UML metamodel changes

There are two types of changes in UML metamodel API.

UML specification changes

The UML 2.1.2 (2.2) specification introduced few changes in the UML metamodel itself. Some metamodel classes were added, some of them were removed, some metaclasses properties were changed. Most of these changes are not in a core UML, so they will affect you if your plugin is oriented to complex things in UML.

Most changes are made in UML Templates, Simple Time, Interactions. We suggest to look at UML 2.1.2 (2.2) specification for such changes if you see some compile errors.

UML metamodel API implementation changes

Earlier UML metamodel API used several interface layers for every compatibility level described in the UML specification. All these layers were merged into one using package merging, this is why earlier API had so many interfaces for every metamodel element in a different package. Such structure was very complicated and hardly understandable for new UML users.

MagicDraw version 15.0 and later UML metamodel API provides just one final merged layer. All intermediate layers were dropped from API. This change reduced a UML metamodel API size in few times.

Interfaces that are removed form the following packages:

- *com.nomagic.uml2.ext.omg*

- *com.nomagic.uml2.magicdraw*

- *com.nomagic.uml2.omg*

We left the same package name for merged interfaces like in a previous API version, that is *com.nomagic.uml2.ext.magicdraw.***. You do not need to make any changes in your code if you were using interfaces from this layer. Otherwise you need to import a statement in your java source files, for example, *import com.nomagic.uml2.omg.kernel.Element* to *import com.nomagic.uml2.ext.magicdraw.classes.mdkernel.Element*.

UML metamodel interfaces use Java 5 generics, so it is much easier to understand types of various collections in the metamodel.