

---

# Example: Modeling and generating a service for a SOA environment

Dieter Moroff

\$Revision: 1.2 \$

## Table of Contents

1. Modelling a simple service .....	1
1.1. Prerequisites .....	1
1.2. Create a business component .....	4
1.3. Create a business service and objects used by the service .....	5
1.4. Add attributes to business objects .....	6
1.5. Creating associations between business objects .....	8
1.6. Adding operations to business services .....	8
2. Generating a web service from the model .....	9
2.1. Creating generator profiles .....	9
2.2. Creating generation sets .....	10
2.3. Start the generation .....	12
3. Implementing and deploy the web service .....	15
3.1. Implementing the business functionality .....	15
3.2. Deploying the web service with JBoss .....	15

To create a service in a SOA environment basically three things are required:

- The business service
- The objects used by the service
- Exceptions thrown by the service

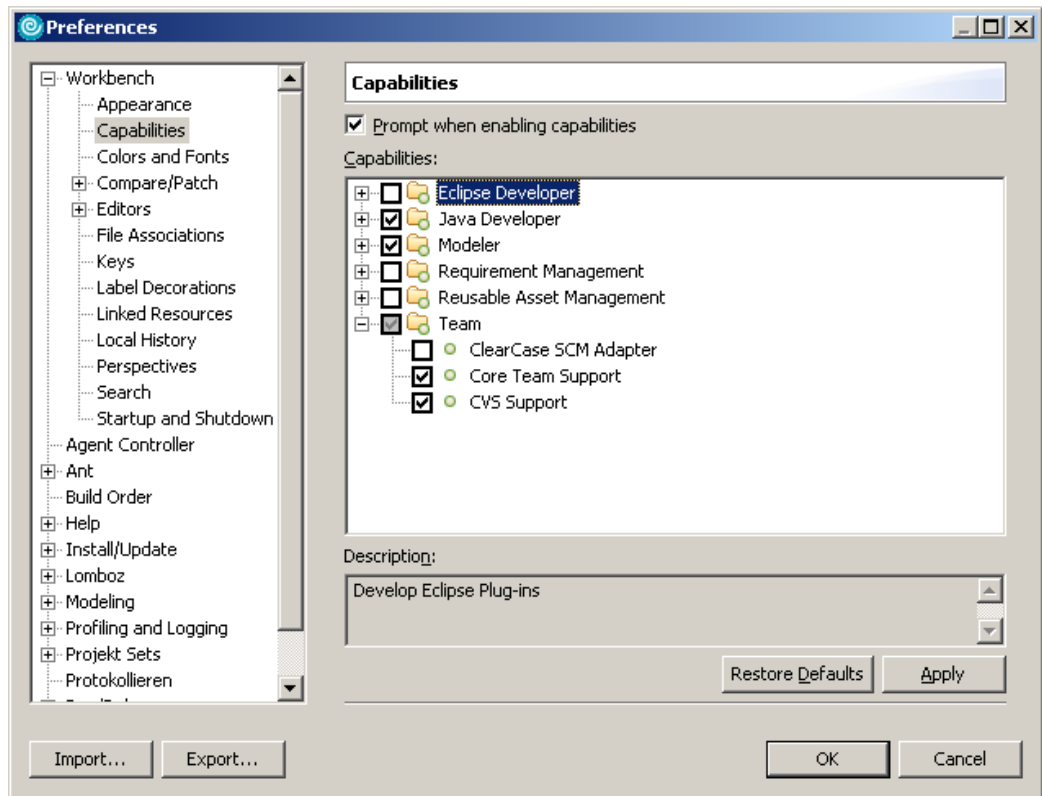
When modelling a service there should be not assumptions to the technology implementing the service. The service can be implemented using J2EE, Web Services, CORBA or and other technologies. Using or implementing the business functionality of the service should also be independent of the technology.

This article describes how to model and generate a simple service using IBM Rational Software Modeler/Architect (RSM/RSA).

