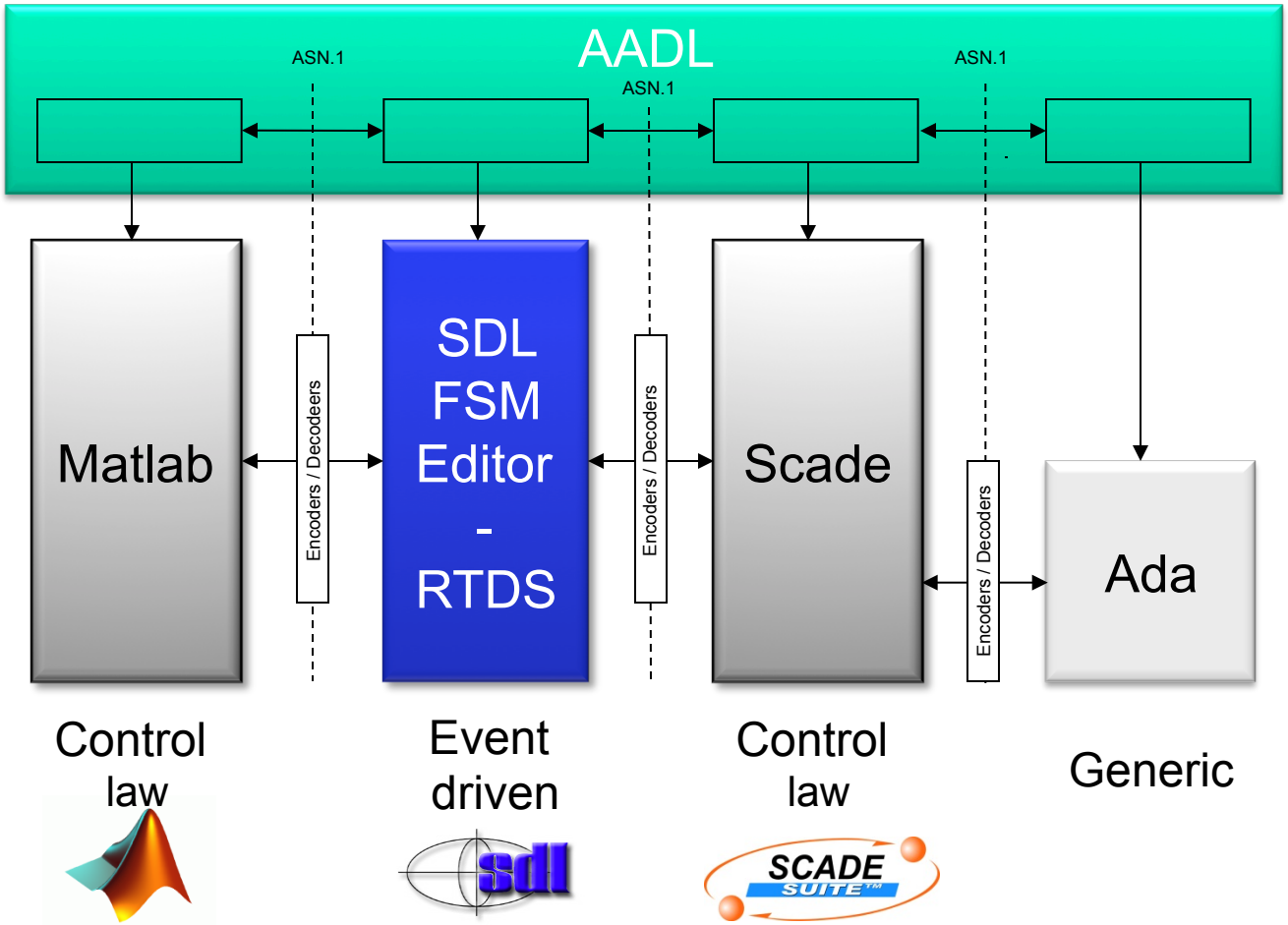
PRAGMADEV
real time development tools

PragmaDev SDL FSM Editor & Real Time Developer Studio within the TASTE framework

ESA – May 22th, 2014

Eric Brunel
eric.brunel@pragmadev.com

ESA Taste framework integration



taste
The Assert Set of Tools for Engineering

esa

- Specification and Description Language, ITU-T standard intended to write the detailed specifications of the protocols in order to make sure they will interoperate.
- Major updater every 4 years since 1976.
- Major releases:
  - SDL 1988: First mature version
  - SDL 1992: Object orientation
  - SDL 2000: UML alignment
  - SDL 2010: C types support in Z.104
- Anual conference
  - SDL Forum (http://www.sdl-forum.org/)
  - SAM workshop (satellite event of Models)
- 11 commercial tools, 10 public domain tools.
- Integrated technology in TASTE ESA framework.

SDL: features

- SDL graphical abstraction (architecture, communication, behavior, services) fits the needs.
- SDL being formal, it is possible to simulate the model.
- SDL being formal, partial or full code generation is possible.
- SDL being object oriented, software components are reusable.
- System are globally asynchronous (GALS) so SDL can be used at system level.
- SDL has the characteristics to describe a good PIM.
- SDL is recognized by certification authorities (European Aviation Safety Agency Certification Memorandum, ETSI, ESA)

SDL: the figures

Years of experience allows to quantify gains of SDL usage.

- C code: 35 to 50 mistakes per 1000 lines
- SDL code: 8 mistakes per 1000 lines

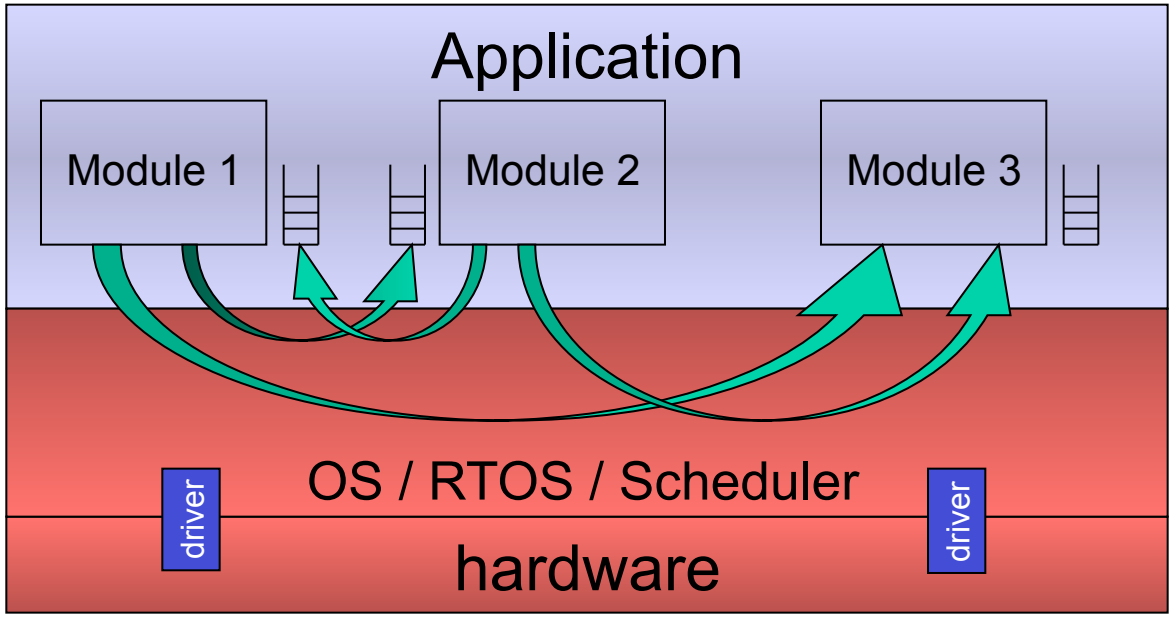
- Development time is globally reduced by 35%
  - Reduced up to 50% in the left branch of the V cycle
  - Less gain on the right side of the V because of the gap with technical reality

Target segment

Event driven systems

Embedded & Real time

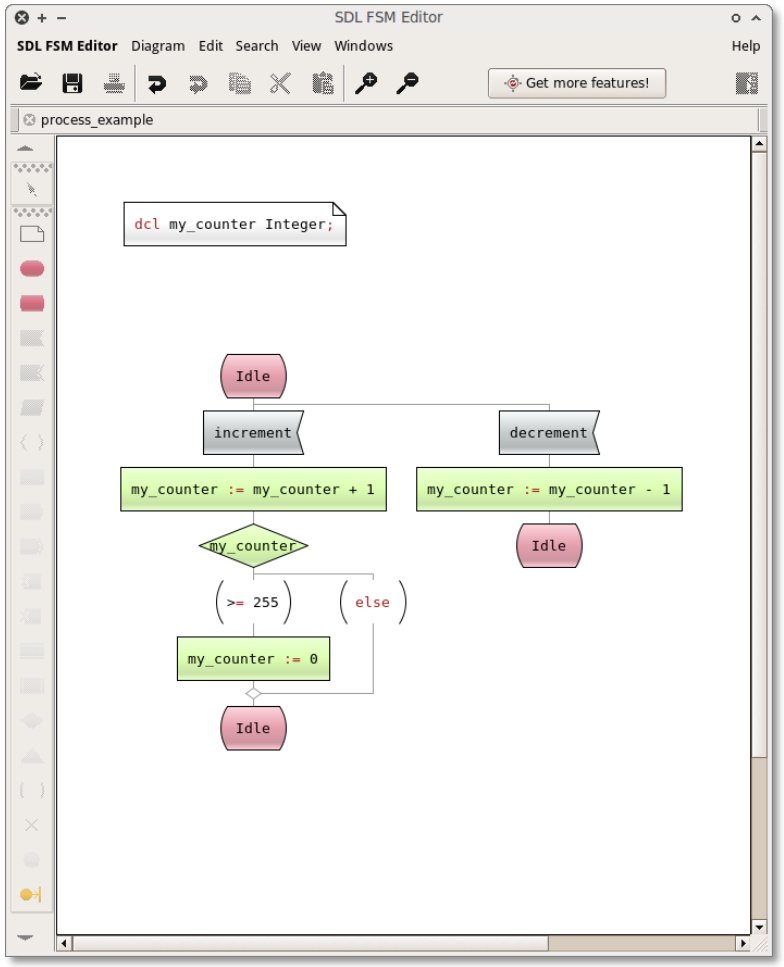
- Decomposed in tasks running concurrently
- Communicating via messages (sporadic or cycling in TASTE)



