taste
The Assert Set of Tools for Engineering

Real Time Developer Studio (RTDS) tool for Verification of properties and automatic code generation

Dedicated to the development of a

modelling tool

for the development of

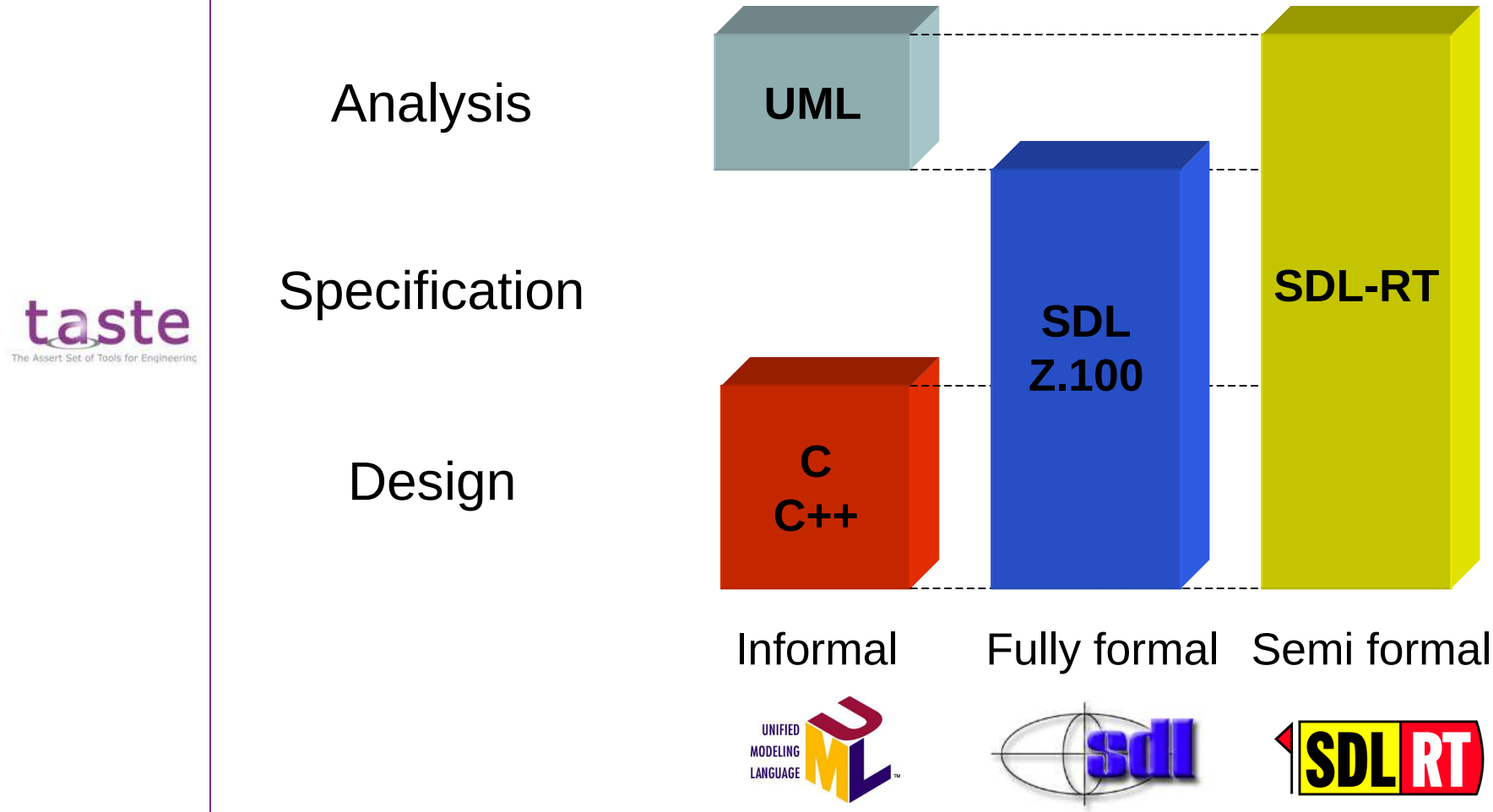
event driven software.

Specification and Description Language

SDL main characteristics:

- ITU-T standard (Z.100),
- Best modeling language for event driven systems (Telecom standards),
- Object oriented (since ’92),
- Formal (complete and non ambiguous),
- A UML profile based on SDL has been standardized,
- Increase quality by a ratio of 5,
- Reduces development time by 30%.

