Standards-based model execution framework
No Magic is the first in the industry to provide extendable model execution framework based on OMG fUML and W3C SCXML standards. It extends MagicDraw capabilities and allows validating system behavior by executing, animating, and debugging UML 2.0 Statemachine and Activity models in the context of realistic mock-ups of the intended user interface.

Why Simulation?
The purpose of a simulation is to gain system understanding without manipulating the real system, either because it is not yet defined or available, or because it cannot be exercised directly due to cost, time, resources or risk constraints. Simulation is typically performed on a model of the system, by visualizing and identifying system defects in early development stages.

fUML – the Foundation Language of Executable UML Models
OMG fUML standard defines a basic virtual machine for the Unified Modeling Language, enabling compliant models to be transformed into various executable forms for verification. It supports structural and behavioral semantics of systems, mainly based on UML Activities.

SCXML – State Chart XML
The W3C SCXML fills the OMG fUML gaps by providing a generic state machine–based execution environment based on Harel state charts. Most things that can be represented as a UML state charts – control systems, view navigation bits, interaction or dialog management, and many more – can leverage the SCXML engine. SCXML file can be used for third-party platform specific code generation.

SysML Parametrics
SysML Constraint blocks are used to specify a network of reusable constraints that represent mathematical expressions, which constrain the physical properties of a system (e.g. according physics laws) or calculate system MoEs (measure of effectiveness), e.g. cost, risk, performance, reliability etc.

Dynamic expressions evaluation in the context of full system simulation allows tracking and maintaining dependencies among critical parameters such as size, weight, speed, power, temperature and others throughout the system life cycle. Cameo Simulation Toolkit uses build-in math solver and/or interfaces to well-known external math solvers such as Matlab and OpenModelica.
Action Language
At some level of modeling, text based language is more convenient. Cameo Simulation Toolkit allows using any JSR233 based scripting language as action language for defining guards, decisions, default values, State's entry/do/exit actions and other OpaqueBehaviors. Javascript, Ruby, Groovy and Python are supported by default, other languages can be added by user.

Validation and Verification
Various kinds of validation and verification options are available: model validation before execution, runtime values constraints, conditional breakpoints, and post-simulation analysis. Constraints can be defined by using OCL or other languages (see Action Languages).

Debugging and Animation
Simulation Toolkit includes an integrated debugger for setting graphical breakpoints, stepping through charts, browsing and editing runtime objects and values and evaluating custom expressions. The execution trace is highlighted in diagrams where user can quickly see tokens and values in a tooltip on activated element symbol.

User Interface Prototyping
While execution allows monitoring values and injecting signals by using debugger, far better is to connect the behavior of the system to the realistic mock-up of the intended UI.

The Cameo Simulation Toolkit unleashes the power of MagicDraw User Interface Modeling Diagrams by constructing and running real Java Swing UI.

As the UI definition itself is in the same model as system, it takes just seconds to drop classes on frames, parts on panels, signals on buttons, properties on labels, sliders or checkboxes to relate user interface components and system properties. Pictures can be dropped directly from web browser or file system and used as part of UI or for dynamic representation of the different system states.

Additionally, Open API allows using dedicated third-party tools of your choice to create and plug realistic system mockups.