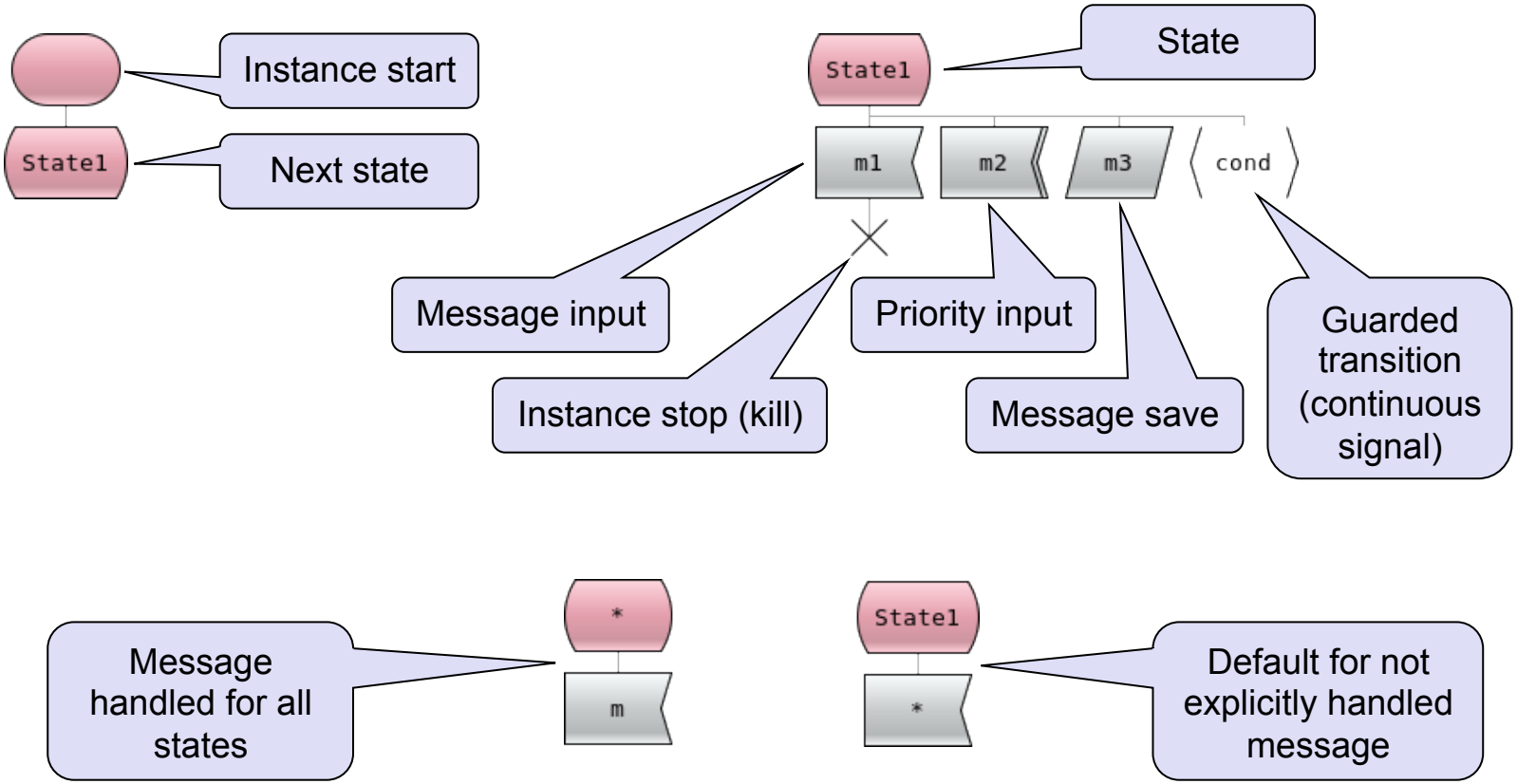
PRAGMADEV
real time development tools

SDL FSM Editor

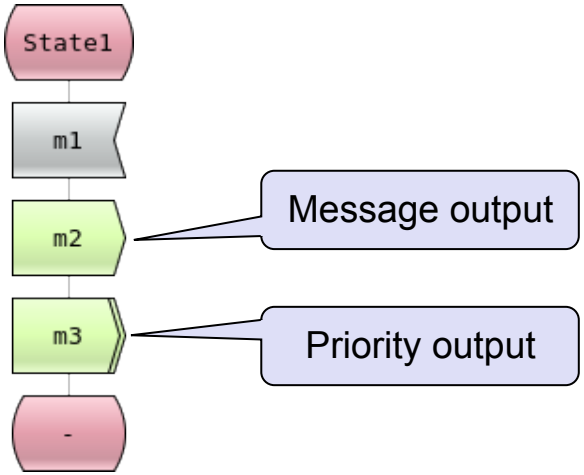
- ASN.1 or SDL abstract data types.
- Syntax highlighting
- Code completion (limited)
- Context-sensitive action availability
- Logical-unit based editing
- Operation preview

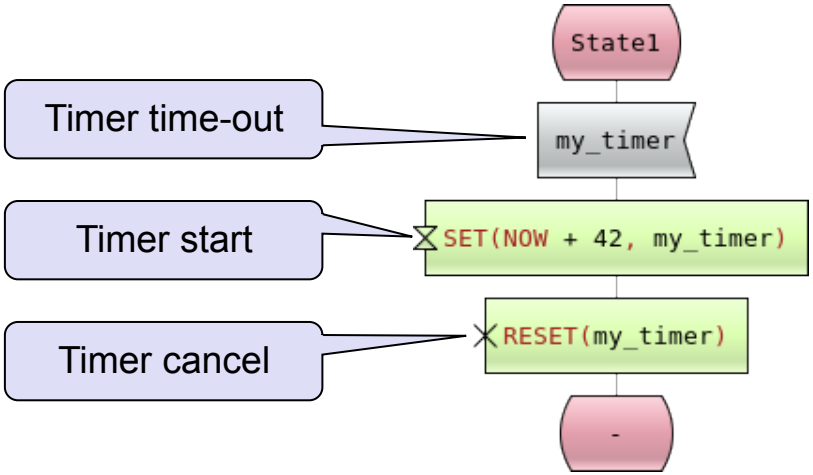


SDL concepts in processes (1)

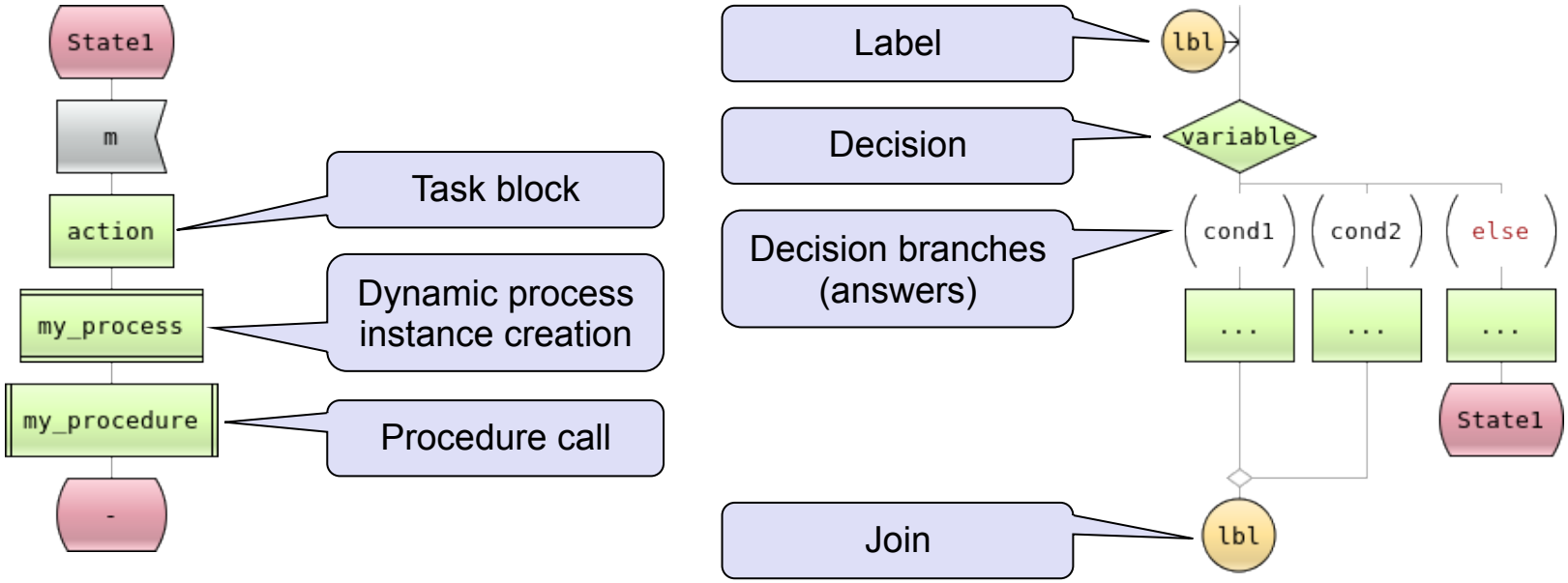


SDL concepts in processes (2)