PragmaDev – Supported languages



PragmaDev – Modelling levels

Informal modelling for requirements: UML

- Edition
- C++ stubs generation

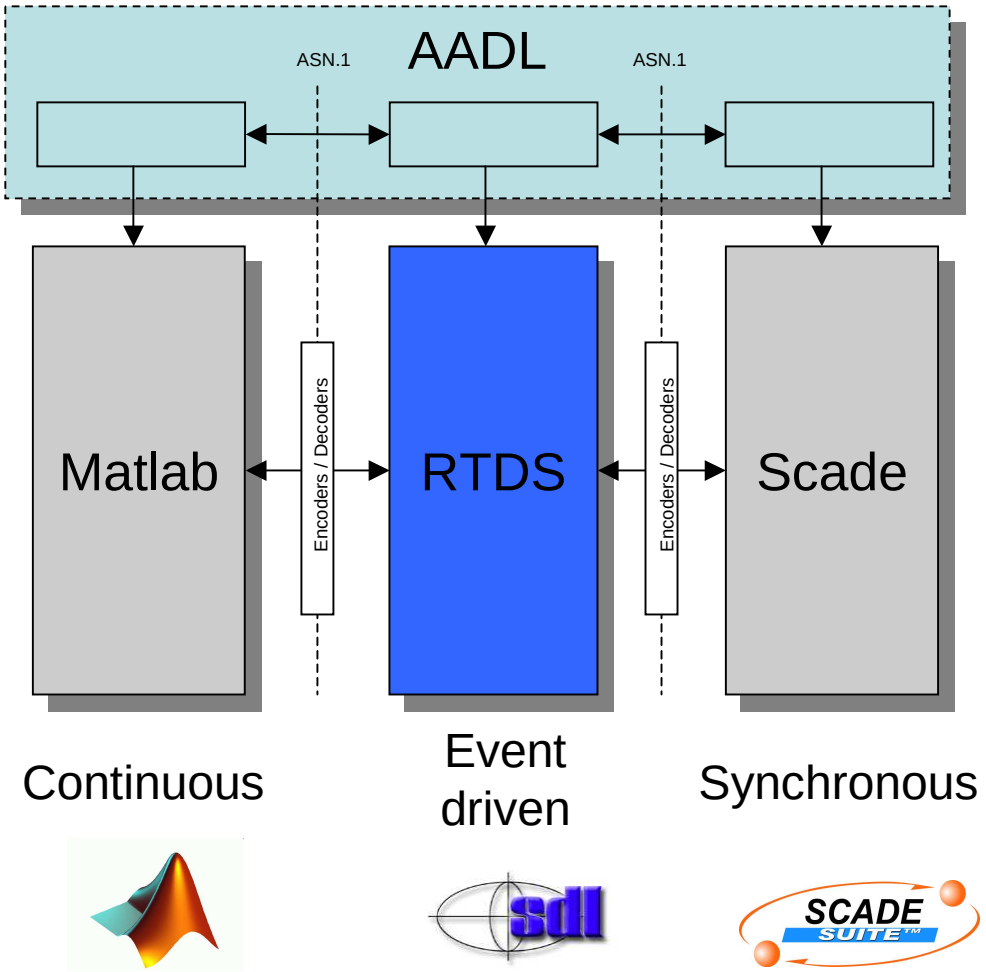
Semi-formal modelling for design: SDL-RT

- Edition
- Syntaxic et semantics checking
- Code generation
- Graphical debugging

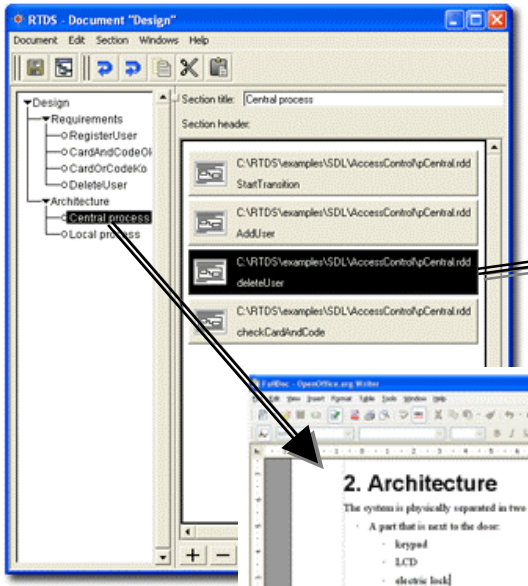
Fully formal modelling for specification: SDL Z.100

- Edition
- Syntaxic et semantics checking
- Simulation
- Verification
- Code generation
- Graphical debugging
- Test

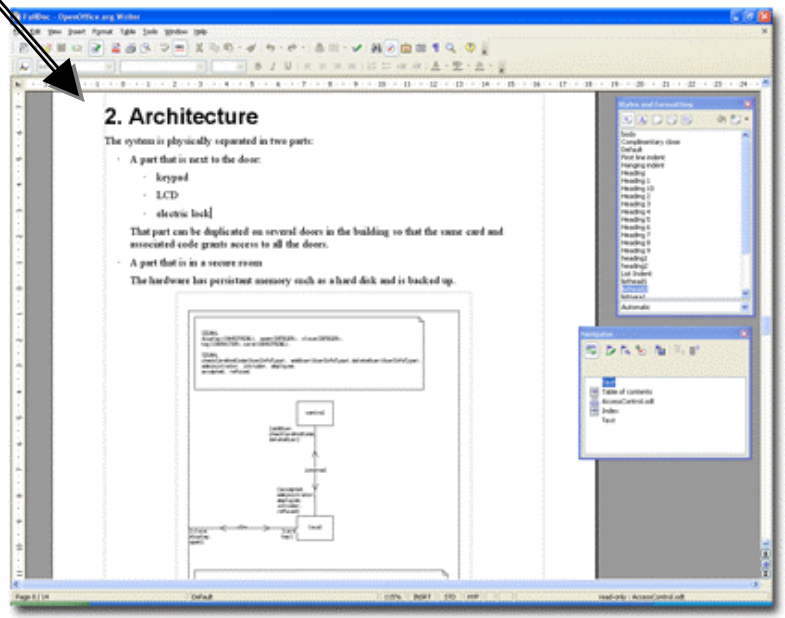
PragmaDev – Positionning in Taste framework