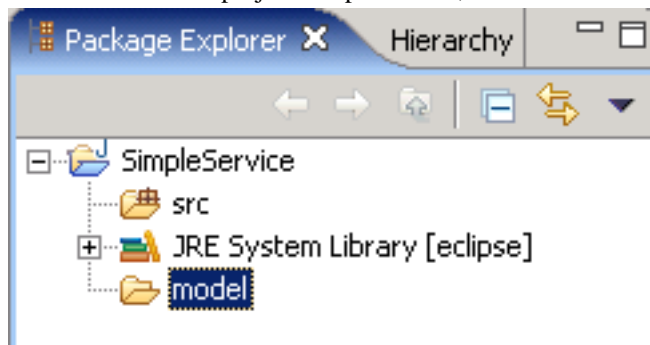
## 1. Modelling a simple service

### 1.1. Prerequisites

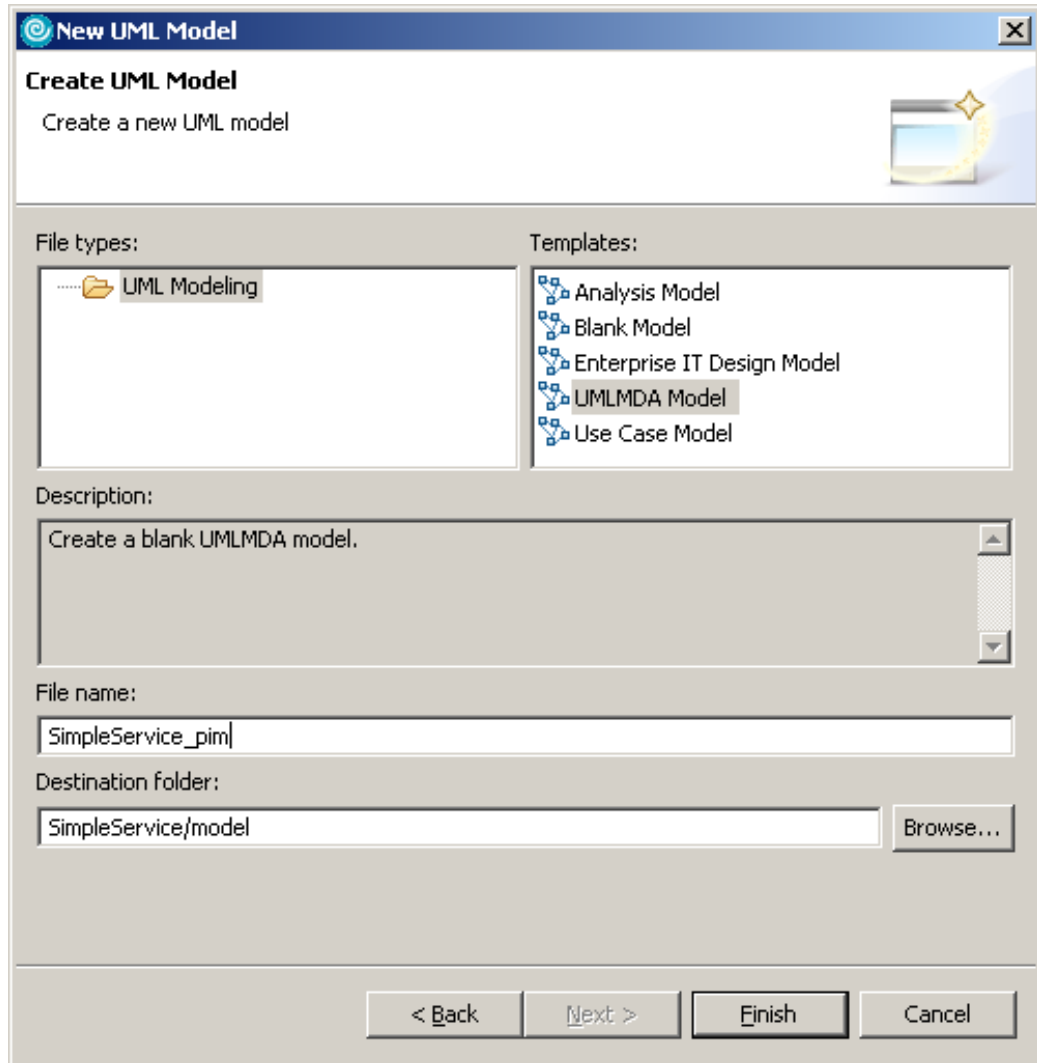
- RSM/RSA 6.0 or higher is running
- UMLMDA Features are installed
- Java and Team capacities are enabled in RSM/RSA, see Window|Preferences..|Workbench|Capabilities.



- Create an new Java project SimpleService, create a model folder

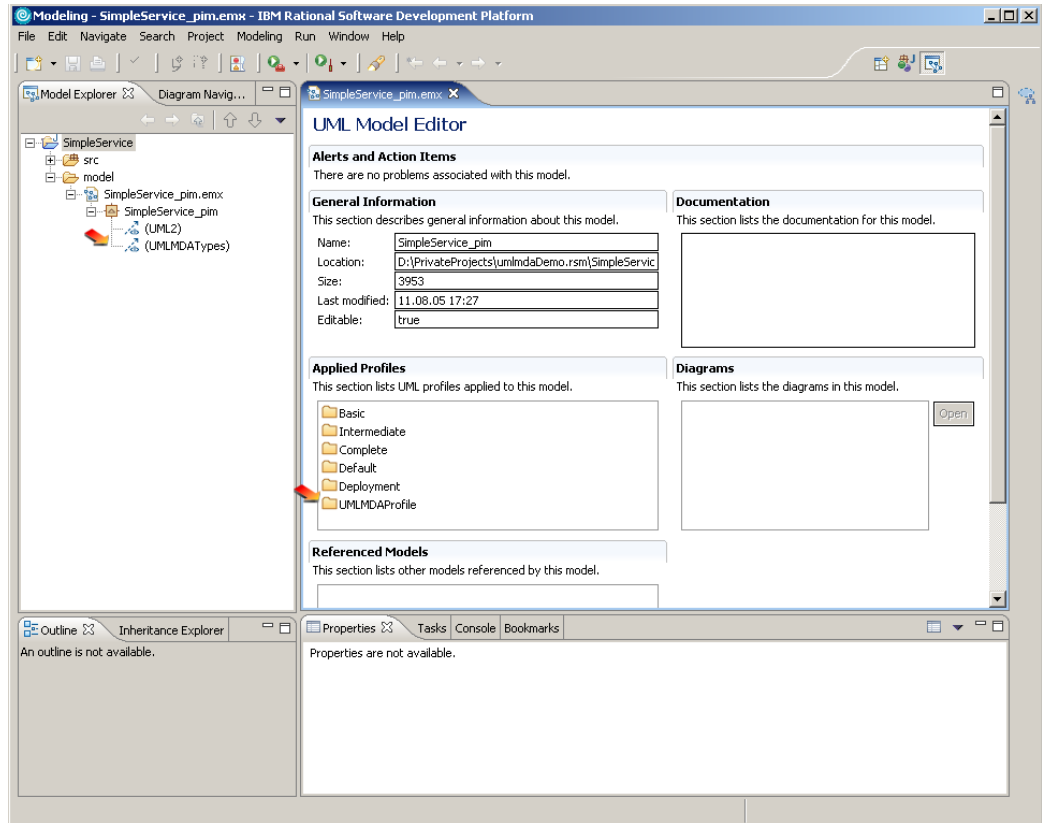


- Create a new UML model, with File|New|Other...|UMLModel. Select UMLMDA Model from templates.



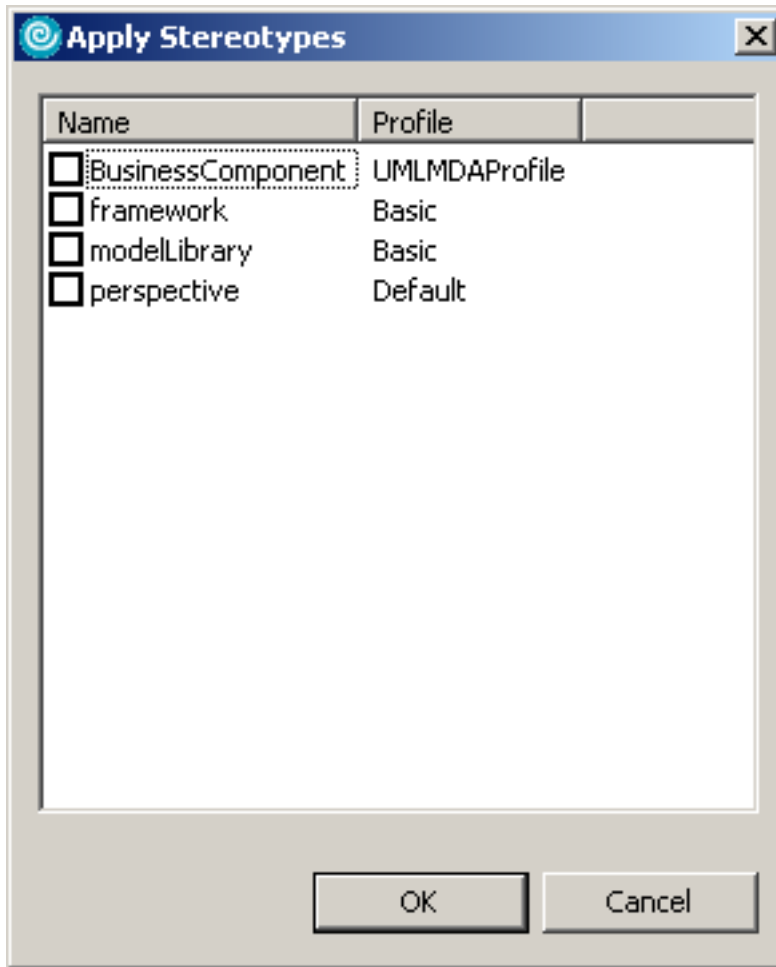
- Open the new model and switch to the Modelling perspective. In the model you will find the UMLMDA Profile and the imported UMLMDA Types.