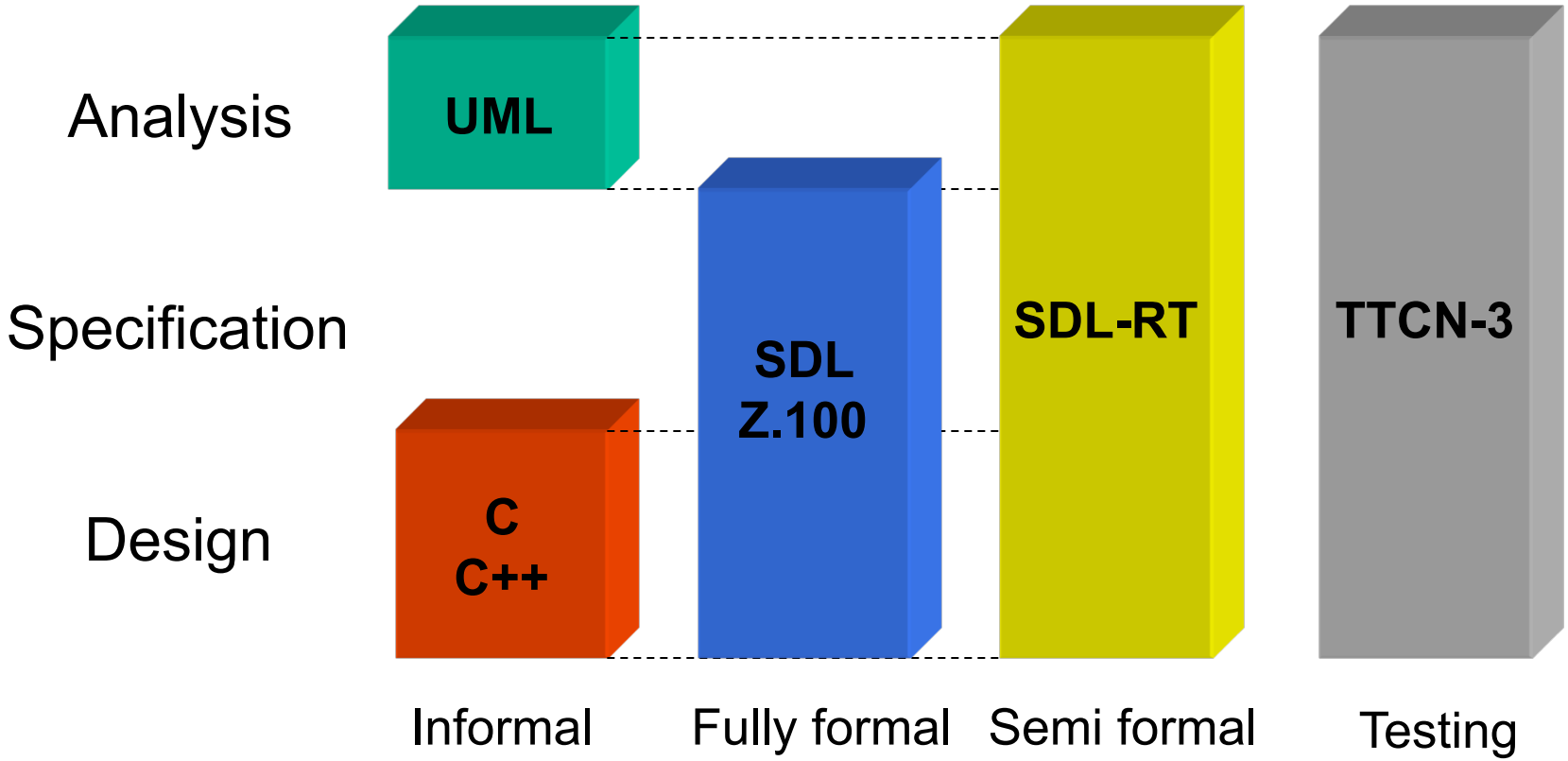


SDL concepts in processes (3)



More “standard” control flow statements available in task blocks (`for`, `break` / `continue`, `if`, …).

RTDS: supported languages



RTDS: supported languages

Informal modelling for requirements: UML

- Edition
- C++ stubs generation

Semi-formal modelling for design: SDL-RT

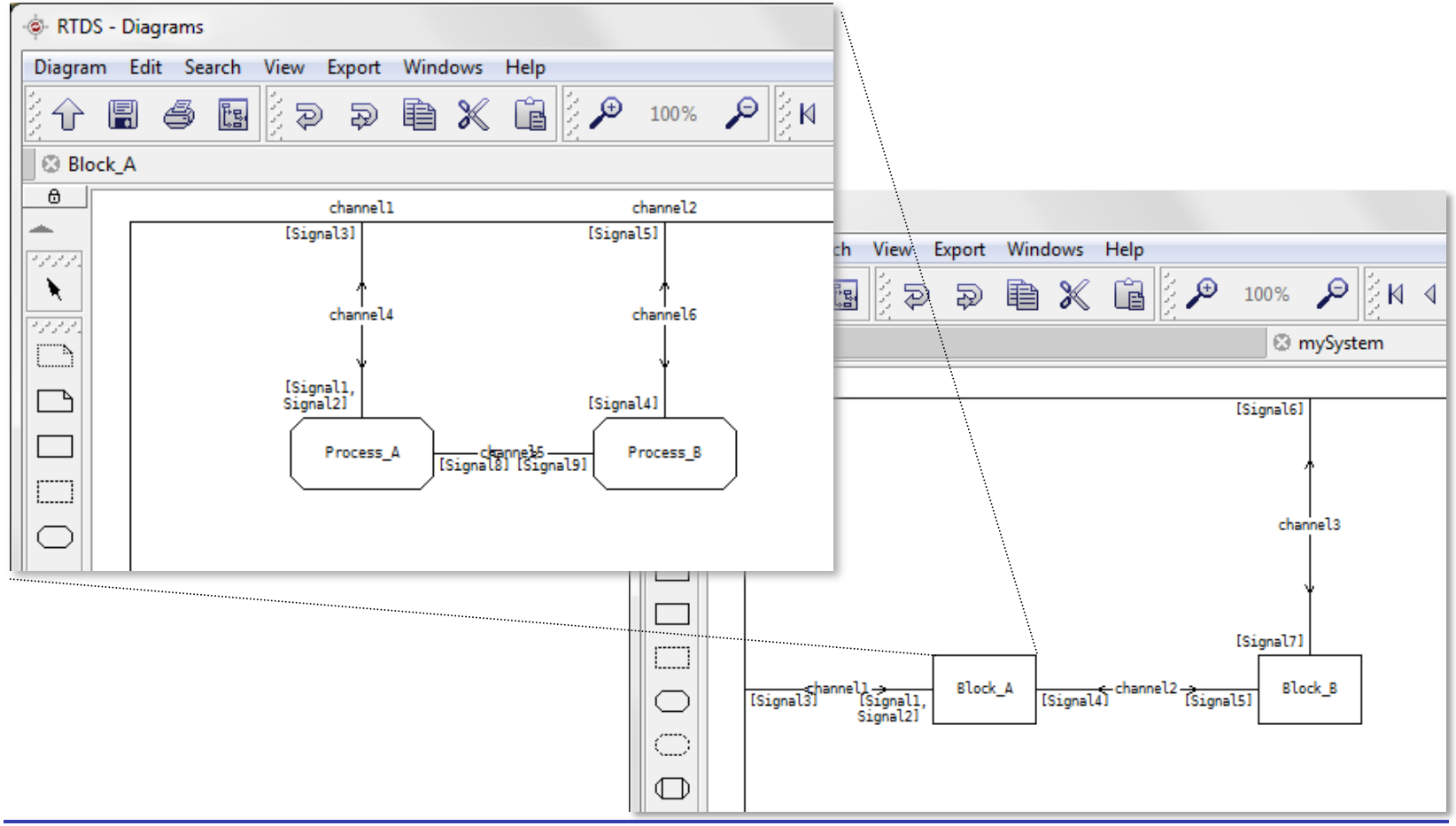
- Edition
- Syntaxic et semantics checking
- Code generation
- Graphical debugging

