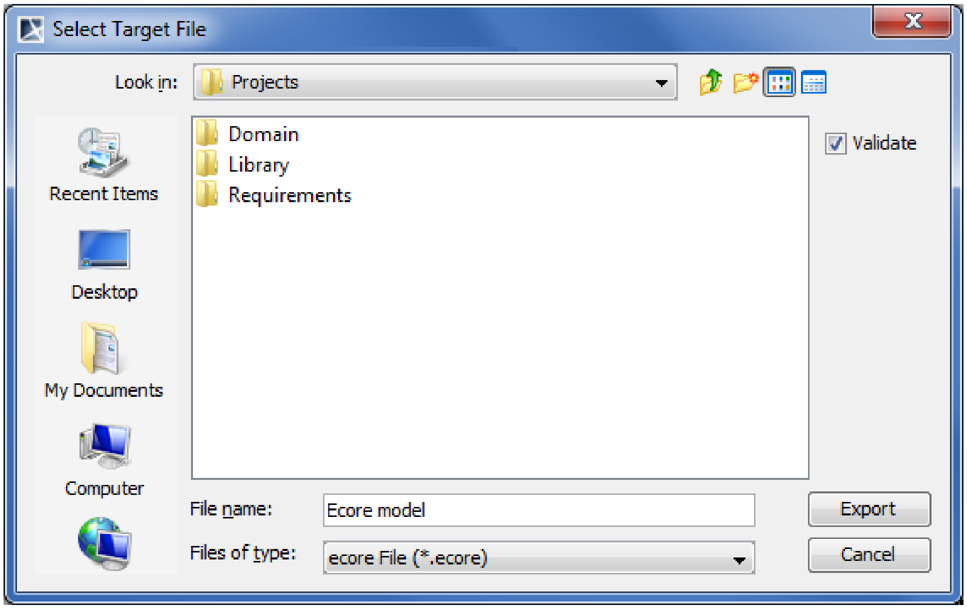
Exporting projects to Ecore files

The Ecore model exporting is very similar to the EMOF / CMOF model exporting. After the Ecore model is created, you can export either the whole project or selected packages to an .ecore file.

Exporting simple projects

To export a project to an Ecore file

1. From the main menu, select **File** > **Export To** > **EMF Ecore File** > **Ecore Whole Model**. The **Select Target File** dialog opens (see the following figure).
2. Type a file name and select a location for the exported model.
3. Click to select or to clear the **Validate** check box.
4. Click the **Export** button when you are finished.



Select Target File

Look in: Projects

Domain
Library
Requirements

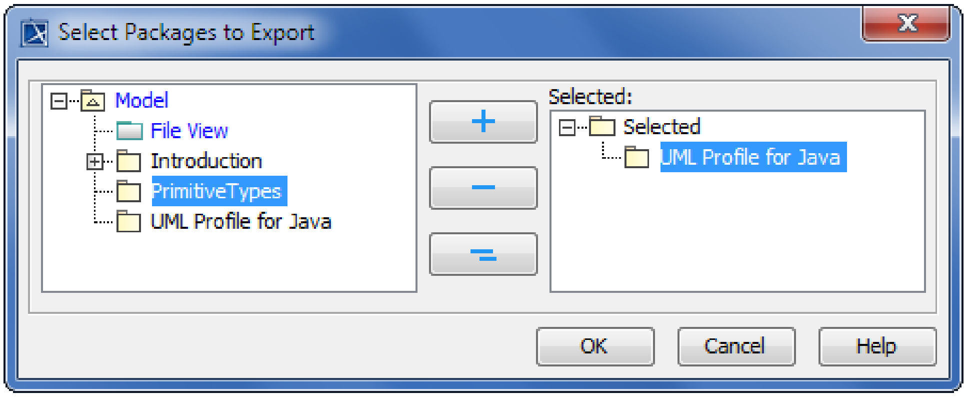
☑ Validate

File name: Ecore model

Files of type: ecore File (*.ecore)

To export selected packages to Ecore file

1. From the main menu, select **File** > **Export To** > **EMF Ecore File** > **Ecore Selection**.
2. In the **Select Packages to Export** dialog, select packages you want to export and click **OK** when you are done.



Select Packages to Export

Model
File View
Introduction
PrimitiveTypes
UML Profile for Java

Selected:

Selected
UML Profile for Java

3. Perform the procedure **To export a project to an Ecore file** starting from the **step #2**.

Exporting projects containing used projects

It is important to notice that any used projects in a project are not exported together with the project on its export to an Ecore file. Only references to these used projects are exported. Therefore the output of the project export is an Ecore file (model) containing the direct content of the project and referencing to other Ecore files (used projects) that are used in the project. For referencing to used projects, i.e., other Ecore files, Ecore references are used.

Each used project used by the project must be exported to an Ecore file individually. For this you have to open each used project as a project and then export it to an Ecore file.

It is strongly recommended to read the following paragraph before your very first attempt to export a project that uses used projects. Getting familiar with this information may help you to escape a serious problem that arises because of the nature of Ecore references.

