Introducing Macro Engine

Macro Engine (previously called Script Engine) in MagicDraw allows you to create your own macro (script) by using BeanShell, Groovy, JRuby, JavaScript (Nashorn and Rhino), or Jython. With Macro Engine, you can control everything that is allowed in Open API, for example, transforming and manipulating models. Learn more about open API of MagicDraw or other Cameo Suite modeling tool (for example, Cameo Systems Modeler) in Developer Guide and find sample macros in *<MagicDraw installation directory>/samples/product features/macros*.

Macro Engine comes with Professional, Architect, and Enterprise Editions starting from MagicDraw version 16.6 and later.

MagicDraw 18.1 Macro Engine supports two types of Javascript: (i) Javascript Nashorn and (ii) Javascript Rhino. This support is intended to prevent some incompatibility issues in the language syntax with macros that were created using Java 7 or earlier (Javascript Rhino), because Oracle changed the built-in Javascript engine from Javascript Rhino to Javascript Nashorn since Java 8 was released.

When you refer to Javascript, Macro Engine will refer to the default Javascript that comes with the JRE, which is Javascript Nashorn. For example, if you use Java 8, the default Javascript will be Javascript Nashorn. If you find any incompatibility issues, Macro Engine provides the migration capability to downgrade the Javascript engine for all of the existing macros to use Javascript Rhino.

To migrate to Javascript Rhino

- Click **Tools** > **Macros** > **Migrate to Javascript Rhino**.

> ⊘ If you are using MagicDraw 18.1 or later with Java 8 and created a macro with the new language syntax in Nashorn, and then downgraded your Java to Java 7, this will cause an execution problem for your macro.