Fully formal modelling for specification: SDL Z.100

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

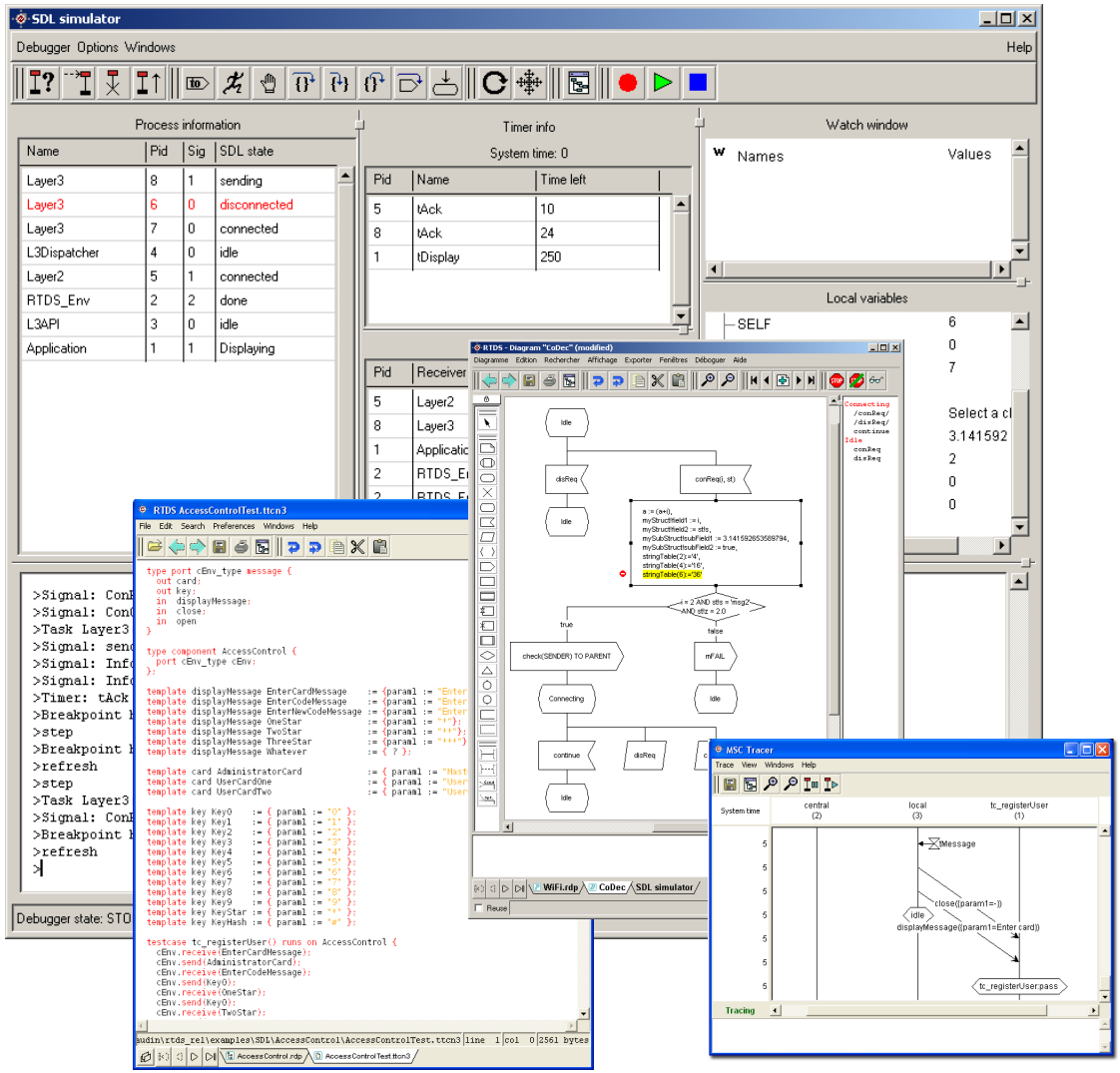
PRAGMADEV
real time development tools

Architecture and communication



real time developer studio

PRAGMADEV
real time development tools



Model simulator

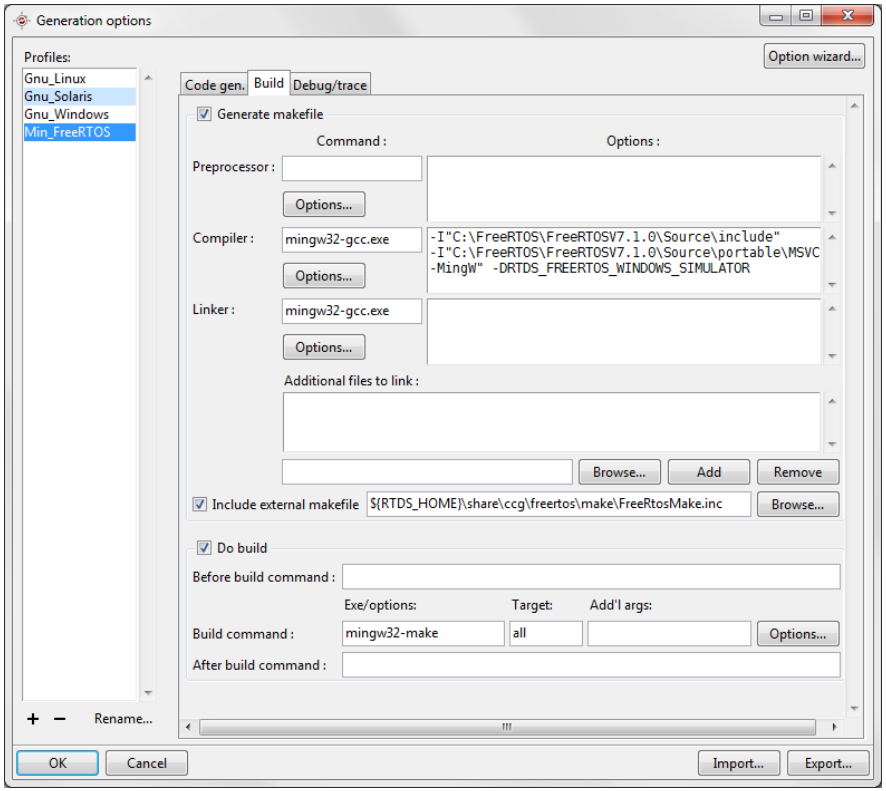
A graphical *debugger* for fully formal models and TTCN-3 test cases

- Set breakpoints and step in the model,
- Dynamic traces.

real time developer studio

PRAGMADEV
real time development tools

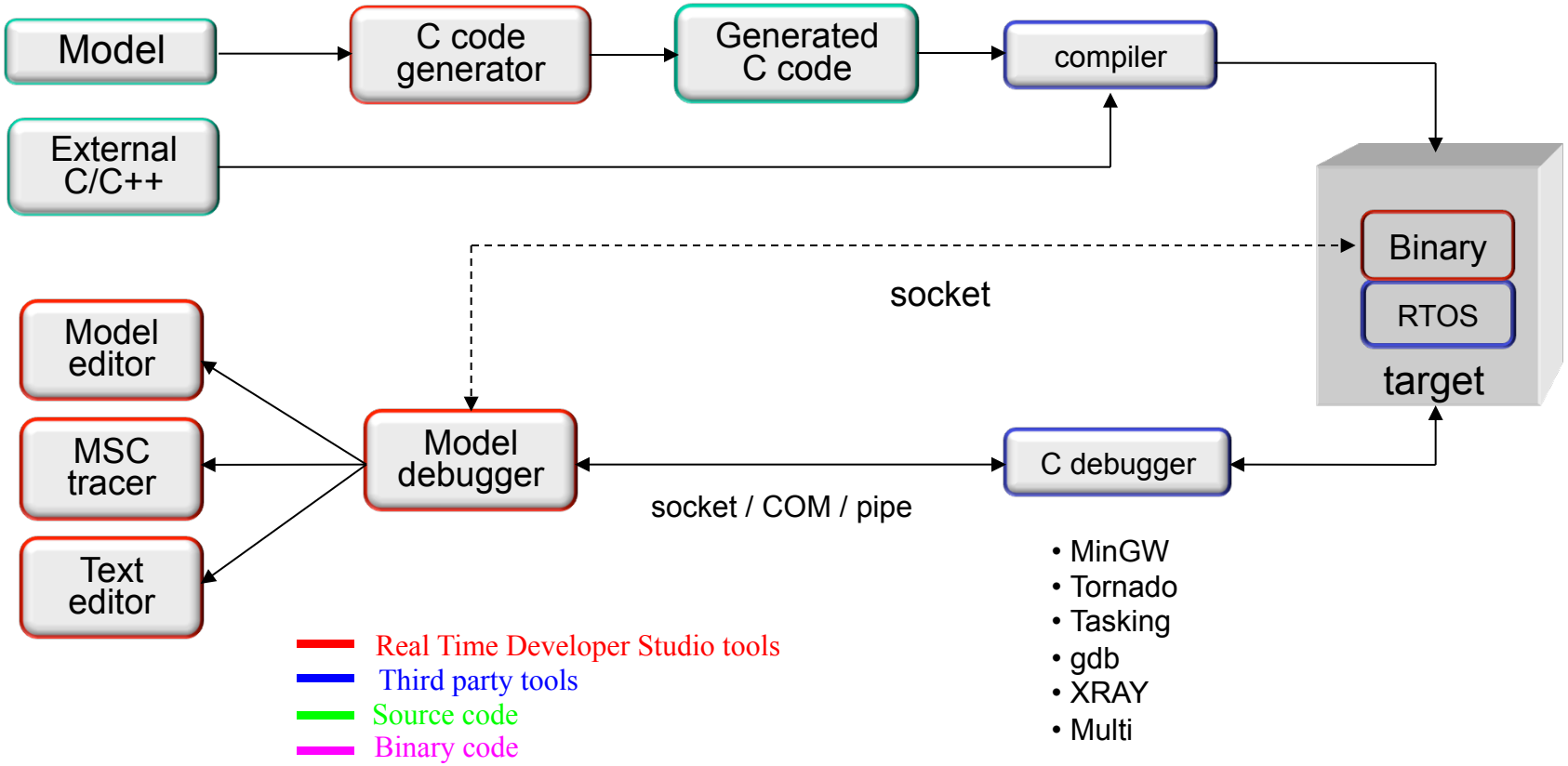
Code generation



- C++ skeleton for static classes
- C or C++ for dynamic classes
- Generated code is legible
- Generation profile wizard
- The code is:
  - Integrated with: FreeRTOS, VxWorks, OSE, OSE Epsilon, CMX RTX, Nucleus, uiTRON, Posix, ThreadX, and Win32,
  - Provided with an scheduler,
  - Royalty free,
  - Documented for customization.