PragmaDev – Documentation generation

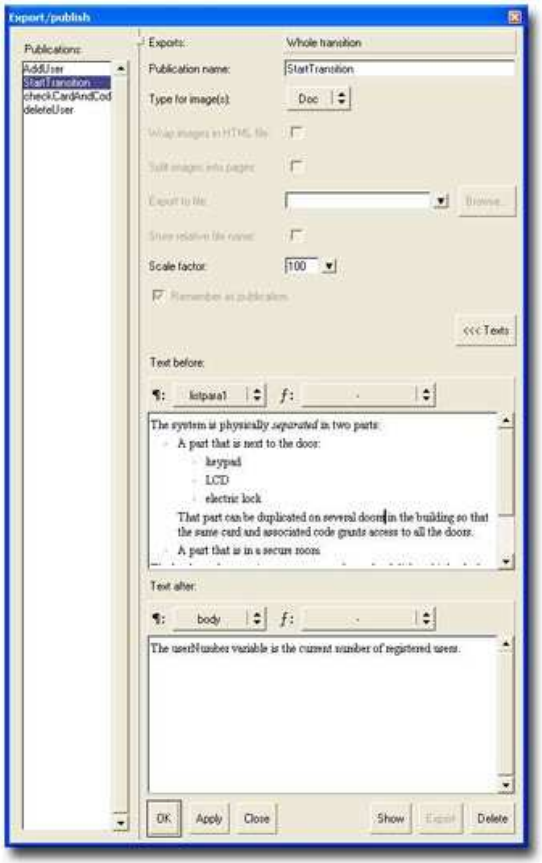


A document is made of logical publications



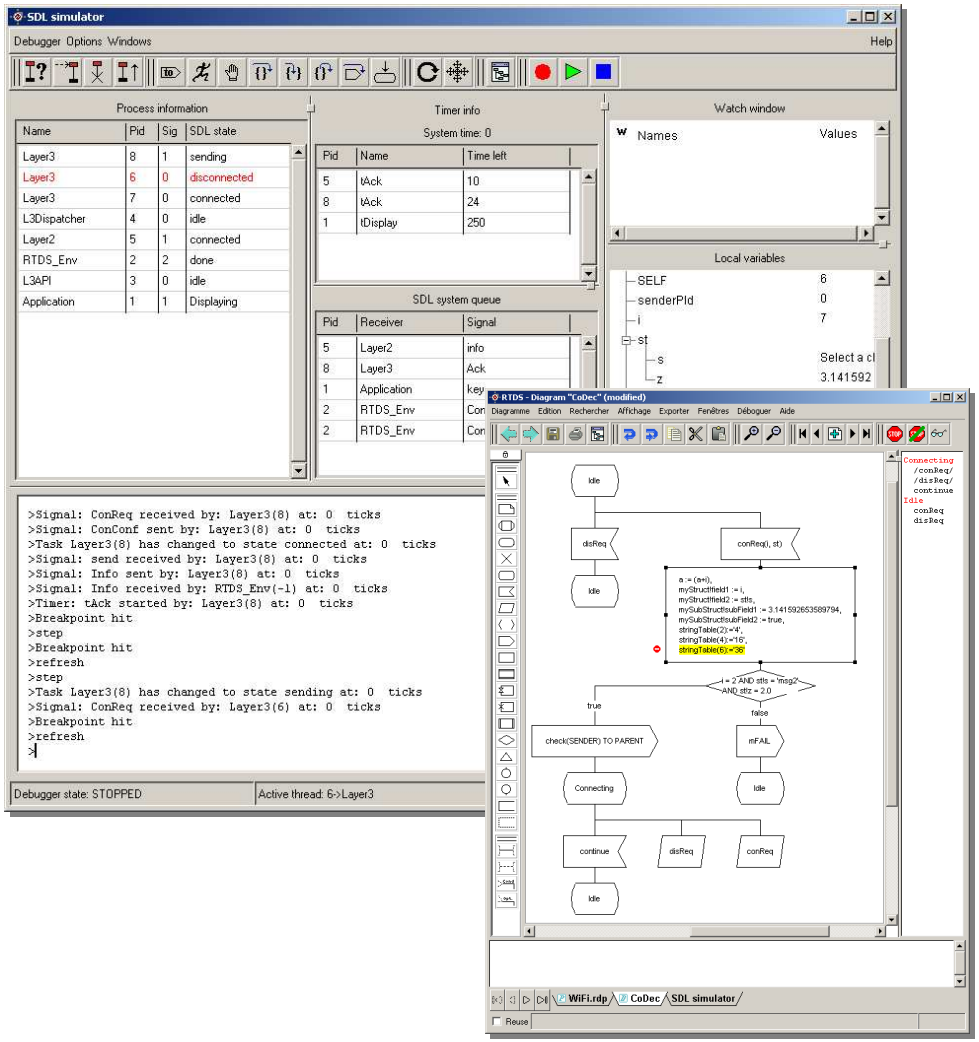
The generated documentation

RTF, OpenDocument, HTML, SGML including table of content and index entries



A publication

