Supported scripting languages

Macro Engine supports five scripting languages: BeanShell, JavaScript (Nashorn and Rhino), Jython, Groovy, and JRuby.

BeanShell
BeanShell is a lightweight scripting language for Java. The shipped version is 2.1.7. The advantage of using BeanShell is that its syntax is compatible with Java; therefore, you can use the code assistant feature in most Java IDE. The BeanShell syntax documentation is available at *http://www.beanshell.org /docs.html*.

JavaScript
JavaScript is a scripting language based on Java syntax. The documentation is available at:
http://docs.oracle.com/javase/8/docs/technotes/guides/scripting/nashorn/index.html and *https://developer.mozilla.org/en/Rhino_documentation*.

The JavaScript *Nashorn* is deprecated with the intention of removing it (learn more about deprecated JavaScript Nashorn).

Please use JavaScript *Rhino* instead - see the JavaScript migration from Nashorn to Rhino engine.

Jython
Jython, a successor of JPython, is a Python programming language implemented in Java. The documentation is available at *http://wiki.python.org/jython*. Besides automating MagicDraw by using Jython scripting, the plugin development in the Jython programming language is also supported in MagicDraw. To find the usages of the JavaScript *Nashorn* in the project, run the **Deprecated JavaScript** validation suite (**Analyze > Validation > Validate**). For more information on Jython, see JPython Scripting in MagicDraw OpenAPI User Guide.

Groovy
Groovy is an agile and dynamic language for the Java Virtual Machine. Macro Engine uses Groovy 2.0.1. The advantage of using Groovy is that its Java-like syntax seamlessly integrates with the existing Java source code and libraries. It supports many IDEs that provide code completion and debugging. BeanShell scripts can be easily moved to Groovy, with some modifications. For more information on Groovy, go to *http://groovy-lang.org/*.

JRuby
JRuby is a 100% pure-Java implementation of the Ruby programming language. The documentation is available at *http://jruby.org/documentation*.

By default, JRuby is not included in the modeling tool, however, you can add it manually.

Related pages

- Using code completion to develop BeanShell scripts
- Using NetBeans IDE to develop Groovy Scripts
- Using Eclipse to develop Groovy scripts
- Installing Gems for JRuby
- Adding a scripting language to MagicDraw