Debugging architecture

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



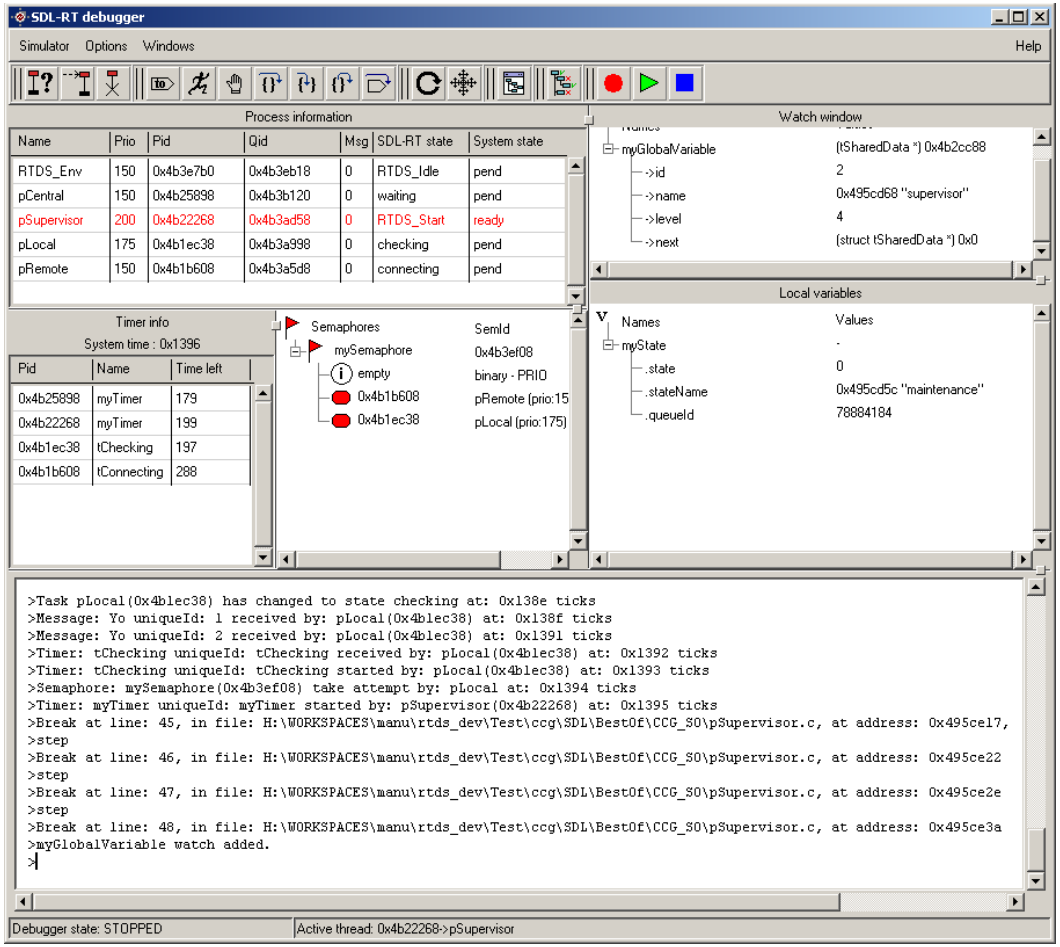
PRAGMADEV
real time development tools

Model debugger

Relies on the target semantic: processor and RTOS.

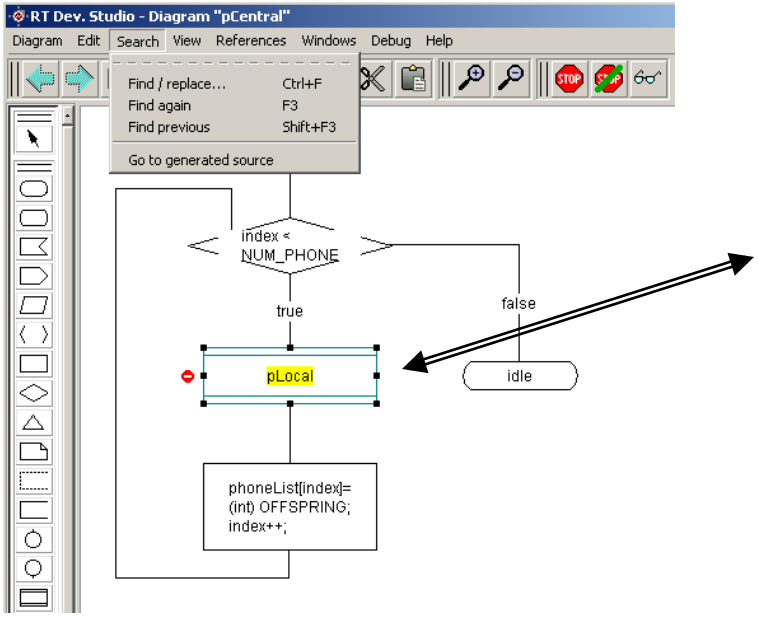
Debug in the model:

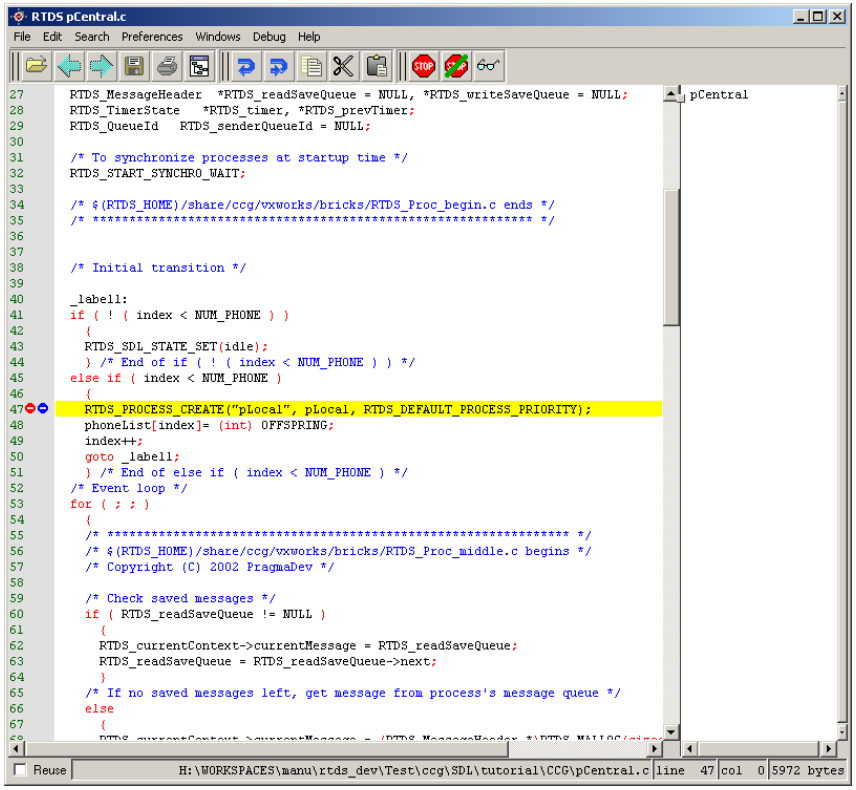
- Breakpoints, stepping, in the SDL/RT diagrams or in the generated C files,
- Dynamic MSC traces,
- Connecting an external tool is possible through a socket.



Debug features

- Switch between
  - Model
  - Generated C/C++ code





Graphical traces

Execution traces:

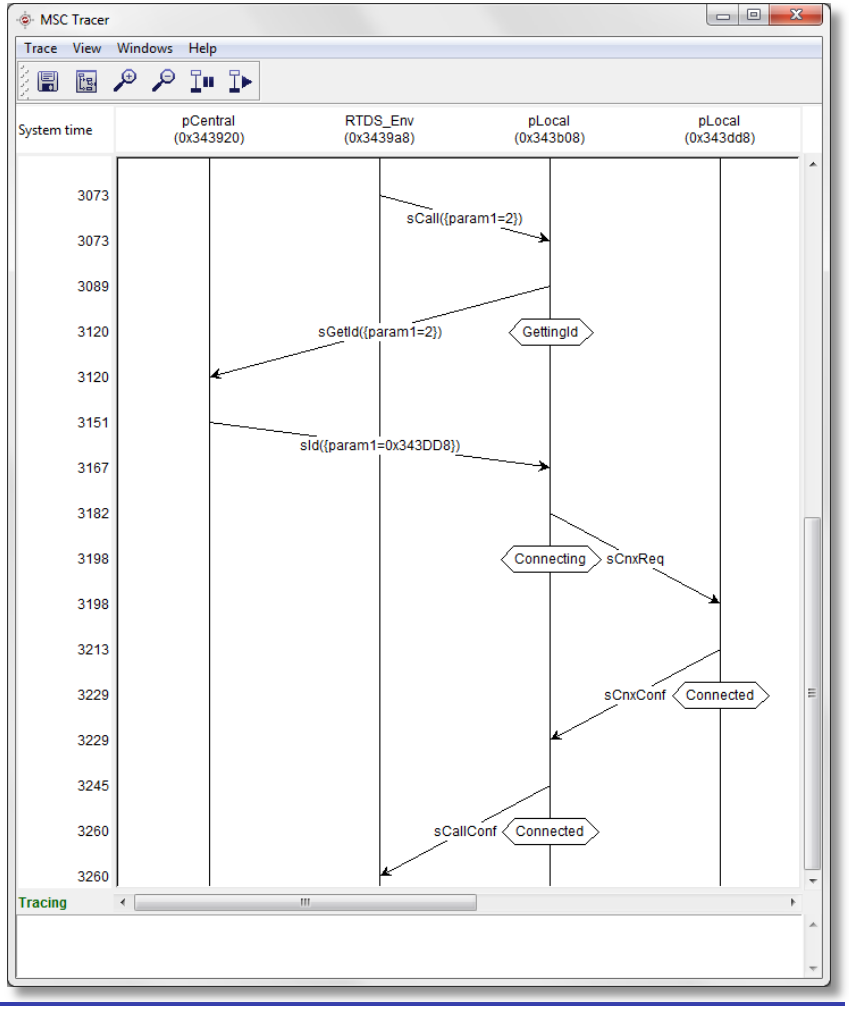
- States,
- Events,
- Semaphores,
- Timers.