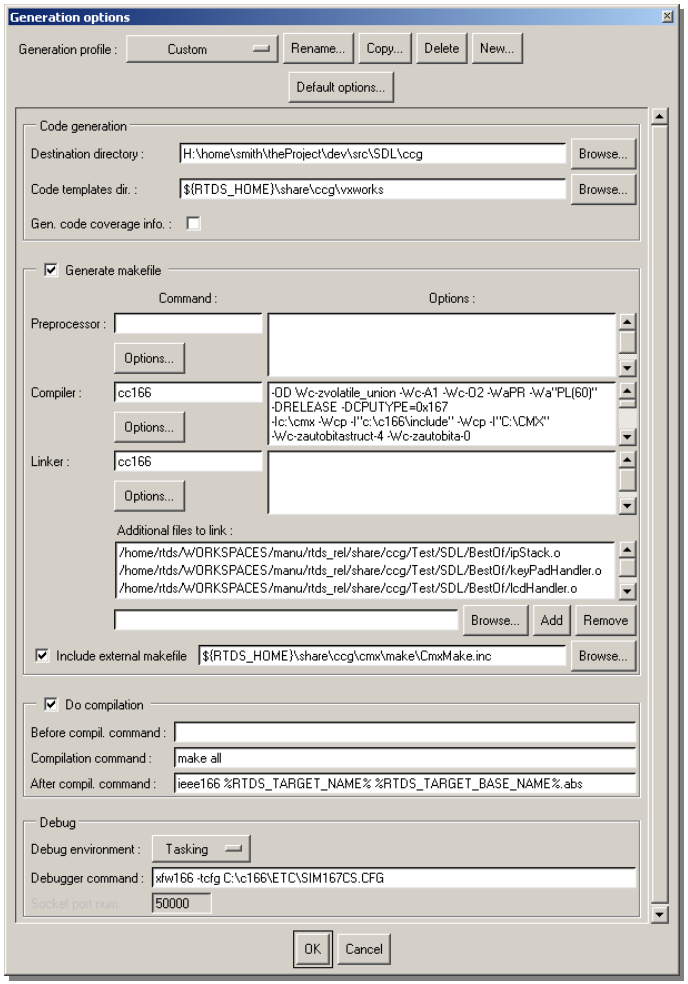
PragmaDev – Model simulation



A graphical *debugger* for fully formal models based on the model semantic

- Set breakpoints and step in the diagrams,
- Externally defined or interactive operator calls,
- Dynamic traces,
- Connecting an external tool is possible through a socket.

