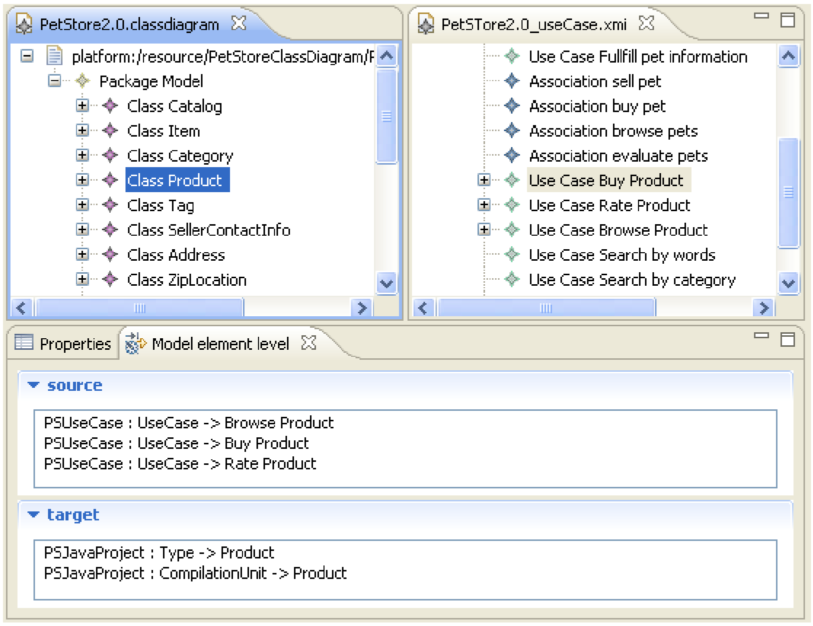
```
Entity
  TextualEntity
  Model
    TerminalModel
    ReferenceModel
      Metametamodel
      Metamodel
  Container
Relationship
Chain
```

```
TerminalModel : PSDataBase
TerminalModel : PSNetwork
TerminalModel : SBVRRequirement
AMWModel : Requirement2UseCase
AMWModel : Network2Deployment
TerminalModel : PSDeployment
AMWModel : UseCase2Network
AMWModel : Requirement2Network
TerminalModel : PSUseCase
TerminalModel : PSClassDiagram
AMWModel : SBVR2Network
```

```
source
target
linked models
  TerminalModel : PSUseCase
  TerminalModel : PSRequirement
  TerminalModel : SBVRRequirement
  TerminalModel : PSDeployment
```

Figure 2 - Model level navigator view

In Figure 2, we see the megamodel navigator allowing the selection of models in the megamodel. The class diagram model is selected and we can see in the "Model level" view the model linked to the "PSNetwork" (i.e.: PSUseCase, PSRequirement …). The linked models can be directly opened by the "Model level" view.



```
platform:/resource/PetStoreClassDiagram/P
  Package Model
    Class Catalog
    Class Item
    Class Category
    Class Product
    Class Tag
    Class SellerContactInfo
    Class Address
    Class ZipLocation
```

```
Use Case Fullfill pet information
Association sell pet
Association buy pet
Association browse pets
Association evaluate pets
Use Case Buy Product
Use Case Rate Product
Use Case Browse Product
Use Case Search by words
Use Case Search by category
```

```
source
  PSUseCase : UseCase -> Browse Product
  PSUseCase : UseCase -> Buy Product
  PSUseCase : UseCase -> Rate Product
target
  PSJavaProject : Type -> Product
  PSJavaProject : CompilationUnit -> Product
```

Figure 3 - Model-element level navigator view

In Figure 3, we present the class diagram and use-case models and the "Model element level" view. On a selection event (here, in the class diagram model it is "Class Product") the linked model-element are found (here, the "Use Case Buy Product" of the use case model). The linked model can be opened using the "Model element level" view. The corresponding model-element is automatically selected in the editor (here the "Use Case Buy Product" item).

The development of this use case, realized by AtlanMod [3], has been supported by the French ANR TLOG IDM++ project (Model Driven Engineering ++) and by the IST European MODELPLEX project (MODELing solution for comPLEX software systems, FP6-IP 34081) [6].

2. Metamodels

We provide in this subsection some general descriptions of the different metamodels involved in the "Pet Store" use case.

2.1. Class Diagram Metamodel

This metamodel defines all the concepts used in a class diagram with the concepts of *packages*, *types* or *classifiers* and also the different types of relations that can be used between some of theses concepts like the *associations* and *generalizations*.

2.2. SQLDDL Metamodel

This metamodel defines a subset of SGL Data Manipulation Language. It contains SELECT, CREATE VIEW, INSERT statements. It is available on the metamodel Zoo [7].

2.3. Deployment Metamodel

This metamodel describe a basic structure allowing the representation of the different nodes and links for the specification of system deployments.

2.4. Java Project Metamodel

This metamodel defines a structure allowing the representation of a Java project with the description of the different owned packages and compilation units.

2.5. Requirement Metamodel

This metamodel defines a basic representation for requirements. It allows the representation of the purpose of the requirement model and the definition of the requirements statements.

2.6. SBVR Metamodel

This metamodel has been retrieved from the Eclipse MDT/SBVR project [9]. It is an implementation of Semantics Business Vocabulary and Business Rules (SBVR) [8] OMG specification.

2.7. Use Case Metamodel

This metamodel defines a basic structure for use cases definition. It allows the representation of use cases that can include or extend others.

2.8. Web Service Link Metamodel

It is a basic metamodel allowing the representation of services in a network and the links between these services.

2.9. Web Site Metamodel

It defines a simple metamodel allowing the representation of a web site internal structure. The web site is represented by a set of documents that can be HTML or JSP pages, CSS documents or JavaScript source files. The links between these documents are also represented.

References

[1] The **Pet Store** Use Case: http://www.eclipse.org/gmt/am3/useCases/PetStoreNavigability/

[2] The **Pet Store Application**: http://java.sun.com/developer/technicalArticles/J2EE/petstore/

[3] The **AtlanMod** Team: http://www.emn.fr/x-info/atlanmod/index.php/Main_Page

[4] The Eclipse/GMT **AM3** Component: http://www.eclipse.org/gmt/am3/

[5] The Eclipse/GMT **AMW** Component: http://www.eclipse.org/gmt/amw/

[6] The **MODELPLEX** IST European Project: http://www.modelplex-ist.org

[7] The **AtlanMod** Metamodel Zoo: http://www.emn.fr/x-info/atlanmod/index.php/Atlantic

[8] The **SBVR** OMG Specification: http://www.omg.org/spec/SBVR/1.0/

[9] The Eclipse/MDT **SBVR** Component: http://www.eclipse.org/modeling/mdt/?project=sbvr