Trace level configuration

Display of system time

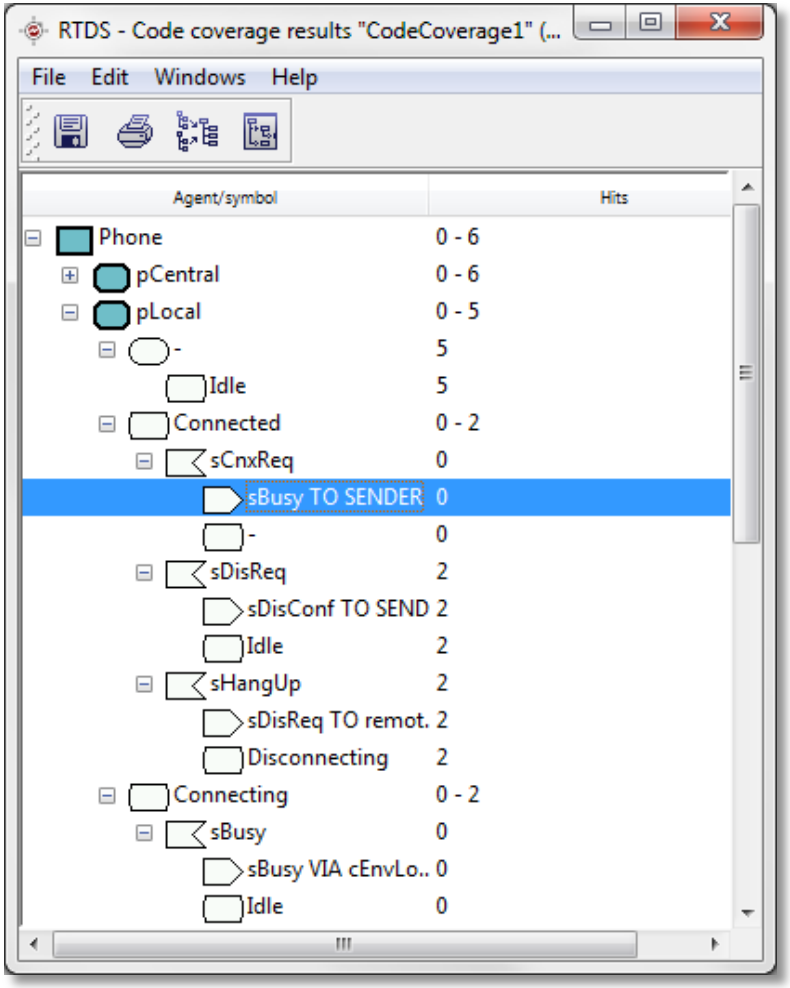
MSC Diff allows to check:

- Conformity,
- Non-regression.



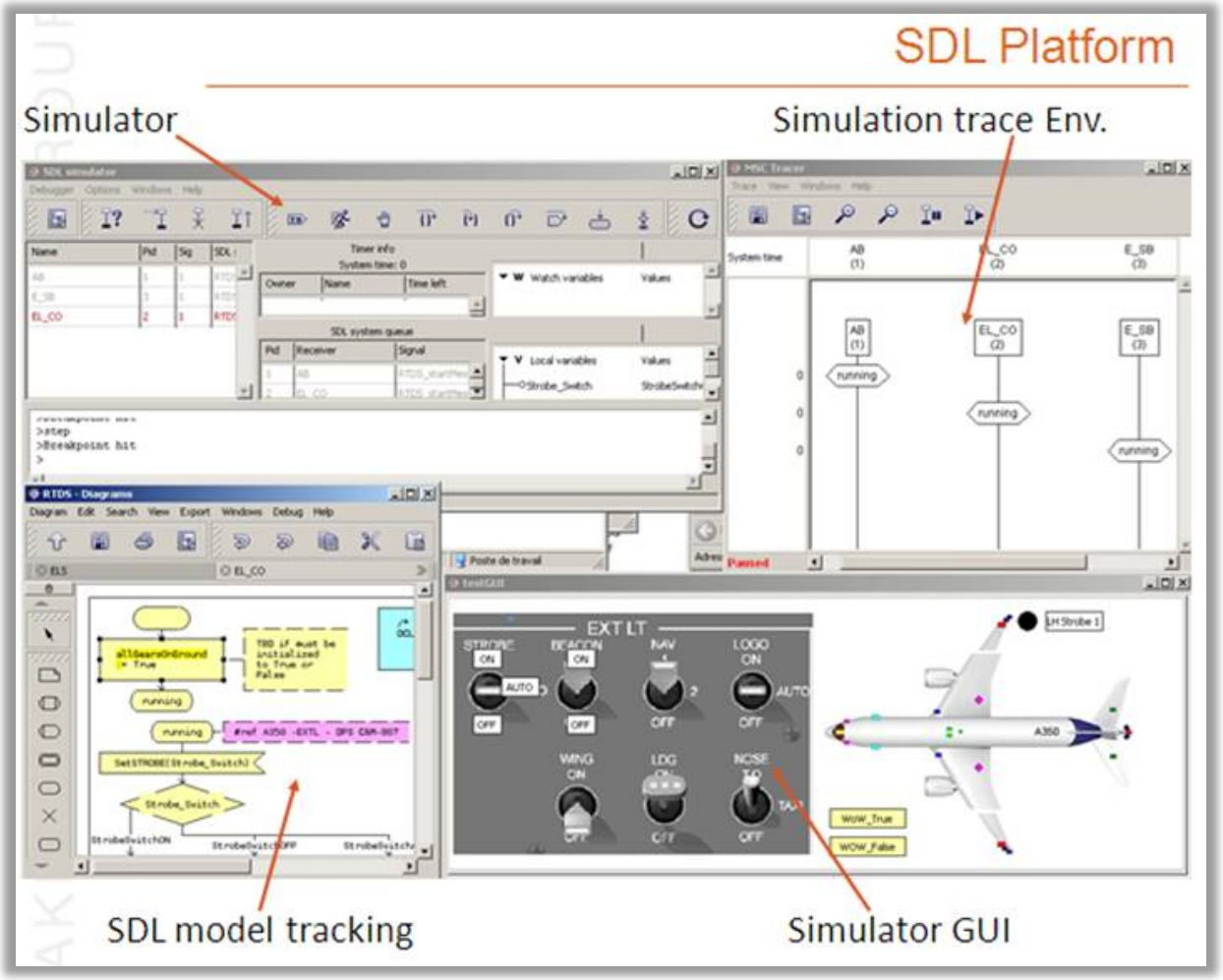
Model coverage

- Graphical model coverage analysis
- Merge feature



PRAGMADEV
real time development tools

Prototyping interface



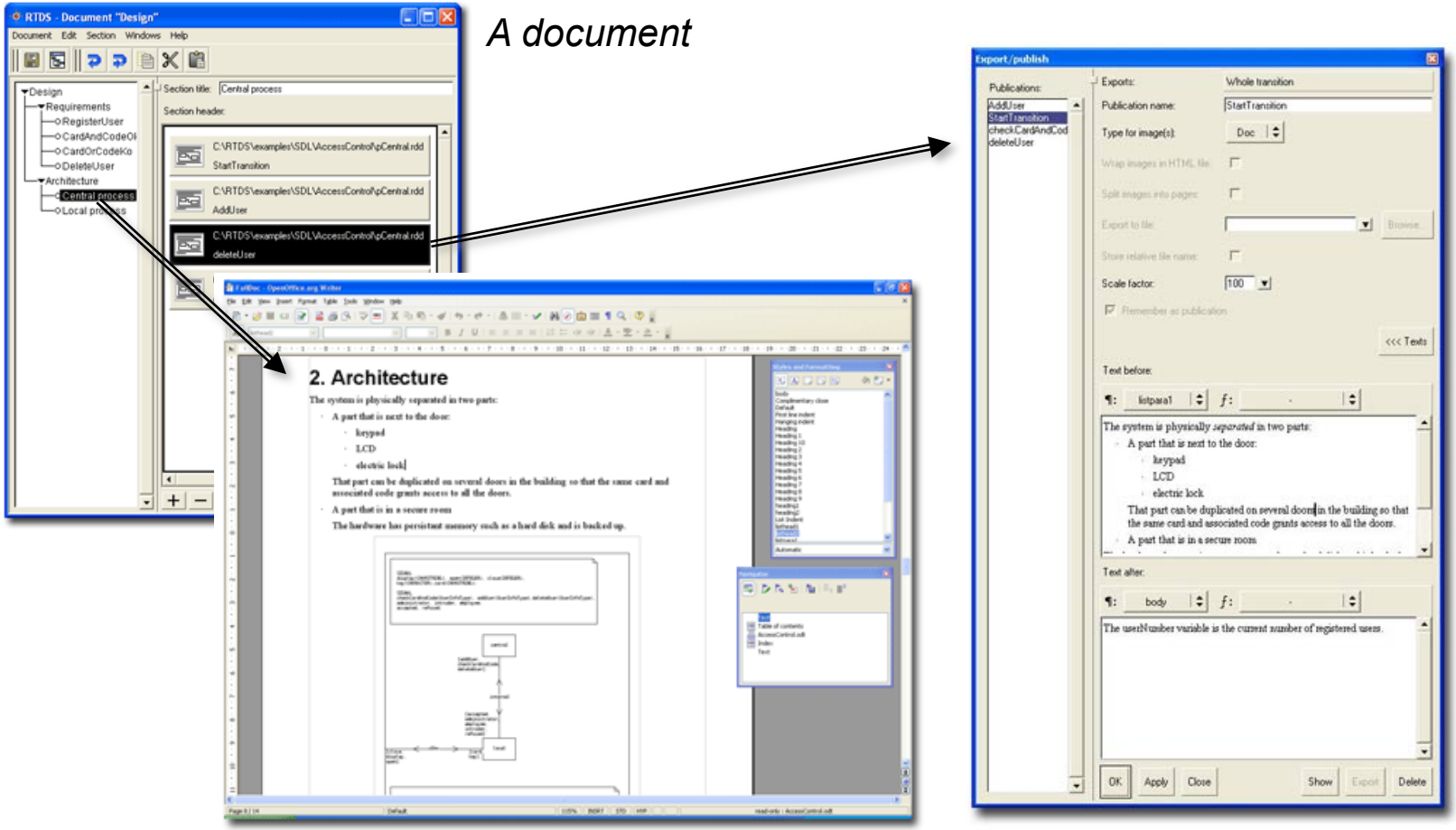
- Knows about the model inputs and outputs.
- Connects automatically to the simulator or the debugger.

Documentation generation

- Logical publications (state, transition, partition, diagram)
- Comments preceeding or following the publication
  - Styles for paragraphs
  - Styles for characters
- Export format
  - RTF
  - OpenDocument
  - HTML
  - SGML
- Exported elements
  - Texts with publications
  - Index entries
  - Table of contents entries

PRAGMADEV
real time development tools

Documentation generation



A document

The generated documentation

A publication

real time developer studio

Standard testing language

- Relies on basic services
  - Messages
  - Procedures
  - Timers
  - Parallel execution
- Based on TTCN-3 international standard:
  - Data types definitions or ASN.1,
  - Templates definitions,
  - Test cases,
  - Verdict,
  - Execution control.