## Example: Modeling and generating a service for a SOA environment



## 1.2. Create a business component

A business component is an element in the SOA environment which provides services to other components. A business component is modeled as a package. Create a package in the desired namespace and add the stereotype <<BusinessComponent>> on the stereotypes properties tab.

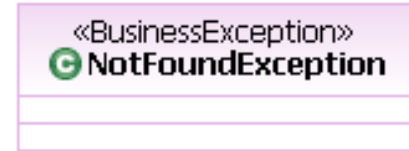


### 1.3. Create a business service and objects used by the service

Open a class diagram in the business components package. Add classes for the business service, the objects used by the service and the exceptions thrown by the service. Add the stereotypes <<BusinessService>>, <<BusinessObject>> and <<BusinessException>> to the classes, only one of these stereotypes should be added to a class. To make the diagrams easier to understand, it is a good idea to use different colors for the different stereotypes (also it makes no difference in the model information).

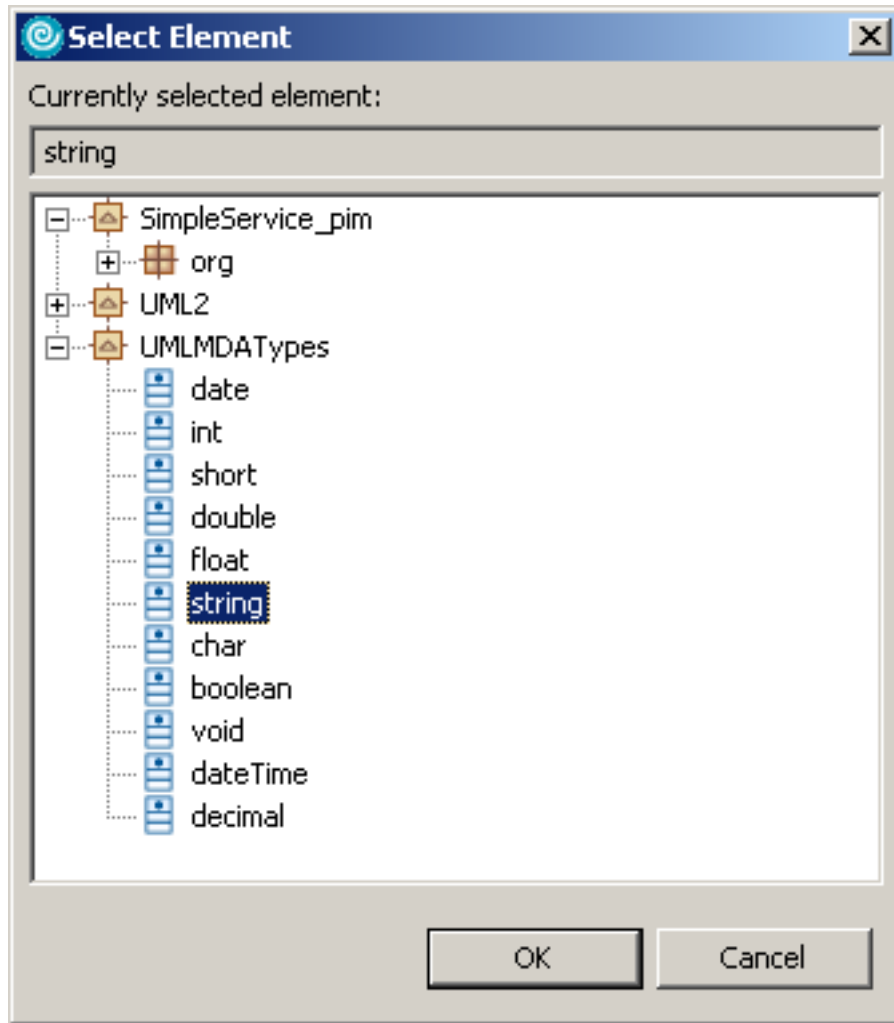
#### Caution

If more than one stereotype is added to a class an error is generated during generation.