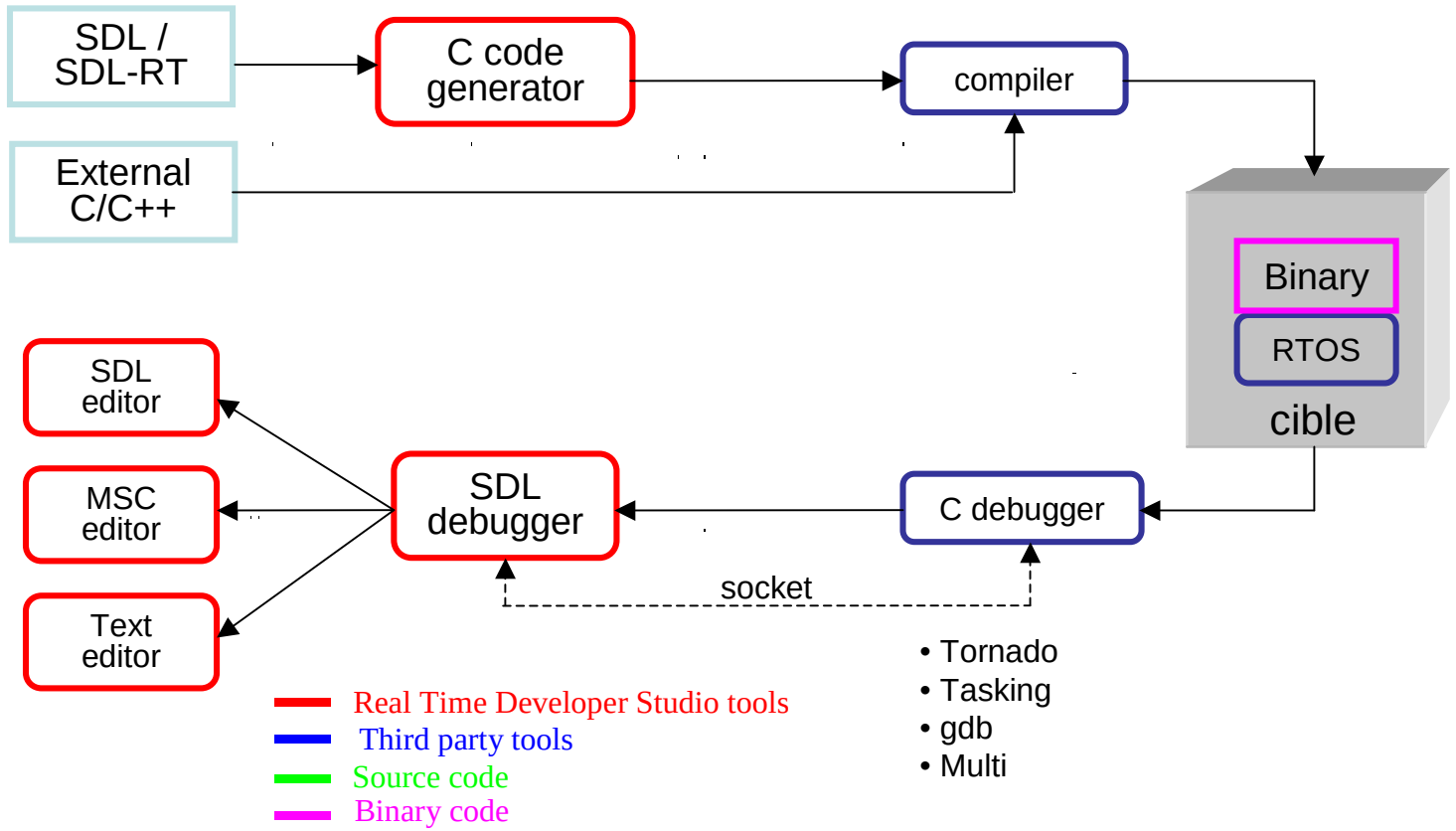
PragmaDev – Code generation options



- Control of memory allocation,
- C scheduler provided,
- Generated code is legible,
- Generation profile wizard,
- Royalty free,
- Documented for customization.

PragmaDev – Code generation and debug

The Model debugger relies on a traditional C debugger or cross debugger to provide graphical debugging.



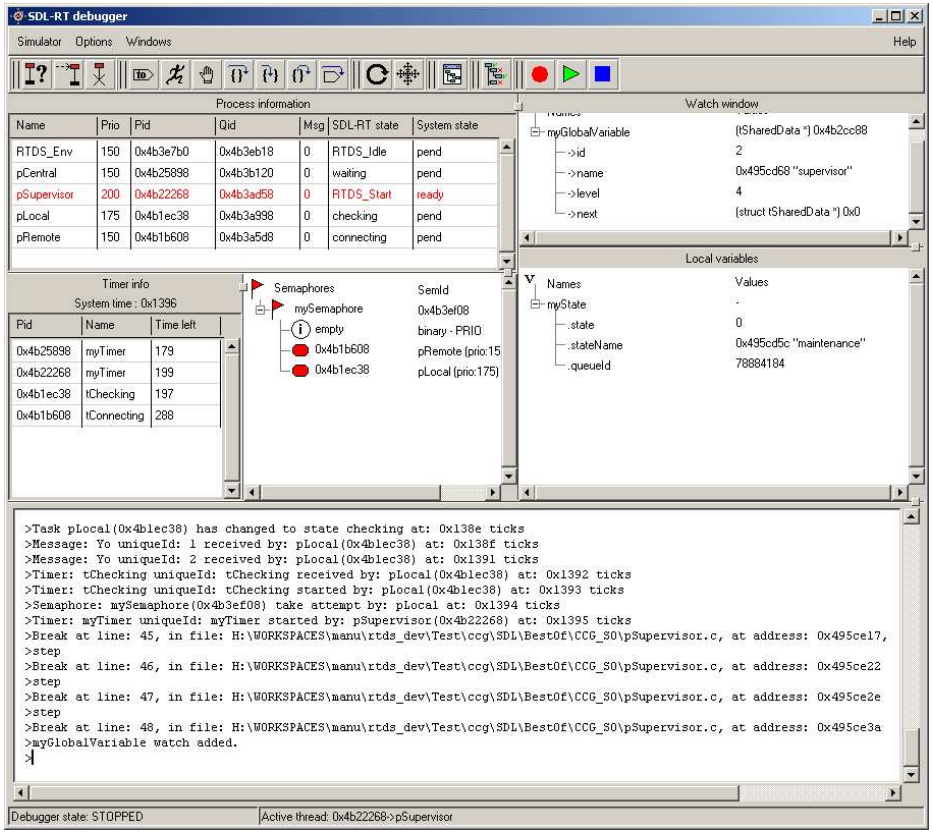
PragmaDev – Model debugger

Depending on the integration model:

- Relies on the target semantic: processor and RTOS,
- Relies on the SDL semantic.