TTCN-3
Testing & Test Control Notation

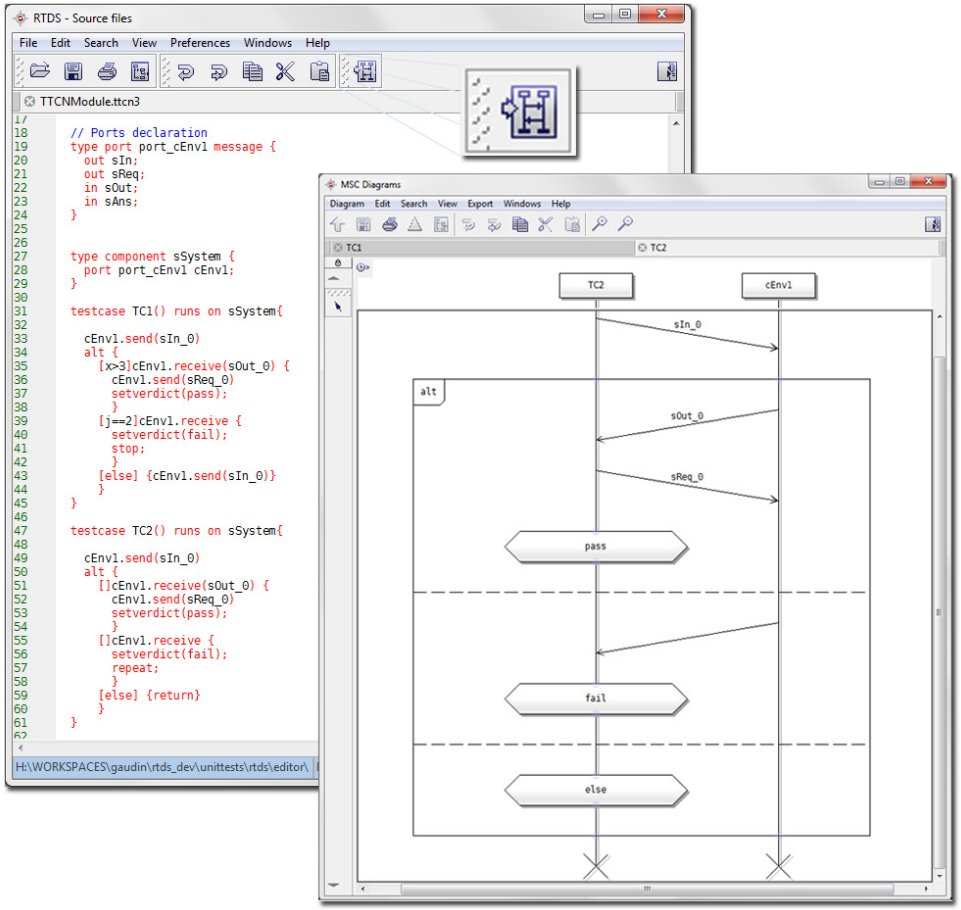
ETSI
World Class Standards

ITU
International Telecommunication Union

PRAGMADEV
real time development tools

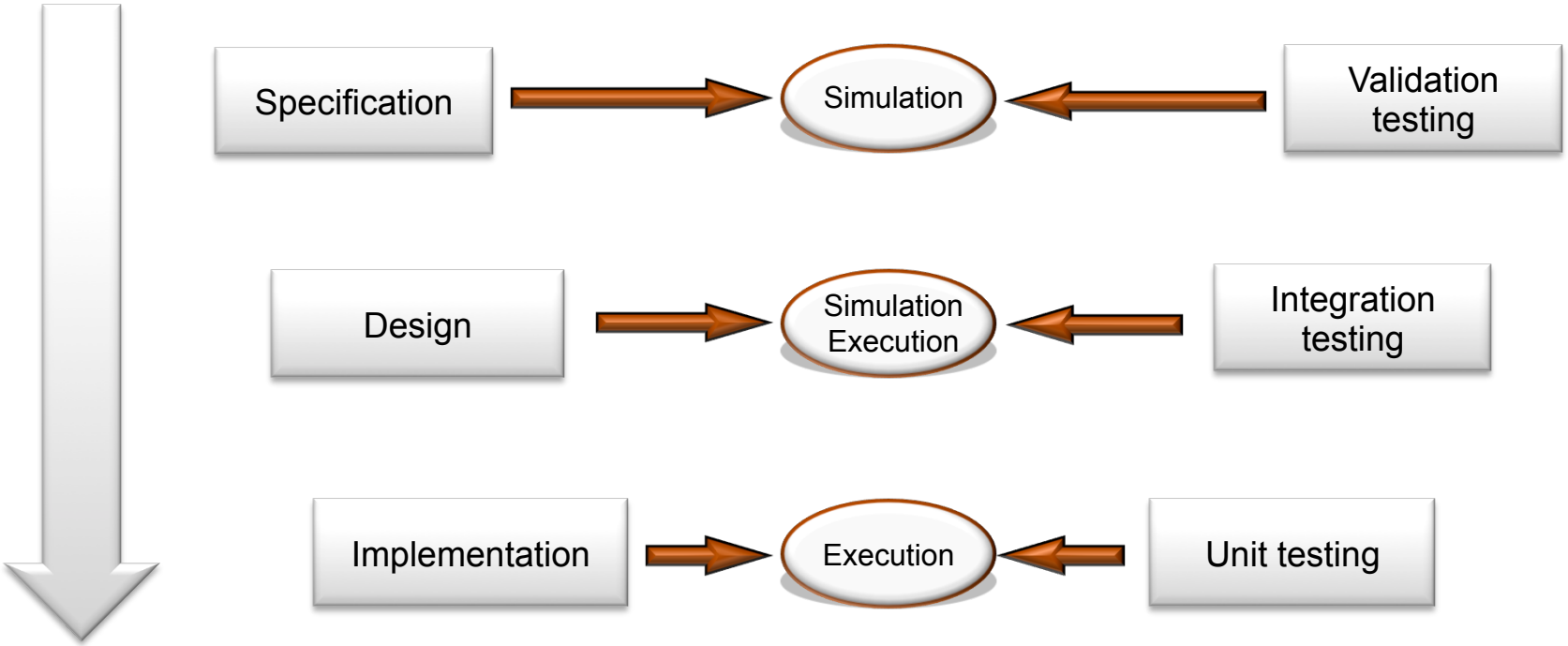
TTCN-3 support

- Textual language
- Simulator with Test manager
- C++ code generator
- TTCN-3 to MSC generation
- MSC to TTCN-3 generation
- TTCN-3 generation from a property on the model (Verimag)
- TTCN-3 generation based on model coverage (to come)