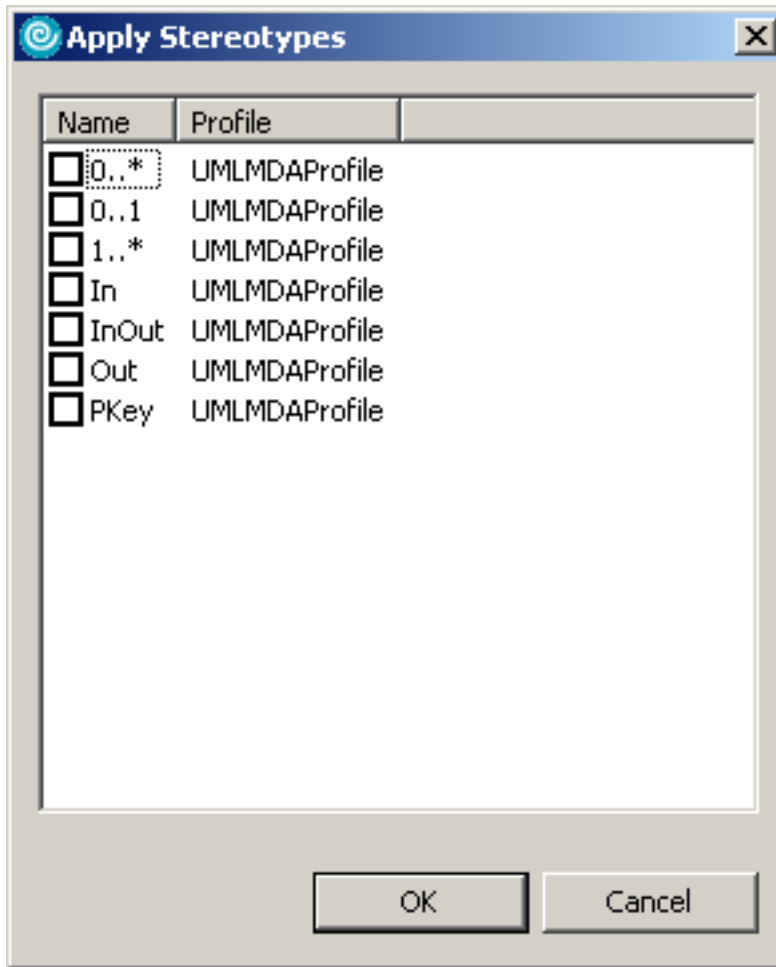


## 1.4. Add attributes to business objects

Add attributes to the business object, use the data types provided from the UMLMDA type library. You can enter the name of the data type following the attribute name (i.e. `firstname : string`), or select the type from the General property tab invoking Select type ....



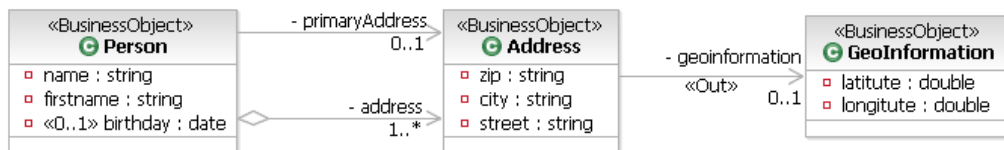
To make an attribute optional, add the stereotype `<<0..1>>` to the attribute. For future use are the stereotypes `<<0..*>>` and `<<1..*>>` to support multivalued attributes. The stereotypes `<<In>>`, `<<Out>>` and `<<InOut>>` can be used to optimize the communication. Attributes marked with the stereotype `<<In>>` are only passed to the service using the containing class as a service input, using the class as service output the attribute is omitted. Marking with the `<<Out>>` stereotype works the other way around. `<<InOut>>` is the default behaviour, the attribute is used as input and output, it's just to make this visible.



The visibility of attributes are ignored by the UMLMDA generators.

## 1.5. Creating associations between business objects

Associations between two business objects can be made. Using simple association, aggregation or composition is only important when business objects are used for persistence. An association must be navigable at least to one end.

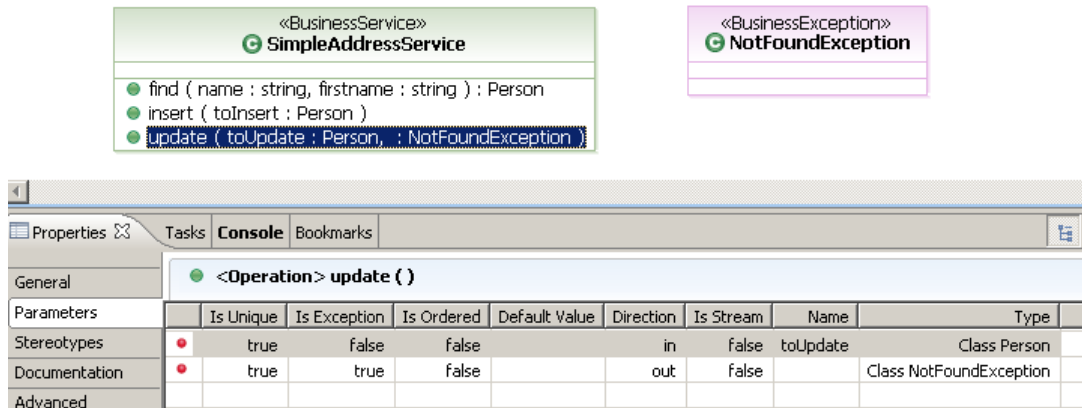


Like attributes, an association can be marked as `<<In>>` and `<<Out>>`. If marked this way, the whole object is passed only in the desired usage.

## 1.6. Adding operations to business services

To provide functionality to other components, operations must be added to the business services.





### 1.6.1. Modeling exceptions

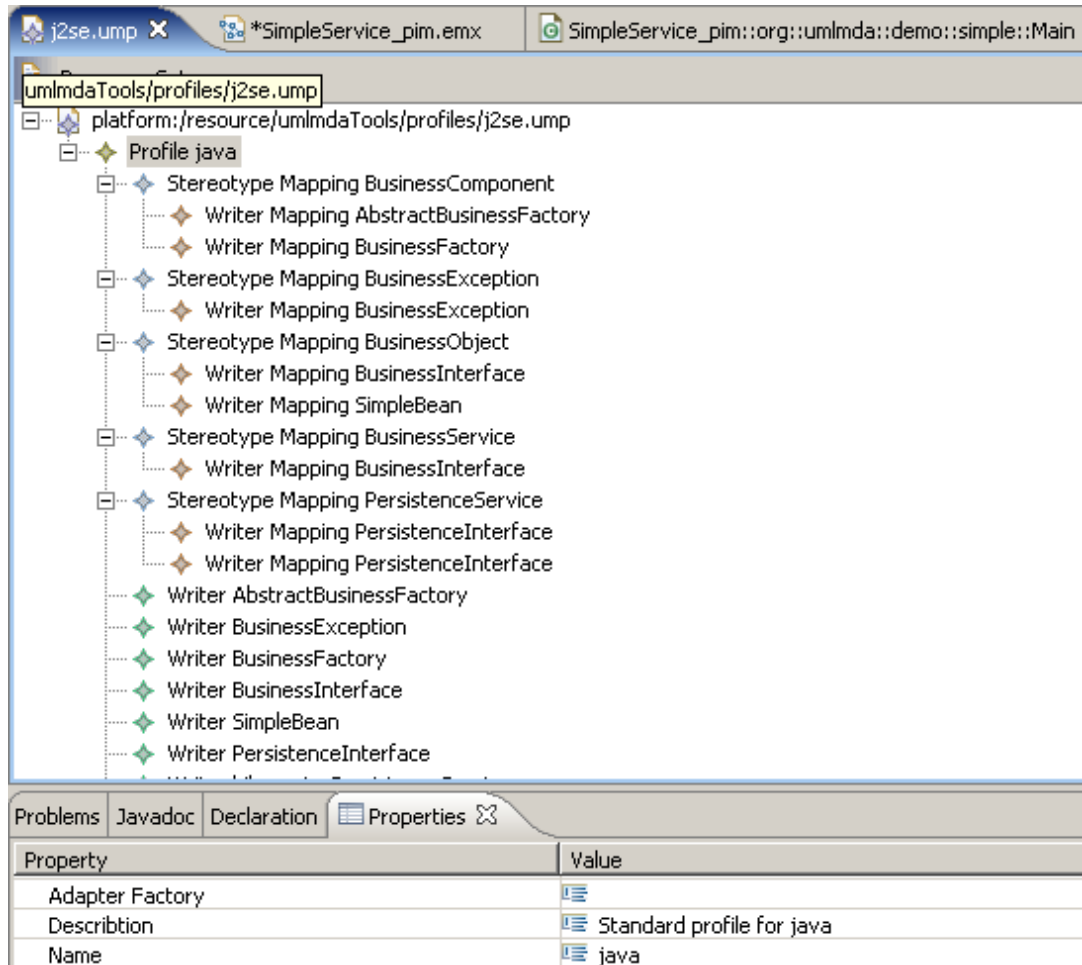
To model an exception a class with the stereotype `<<BusinessException>>` have to be modelled. To add the exception to a method go to the Parameters property tab, create a new parameter, set Is Exception to true and Direction to out and remove the Name and choose a class with the stereotype `<<BusinessException>>`.

