Spoofax.modelware

Oskar van Rest, Jim Steel, Eelco Visser & Guido Wachsmuth

TU Delft

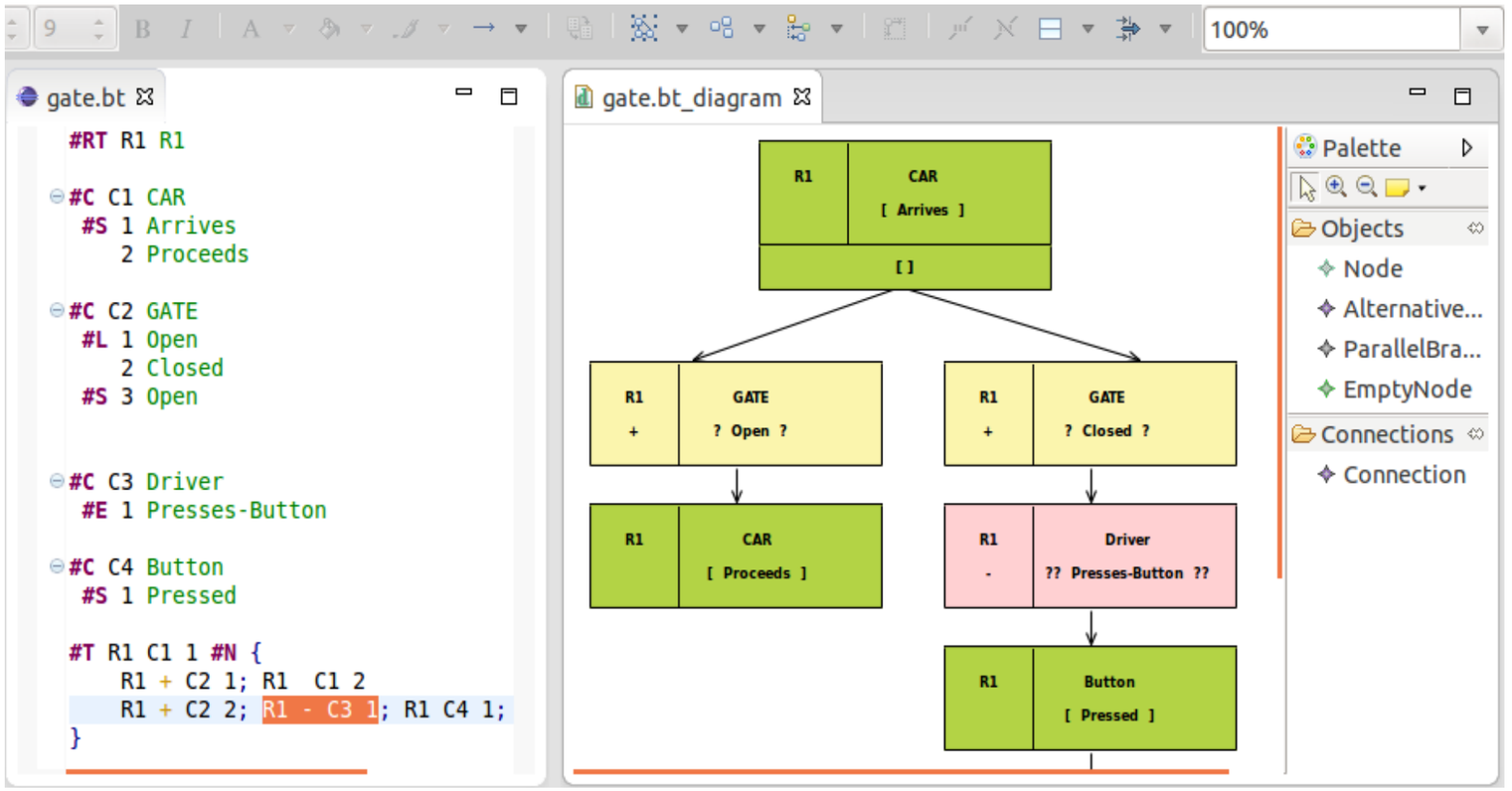
Integrating Text and Graphics

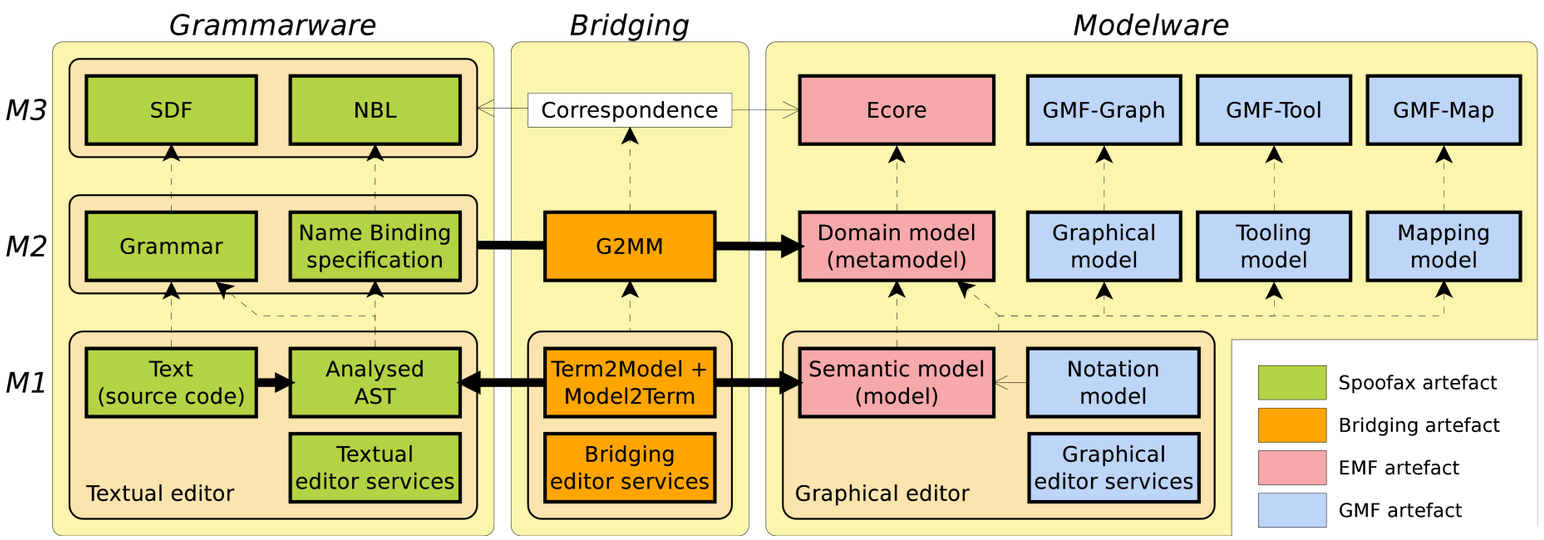
Spoofax and **GMF**
- are platforms for developing domain-specific languages with full-featured Eclipse editor plug-ins
- differ in their application domain: textual and graphical languages respectively

Spoofax.modelware
- bridges between Spoofax and GMF to support integrated textual and graphical editors
- use cases: multi-view editing, visual programming, textual modelling (e.g. **Behavior Trees**), etc.

Features

- metamodel generation (and customisation) from a textual language definition
- real-time synchronisation between text and graphics by means of generic term-to-model and model-to-term transformations
- editor services relevant to combined editors





Layout Preservation

Textual layout information consists of white space, comments, and all other information that is not part of the abstract syntax tree (AST).

Graphical layout information consists of positioning and size of nodes in the diagram, and all other information that is not part of the semantic model.

Graphical layout information needs to be preserved during text edits, and vice versa. A combination of Spoofax's layout preservation algorithm, EMFCompare and GMF's Canonical Containers is used to solve this problem.

Editor Services

Editor Services increase the productivity of programmers and modellers.

Some textual editor services
- Semantic Errors and Warnings
- Reference Resolution
- Code Completion
- Code Refactorings

Some graphical editor services
- Auto-layout
- Type and Selection Hiding
- Zooming
- Grid Snapping

Bridging editor services
- Selection Sharing
- Synchronised Save, Undo and Redo

Error Handling

Errors in text and models are inevitable during the process of programming or model construction. To allow for real-time synchronisation between text and graphics, error handling is needed.

Syntactic errors in text are handled by the parser, which applies error recovery and discards erroneous regions of code that cannot be recovered.

Syntactic errors in models cannot occur due to the way models are constructed.

Semantic errors in text can generally be propagated to the model, except for unresolved references, which are simply left out of the model until they are resolved.

Semantic errors in models can generally be propagated to the text, except for underspecified attributes and references. The generation of default values provides a solution to this problem.

The University Of Queensland AUSTRALIA

School of Information Technology & Electrical Engineering

INNOVATION EXPO