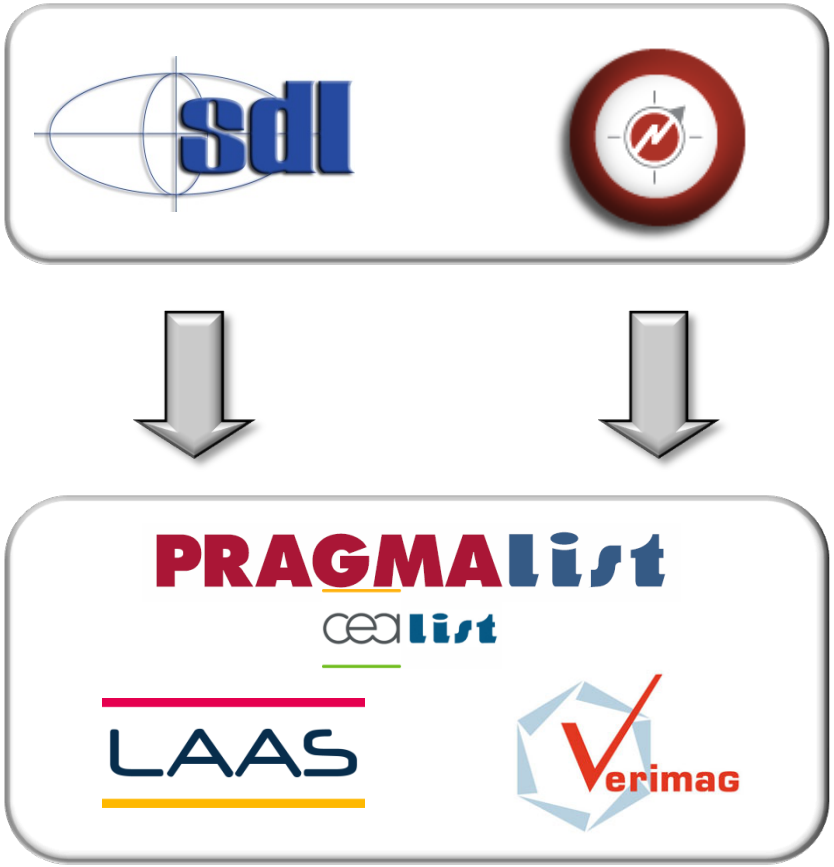
real time developer studio

Continuous integration

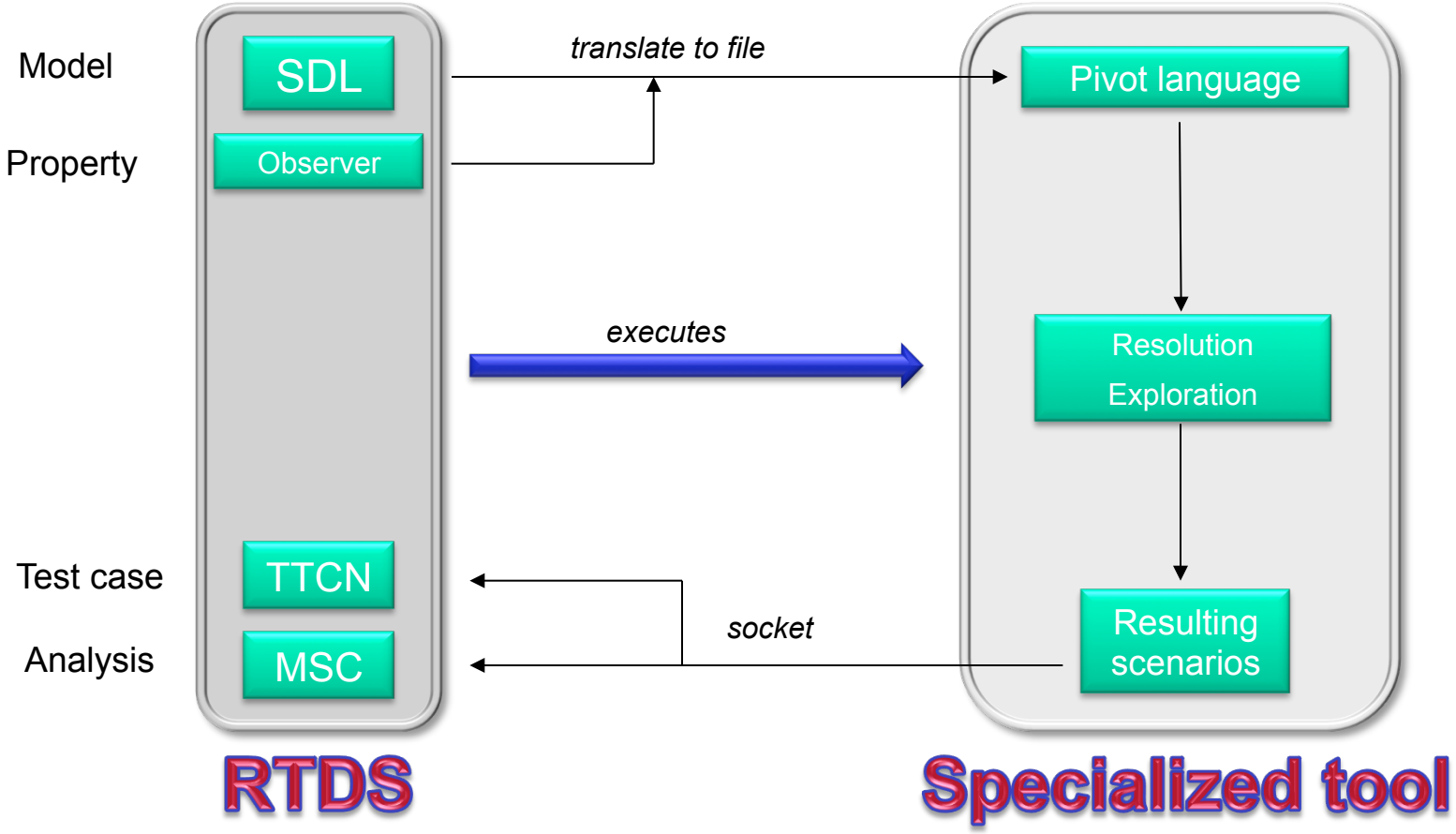


Model checking

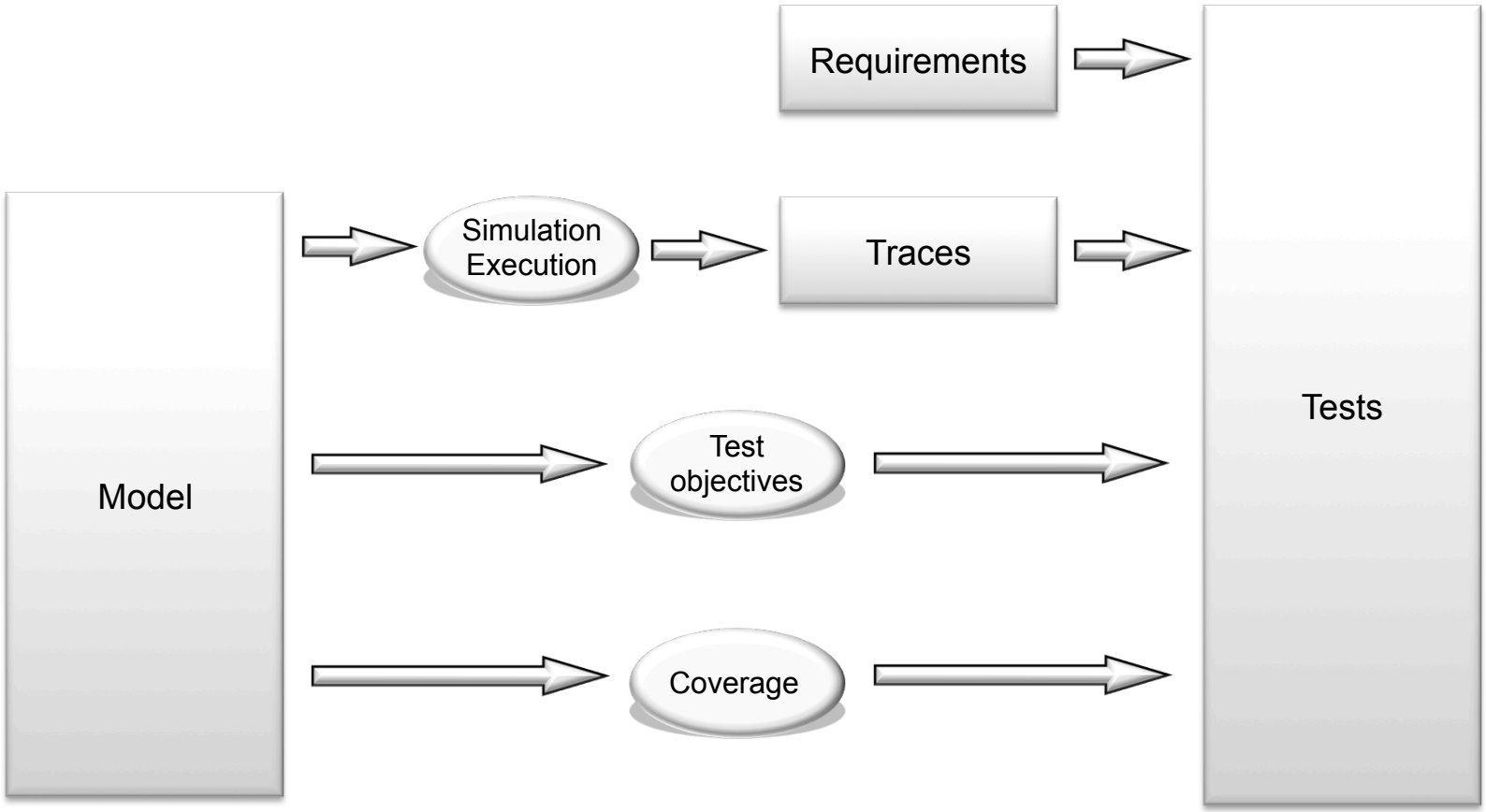
- Partnership with specialized labs:
  - Exhaustive simulation,
  - Symbolic resolution.
- Properties:
  - Model coverage,
  - Static or dynamic property:
    - Property verification,
    - Test objectives.
- RTDS feature:
  - Export,
  - Execute a script,
  - Get the results back.



Implementation

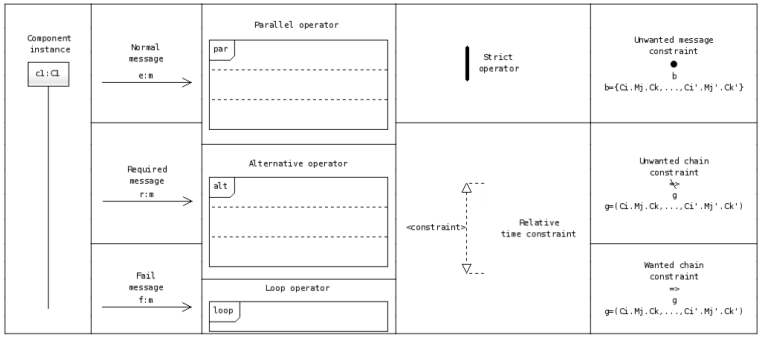


Reference testing

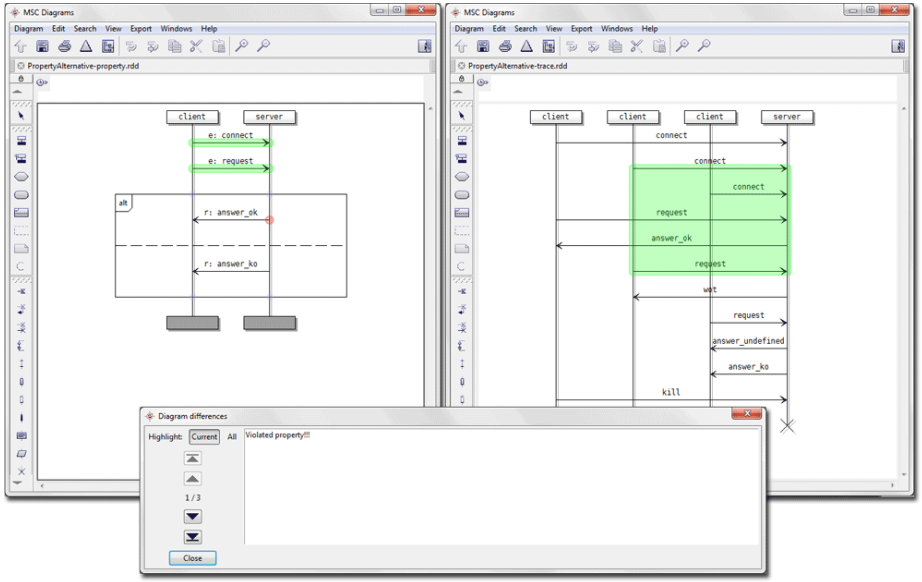


PRAGMADEV
real time development tools

Property Sequence Chart



- PRESTO european project:
  - Functional property verification.
  - Non functional property verification.
- Free tool: PragmaDev Tracer



ARTEMIS

Conclusion / Future

- SDL FSM Editor:
  - Editor only.
  - More user-friendly then RTDS, but limited.
  - Evolutions planned, but will remain limited.
- RTDS:
  - Much more features: debug, documentation, test, validation, …
  - V5 will integrate the SDL FSM Editor, and open the same diagrams.