## 2. Generating a web service from the model

The following sections describe how the generate code from a model using the transformation facilities of RSM/RSA. Using the ANT-Task is quite different.

### 2.1. Creating generator profiles

UMLMDA generator profile are to define which writers exists and used for the specific stereotypes. The generator profiles are not specific to and model they can be used out of the box from the UMLDA distribution.

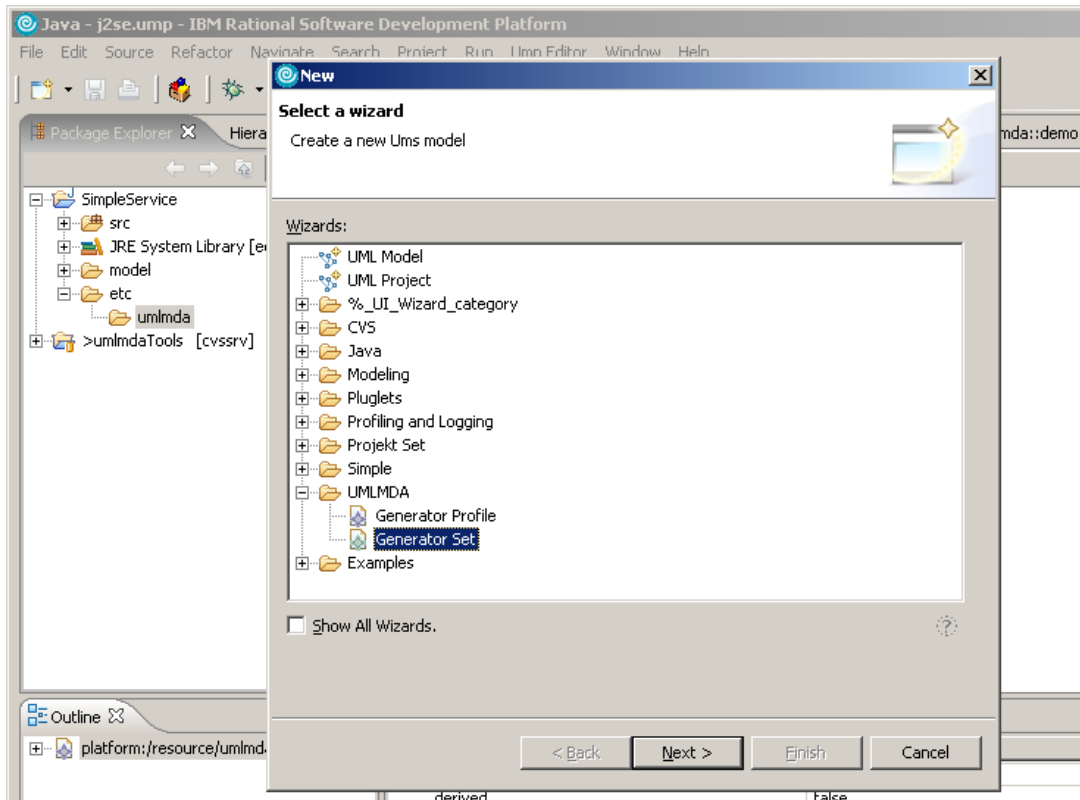


To generate the profiles must be contained in a workspace project.