Debug in the model:

- Breakpoints, stepping, in the diagrams or in the generated C files,
- Dynamic MSC traces,
- Connecting an external tool is possible through a socket.



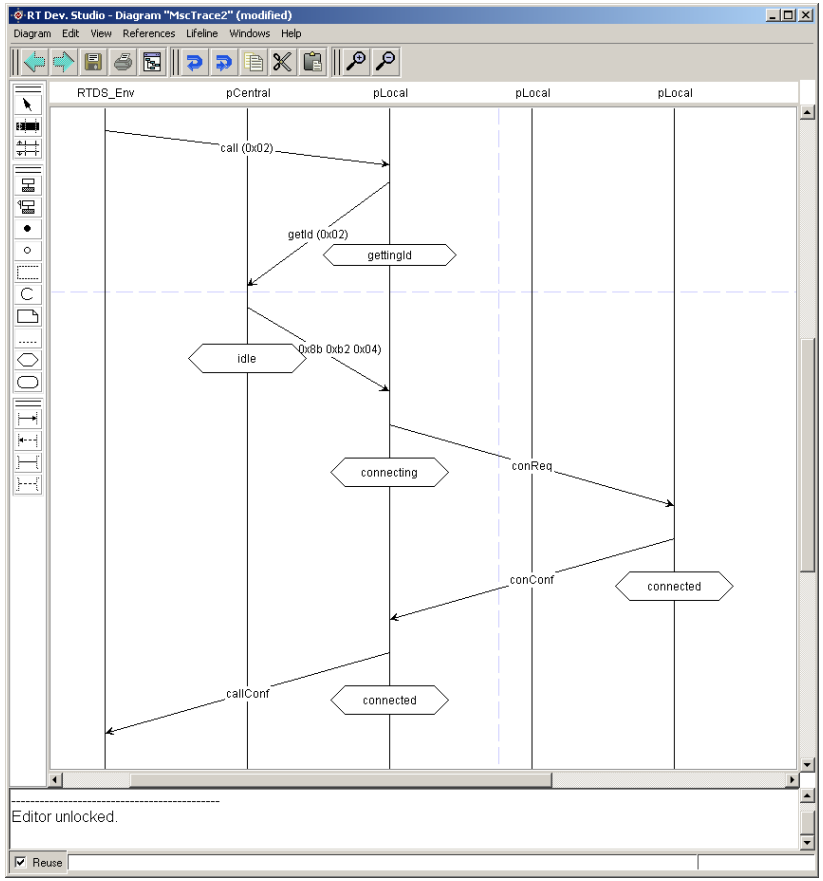
PragmaDev – Message Sequence Chart

Execution traces:

- States,
- Events,
- Semaphores,
- Timers.

Trace level configuration

Display of system time

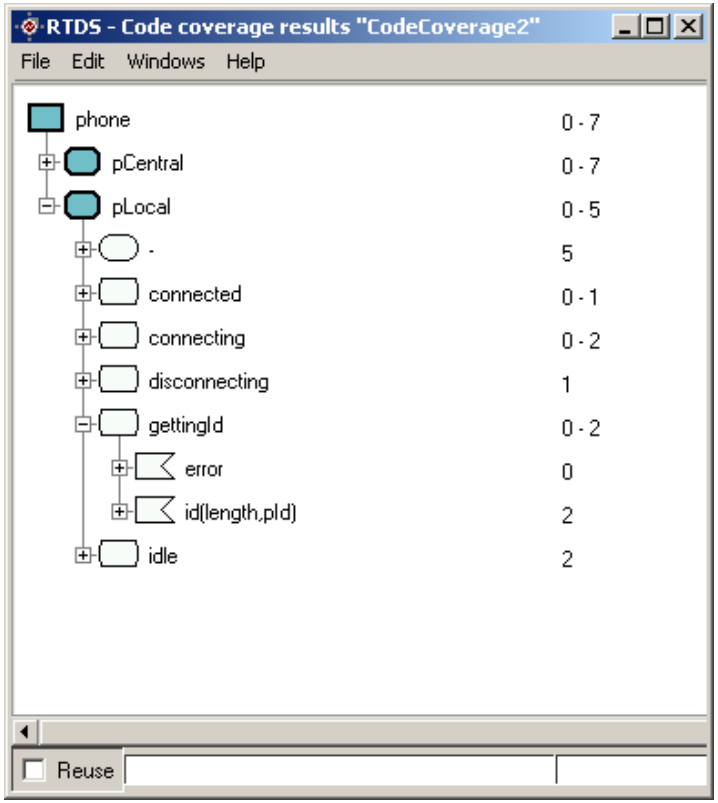


MSC Diff allows to check:

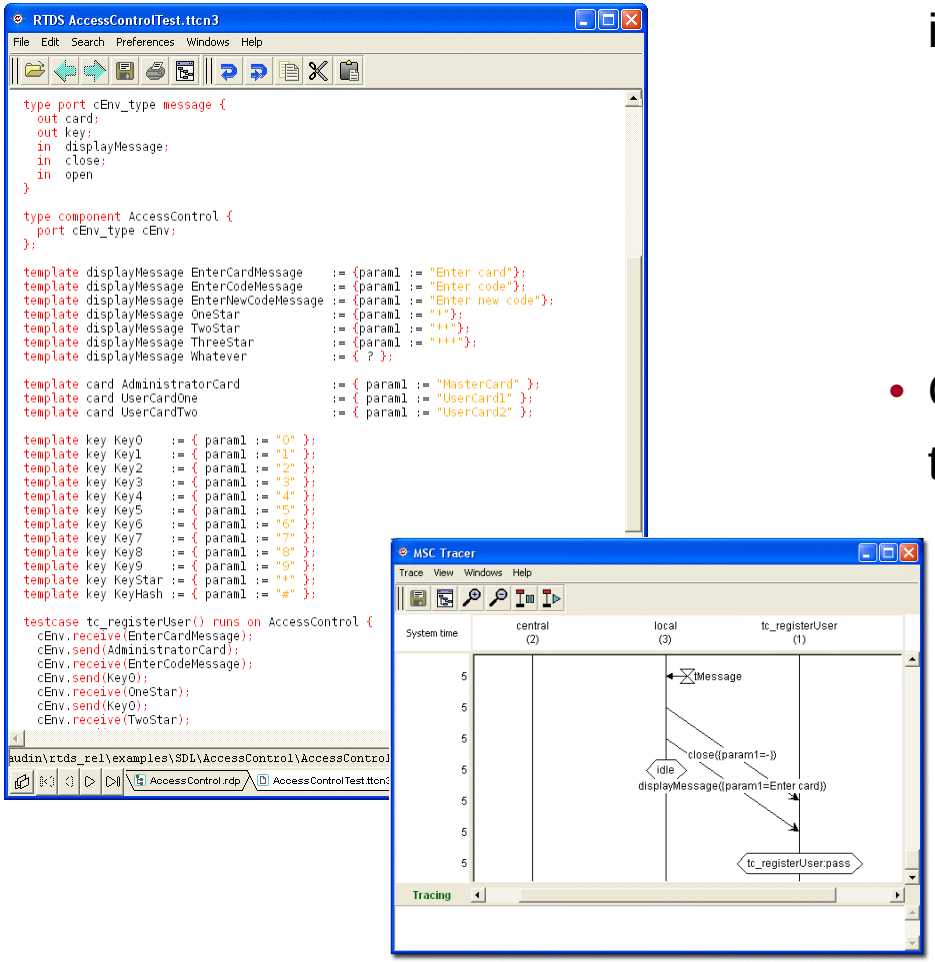
- Conformity,
- Non-regression.