Ecore references, crossing a resource boundary (when the element in one file references the element in another file), *are qualified-name-based, but not id-based* as in case of CMOF, EMOF, or UML. Hence when exporting a project with references to elements in other projects (used projects), the export tool must know the full path of elements in the used project's Ecore file. This information cannot be determined from UML model without additional information. For this the following approach is adopted:

1. **Save the used project after the export.** After a project is exported to an Ecore file, the qualified names (paths) of the shared elements are recorded in special stereotypes («EcoreExportServiceInformation») / tags (ecoreExportPath) of the project's shared packages. The side effect is that the project is modified during the export. To preserve this information for later usage, you need to save the project after the export.

2. **Export used projects before exporting the main project.** When a project that references to elements in one or more used projects is exported to an Ecore file, the export tool needs to know these elements' paths, saved on the appropriate used project export to Ecore. If this information is missing (e.g., in case the project is being exported before used projects are exported), the export tool tries to guess the correct path of each element in the used project and gives warnings about this. That is why used projects should be exported to Ecore before exporting the main project that uses these used projects.

Exported data type mappings

Standard UML data types are exported as standard Ecore data types. The following table shows which Ecore data type corresponds to which UML data type.

UML data type	Ecore data type
String	EString
Boolean	EBoolean
Integer	EInt
UnlimitedNatural	EInt
Real	EReal

Standard data types from the MagicDraw profile are exported as Ecore types. The following table shows which Ecore data type corresponds which UML data type.

Data types in MagicDraw profile	Ecore data type
boolean	EBoolean
byte	EByte
char	EChar
date	EDate
double	EDouble
float	EFloat
int	EInt
real	EReal
long	ELong
short	EShort

The void data type is exported as an absence of the type.

References to Ecore model elements (standard datatypes like EInt, metaclasses like EStructuralFeature), defined in the standard Ecore library are exported as standardized Ecore references to Ecore metamodel elements (the resource identifier part of the Href is *http://www.eclipse.org/emf/2002/Ecore*).

Ecore elements validation

The modeling tool provides a validation suite for validating a model that is being exported to an Ecore file. This suite contain batches of rules to check exported elements. Warnings about not exported elements (for example, diagrams, behavioral elements, or other) are displayed after the validation process is completed.

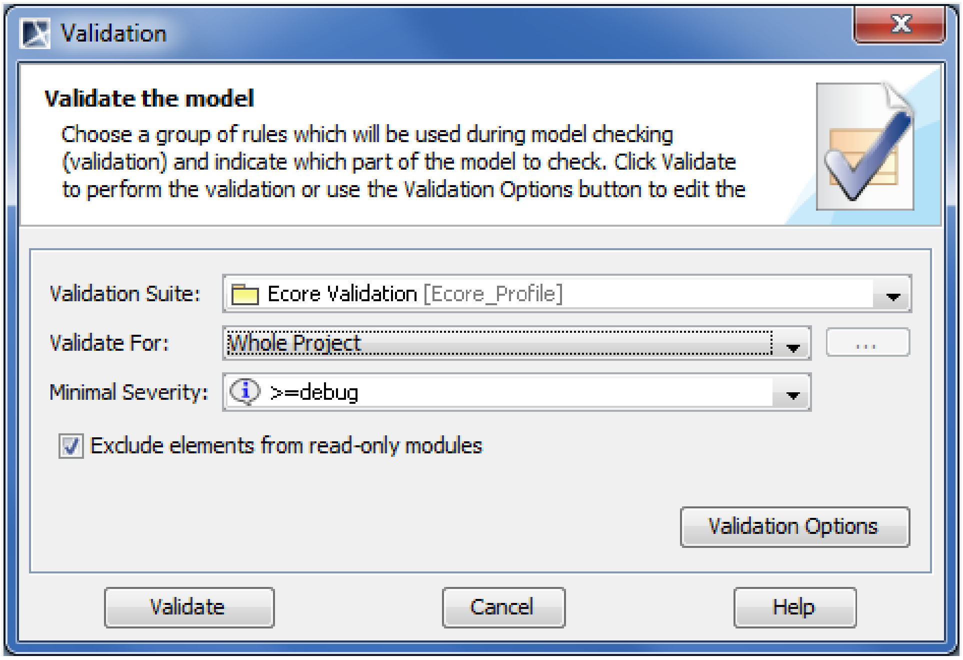
The validation process does not preclude the model from being exported. Unsuitable elements are simply skipped.

You can run the Ecore validation on a model (project or used project) export to Ecore. All UML elements that are not suitable for the Ecore, will be highlighted.

You can also run this validation suite at any time while you are developing an Ecore model.

To run the Ecore validation

1. From the main menu, select **Analyze** > **Validation** > **Validate**. The **Validation** dialog opens.



Validation

Validate the model

Choose a group of rules which will be used during model checking (validation) and indicate which part of the model to check. Click Validate to perform the validation or use the Validation Options button to edit the

Validation Suite: Ecore Validation [Ecore_Profile]

Validate For: Whole Project

Minimal Severity: >=debug

☑ Exclude elements from read-only modules

2. In the **Validation Suite** drop-down list, select **Ecore Validation**.
3. In the **Validate For** drop-down list, select the scope of the validation.
4. In the **Minimal Severity** drop-down list, select the level of severity.
5. Click **Validate**.