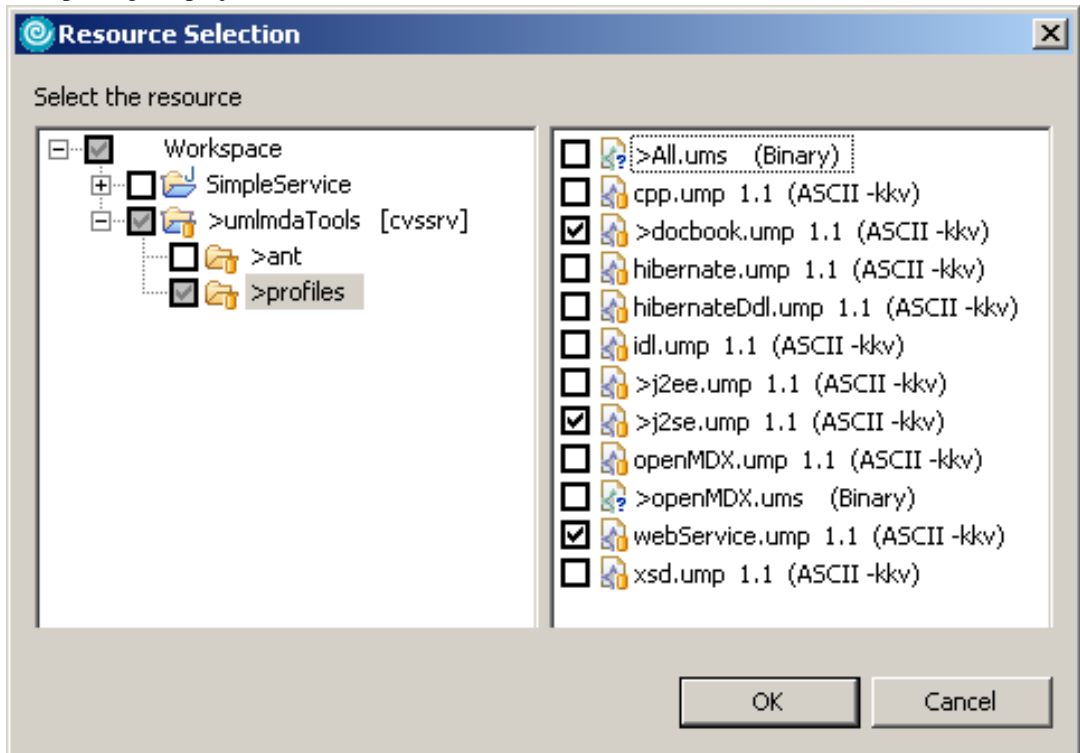
## 2.2. Creating generation sets

The second file for generation with UMLMDA is the generation set. The generation set can define which profiles are used, for which namespace the generation will be done and a relative destination path where the generated files will be stored. To create a generation set create the folder `etc/umlmda` in your project. With `File|New|Other...` you start a wizard.

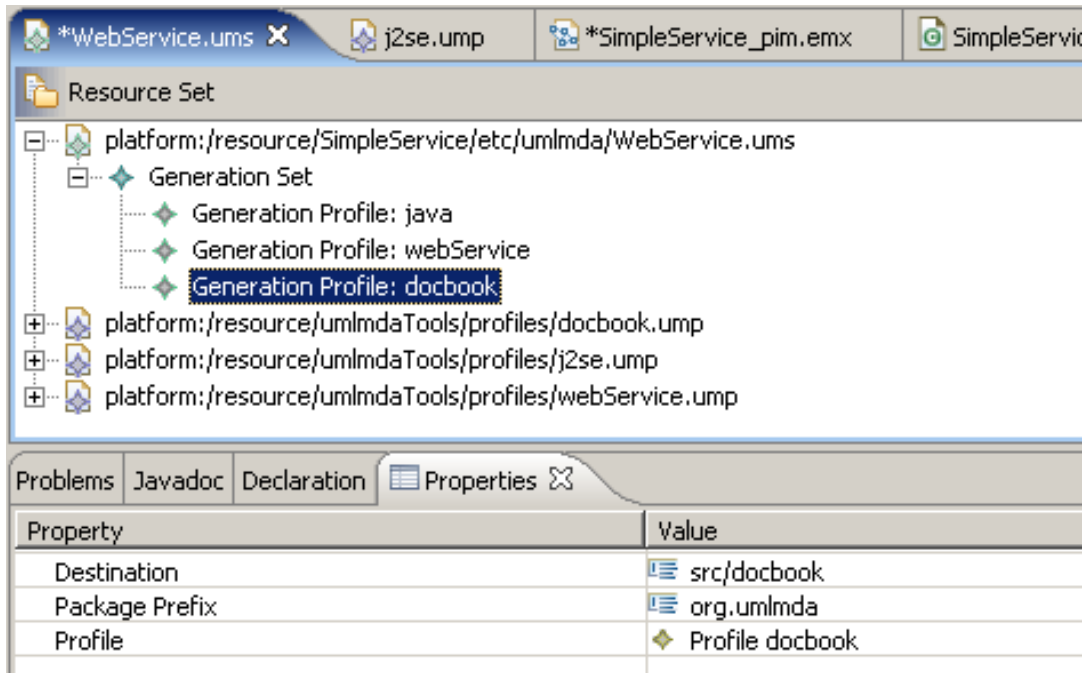
Example: Modeling and generating a service  
for a SOA environment



Select UMLMDA|Generator Set. Enter the name of the generation set and choose a XML encoding. First you have to import the generation profiles you want to use, select Load Resource ... from the context menu. And choose all generation profile (ending with ump). Choose the resources from the workspace, choosing from the file system will result in absolute file path, which may cause problems porting the project to other locations.



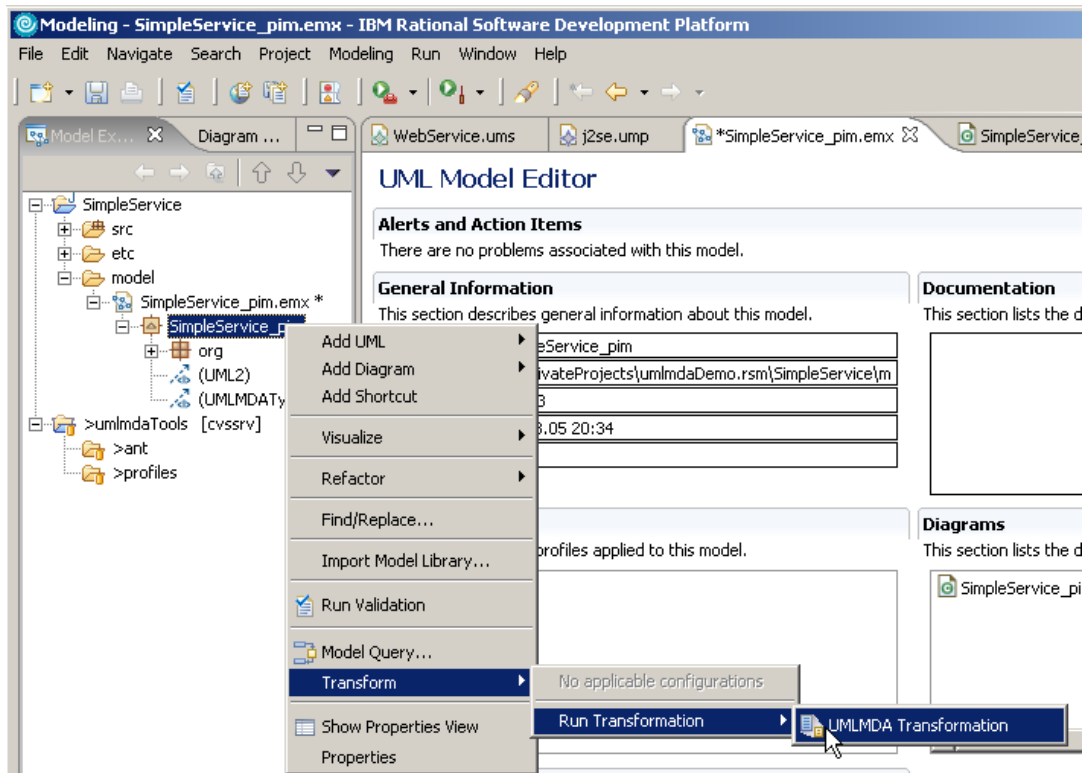
In our example we want to generate code for webservices, we chooes docbook.ump (to generate DocBook based documentation), j2se.ump to generate Java business interfaces and dtos and webService.ump to generate web services.