PragmaDev – Model coverage

- Graphical model coverage analysis
- Merge feature



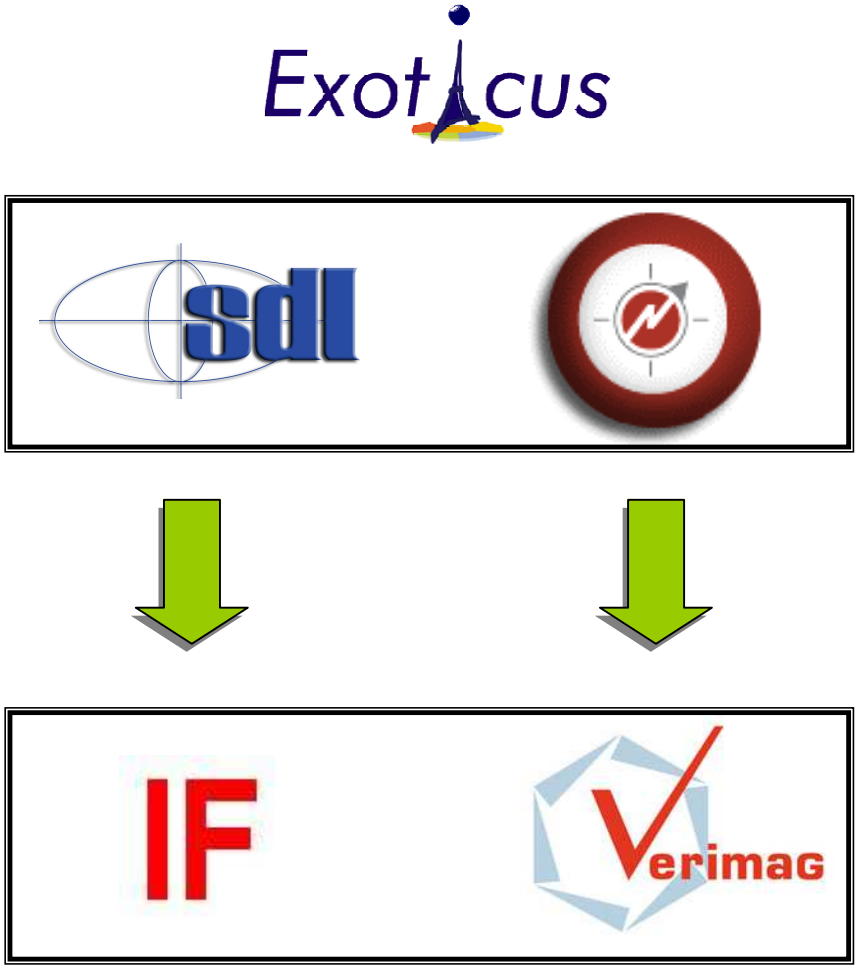
PragmaDev – Model based testing



- Based on TTCN-3 international standard:
  - Data types definitions,
  - Templates definitions,
  - Test cases,
  - Execution control.
- Connects automatically to the Simulator:
  - Breakpoints in the model or in the test suite,
  - Verdict displayed in the trace.

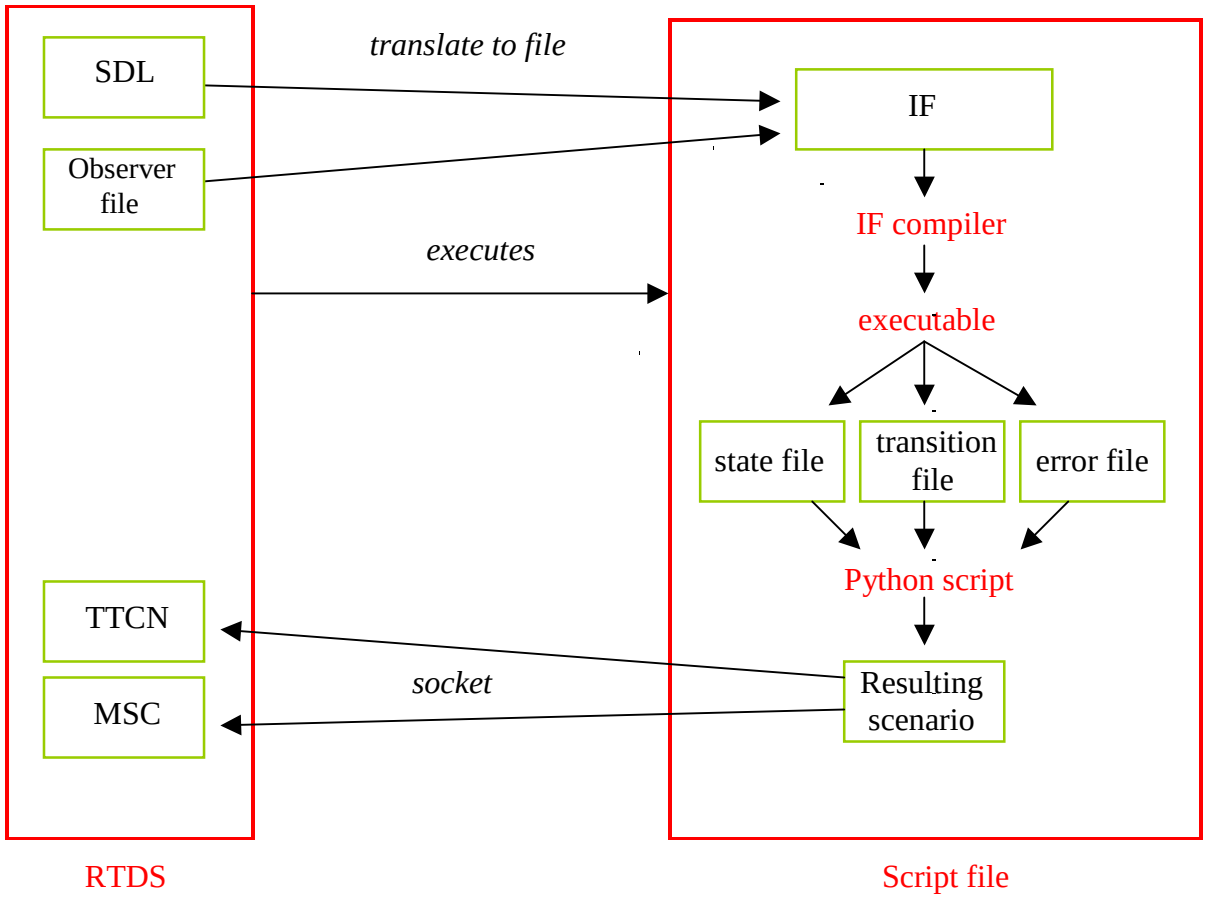
PragmaDev – Model checking

- Partnership with Verimag on IF technology.
  - Exhaustive simulation,
  - Observers,
  - Test generation.
- RTDS feature
  - Export to IF,
  - Execute a script
  - Generate an MSC feedback.



taste
The Assert Set of Tools for Engineering

PragmaDev – External tool integration



PragmaDev – Conclusion

- Three levels of modelling:
  - Informal,
  - Semi-formal,
  - Formal.
- Tools to:
  - Document,
  - Simulate,
  - Validate,
  - Test.
- Based on international standards.
- Integrated in Taste framework.