

MetaEdit+ Workbench: Define modeling languages and generators

MetaEdit+ Workbench allows you to build your own modeling tool — without having to write a single line of code. MetaEdit+ provides a simple yet powerful tool suite for designing your modeling language:

Step 1: Define the language concepts and rules graphically or use form-based metamodeling tools.

Step 2: Draw the notation with Symbol Editor or import your existing graphical elements.

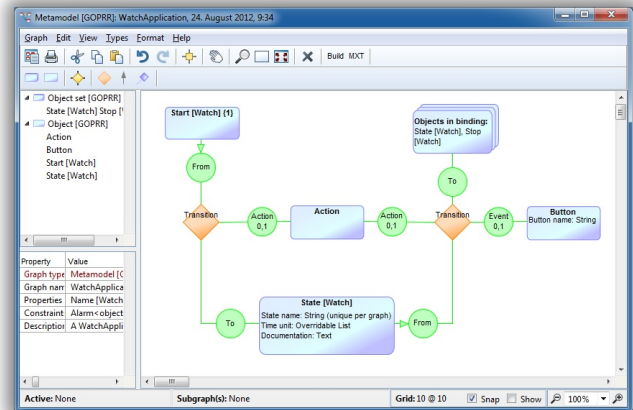
Step 3: Make generators to produce required code, configuration, analysis, testing data, configuration, etc.

Having defined a modeling language — or even a partial prototype — you and your team can start to use it in MetaEdit+. The metamodel can be changed on the fly and existing models updated automatically to the new language version.

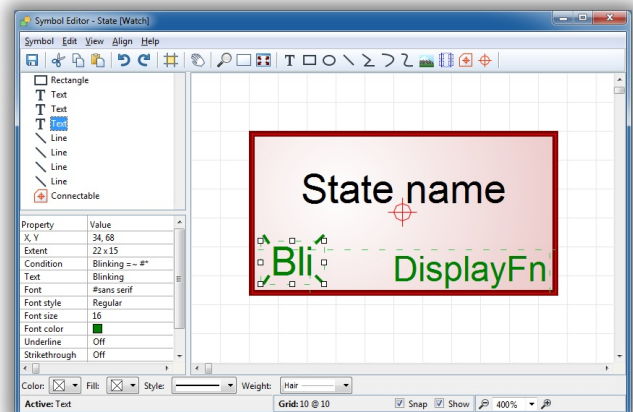
This is why MetaEdit+ is your best choice for creating your own modeling tool:

- Support for several integrated languages
- Graphical and form-based metamodeling: no programming needed
- Integrated metamodeling and modeling (use your language while you define it)
- Models update automatically yet non-destructively when a metamodel changes
- Multiple concurrent metamodelers
- A repository to handle various metamodels
- WYSIWYG Symbol Editor for defining representations for metamodel elements
- SVG and bitmap importing for symbols
- Code generation using templates, visitor pattern, crawlers and multiple streams
- Generate any language and output format
- Code generator debugger
- Generators and metamodels integrated
- Metamodel import and export in XML
- Large metamodel library available

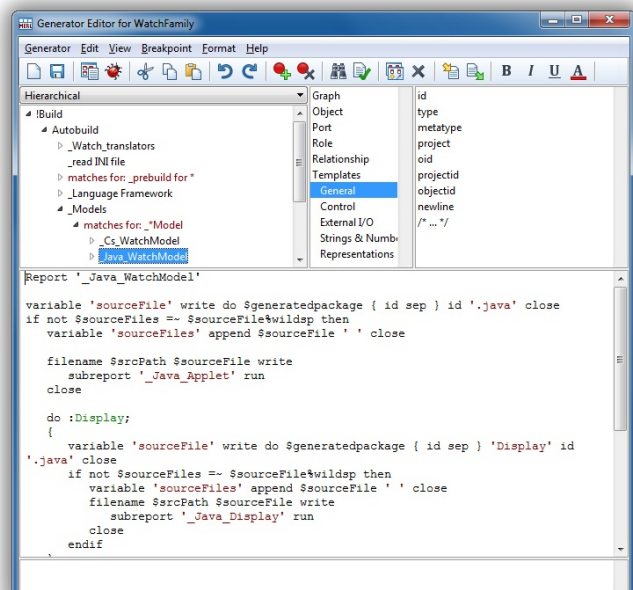
Step 1: Define metamodel



Step 2: Draw notation



Step 3: Make generators



MetaEdit + Modeler: Make models, generate code and documentation

MetaEdit+ Modeler follows the given modeling language definition and automatically provides the full modeling tool functionality: diagramming editors, browsers, generators, trace and refactoring tools, multi-user support, etc.

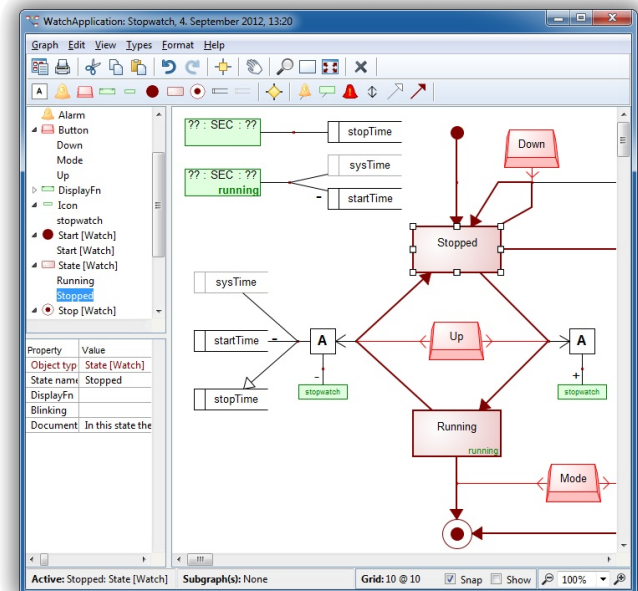
It integrates with your existing application development environment: reading in configuration data, referencing component libraries, simulating models and integrating in your build process.

MetaEdit+ as a multi-user tool. MetaEdit+ can run either as a single-user tool, or simultaneously on many clients connected by a network to a server.

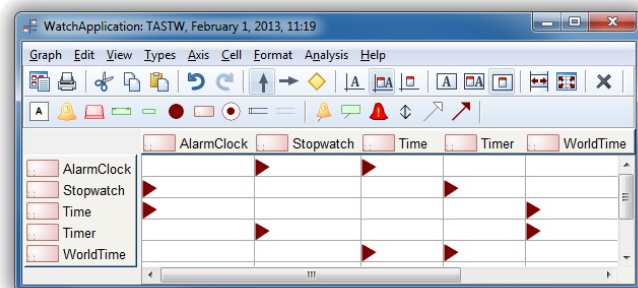
This is why MetaEdit+ is your best choice for a modeling and code generation tool:

- Diagram, Matrix and Table Editors
- Multi-user support: share and reuse models and model elements
- Eclipse and Visual Studio integration
- Straight model-to-code transformations: no need for intermediate formats
- “Live code”: click generated code to see original model element
- Code generation from multiple models
- Model animation and simulation support
- Re-generation support with protected blocks
- Metrics and model checking
- Model import and export in XML
- HTML and RTF document generators
- Integration with version control systems
- Automated trace of modeling history and changes
- SOAP/.NET/Web services API
- Scalable up to 4 billion design objects in one project
- Maintenance, support services and help desk
- Training and consulting services

Diagram Editor



Matrix Editor



Browsers