In the UMLMDA generator set editor, the profiles are assigned to a destination. If a package prefix is defined, only elements which are below this namespace are generated.

## 2.3. Start the generation

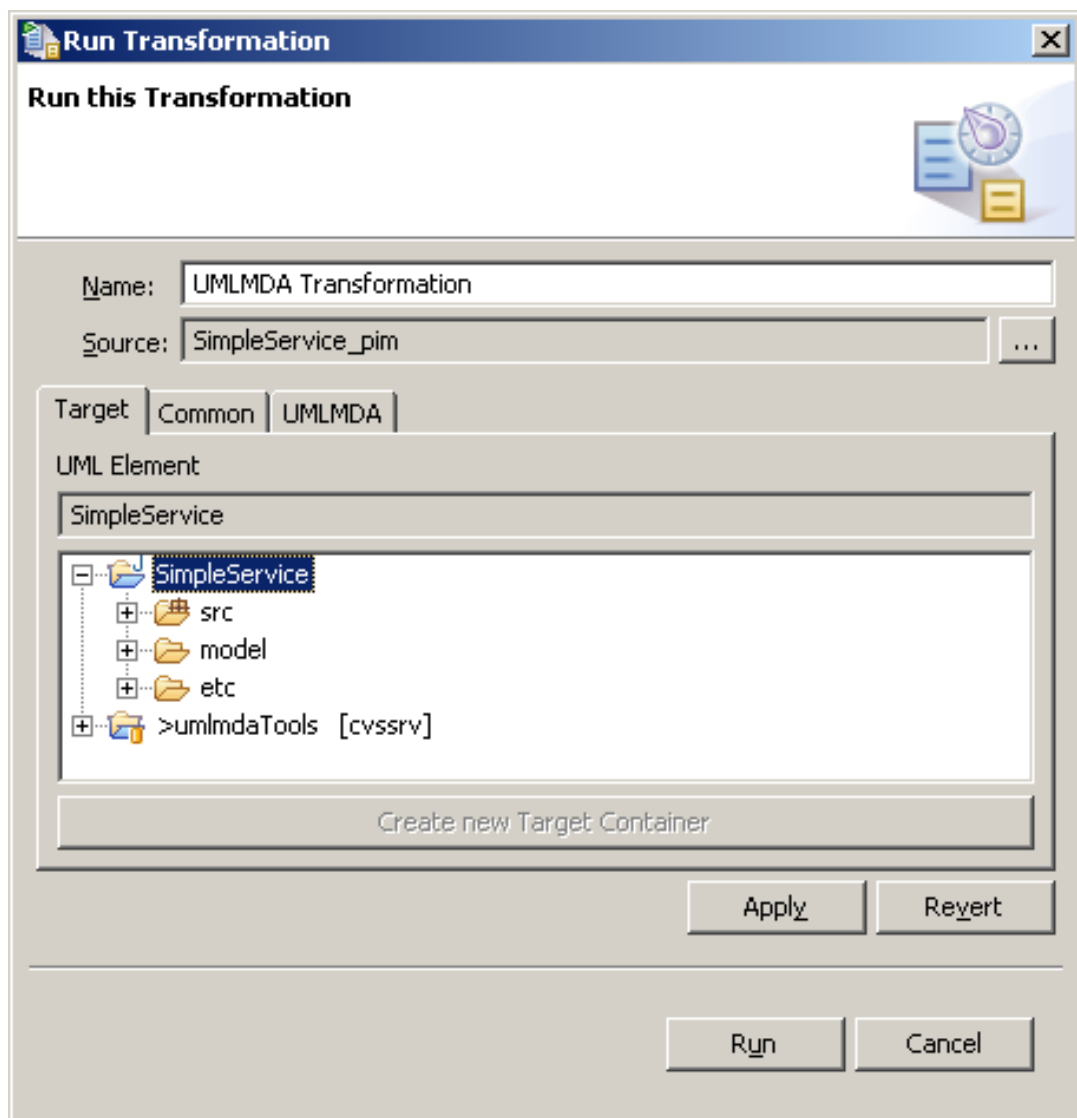
To start the generation in RSM/RSA the source model must be open and the model must be selected in the model explorer. Choose Transform|Run Transformation|UMLMDA Transformation from the context menu.



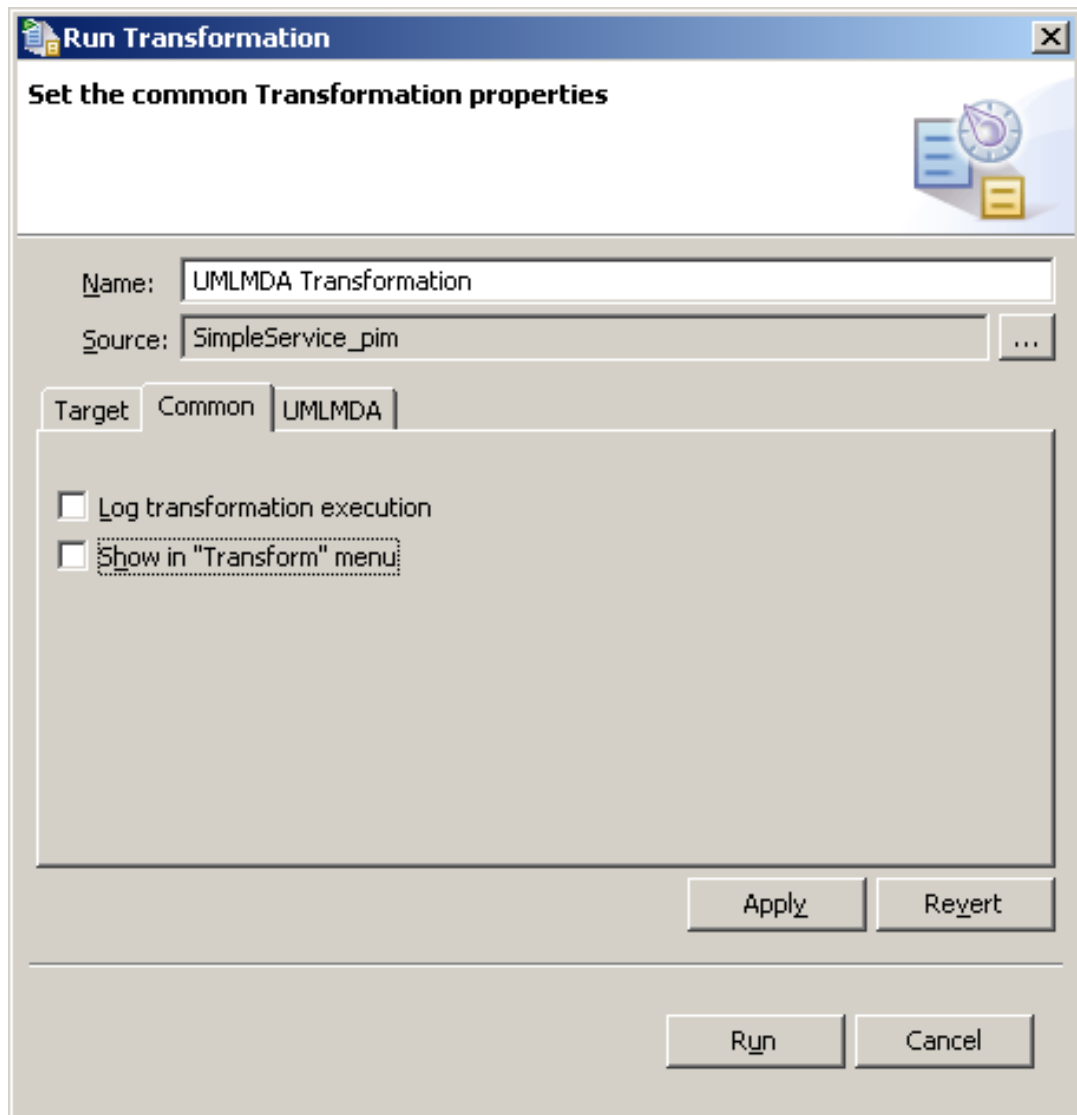
### Caution

Choosing and other item in the model explorer will show the UMLMDA transformation in

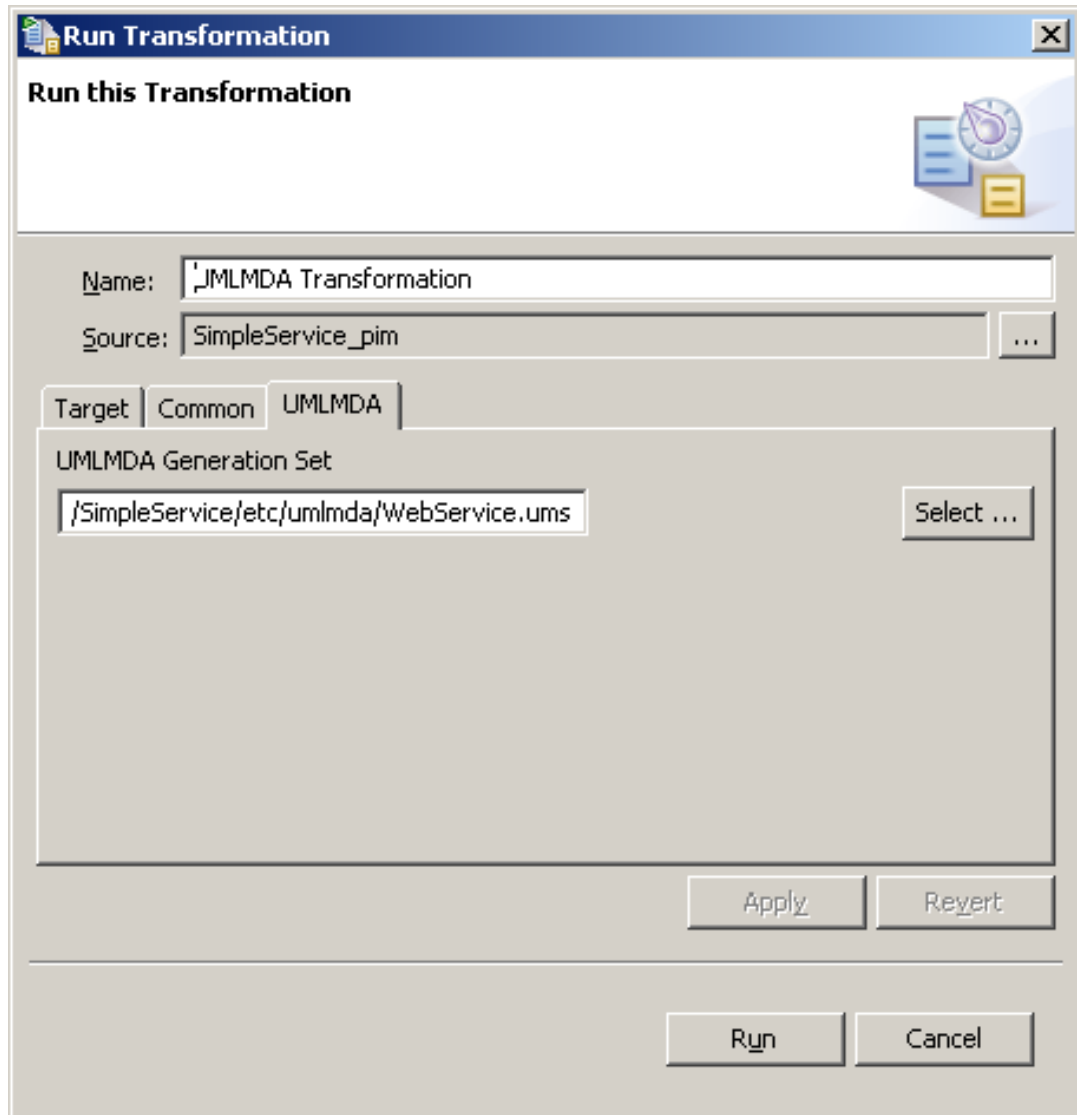
the context menu but it won't work.



On the first tab of the transformation wizard the generation target can be selected. The source must be an open UML model.



On the second page common properties can be set. Uncheck Shown in "Transform" menu, to avoid misuse of a transformation in with a wrong model.



Last select the previous created generation set, and start generation.

## 3. Implementing and deploy the web service

To be done ...

### 3.1. Implementing the business functionality

### 3.2. Deploying the web service with JBoss