PRAGMAList

An integrated tool for **modeling and optimized test generation** driven by ✓ **coverage** and ✓ **properties**

Automatic test generation based on functional coverage

Emmanuel Gaudin

PragmaDev

PRAGMADEV real time development tools

cea tech List

PragmaDev

- French SME,
- Created in 2001 by 2 two experts in modelling tools and languages
- Since creation dedicated to the development of a modelling tool for the development of **Event driven software**.

Aero/Defence

AIRBUS, esa, DGA, CS COMMUNICATION & SYSTEMES

Automotive

RENAULT

Telecoms

Alcatel·Lucent, AEROFLEX, ERCOM, WIPRO, NewLogic, LG Electronics, THOMSON images & beyond, KETI Korea Electronics Technology Institute

Semi-conductor

ST ERICSSON, TOSHIBA, MITSUMI

Several Collaborative Projects with big accounts

Alcatel·Lucent → ExotLicus

Focus on Model Checking

Started in 2005
finished in 2009

THALES → 

Focus on property verification

Started in 2012
finished in 2014

cea list → PRAGMAList

Focus on Model Based Testing

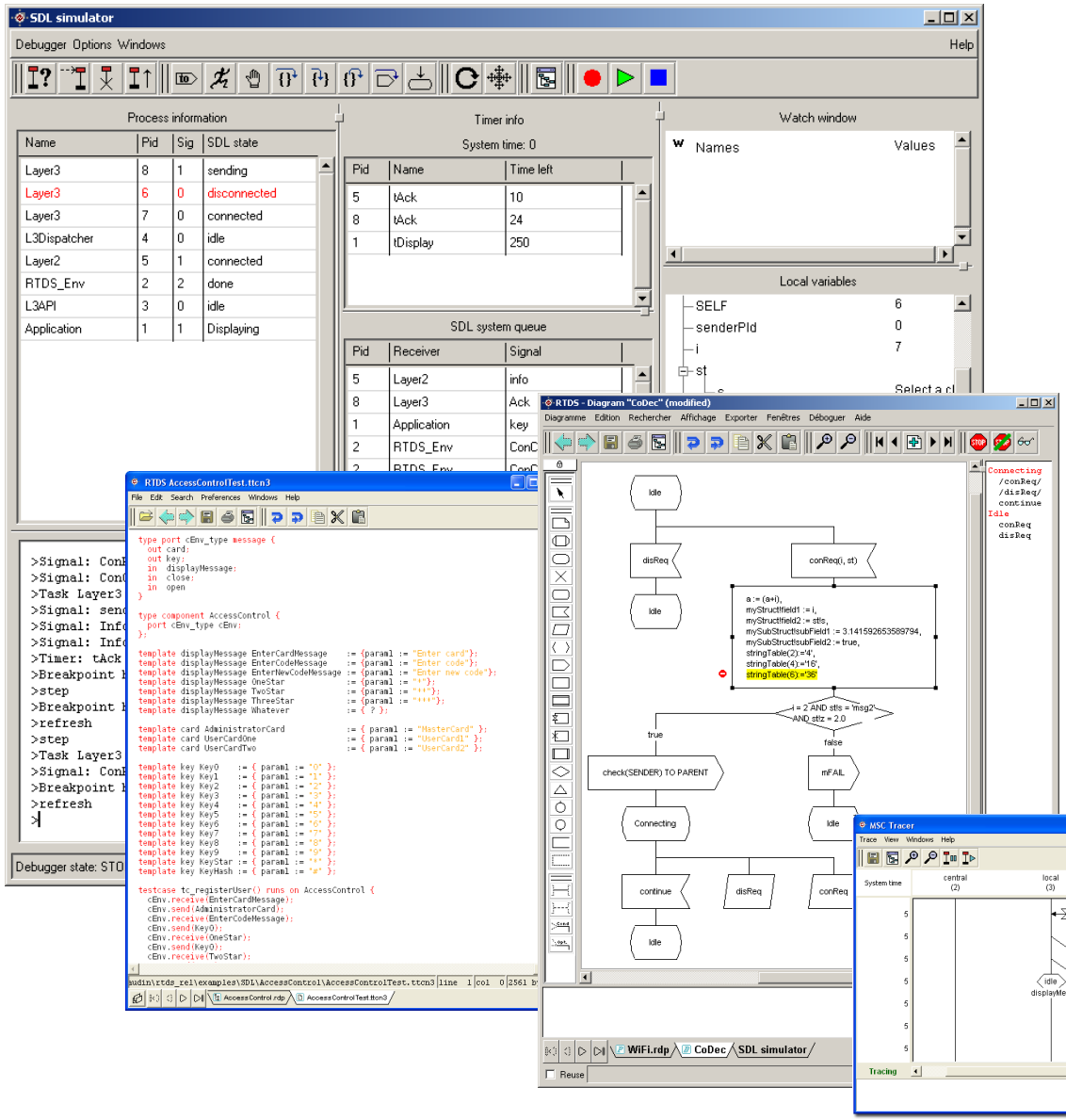
Started in 2013

Requirements for a good modelling language

- The abstract model must be platform independent, as its name states.
- The abstract model must be translatable to an execution platform.

- For that purpose, the abstract model is based on a virtual machine offering:
  - Some **basic services.**
  - An execution **semantic**.

Key features for Model Based Testing capabilities



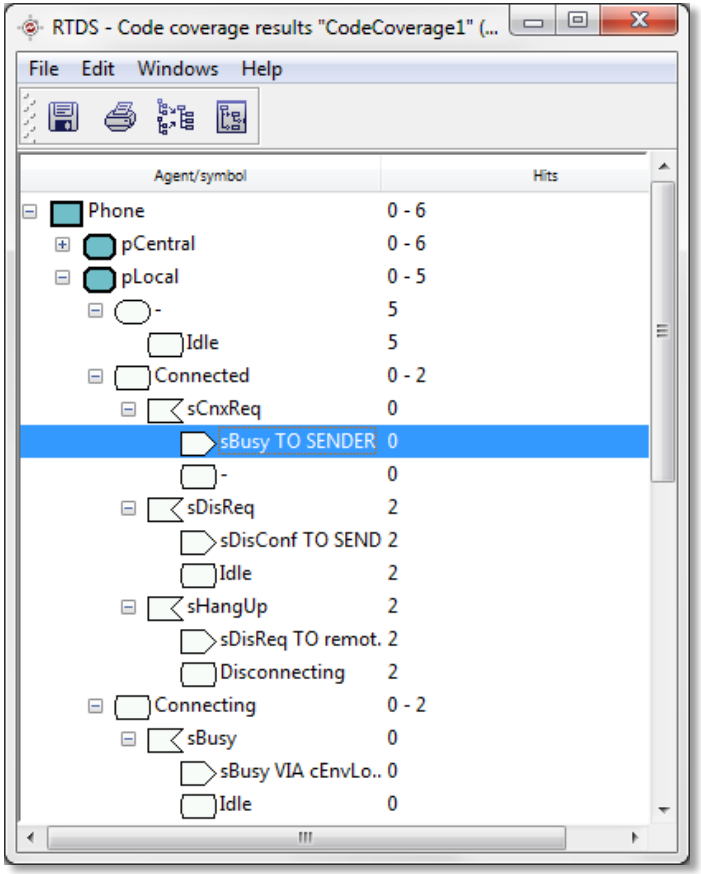
Verify the model

Model Simulator is similar to a graphical *debugger* for fully formal models:

- Breakpoints
- Step
- View internals
- Graphical traces

Model coverage

- Graphical model coverage analysis
- Merge feature



The model is :
- Correct
- Fully covered

Requirements for a good testing language

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

TTCN-3
Testing & Test Control Notation

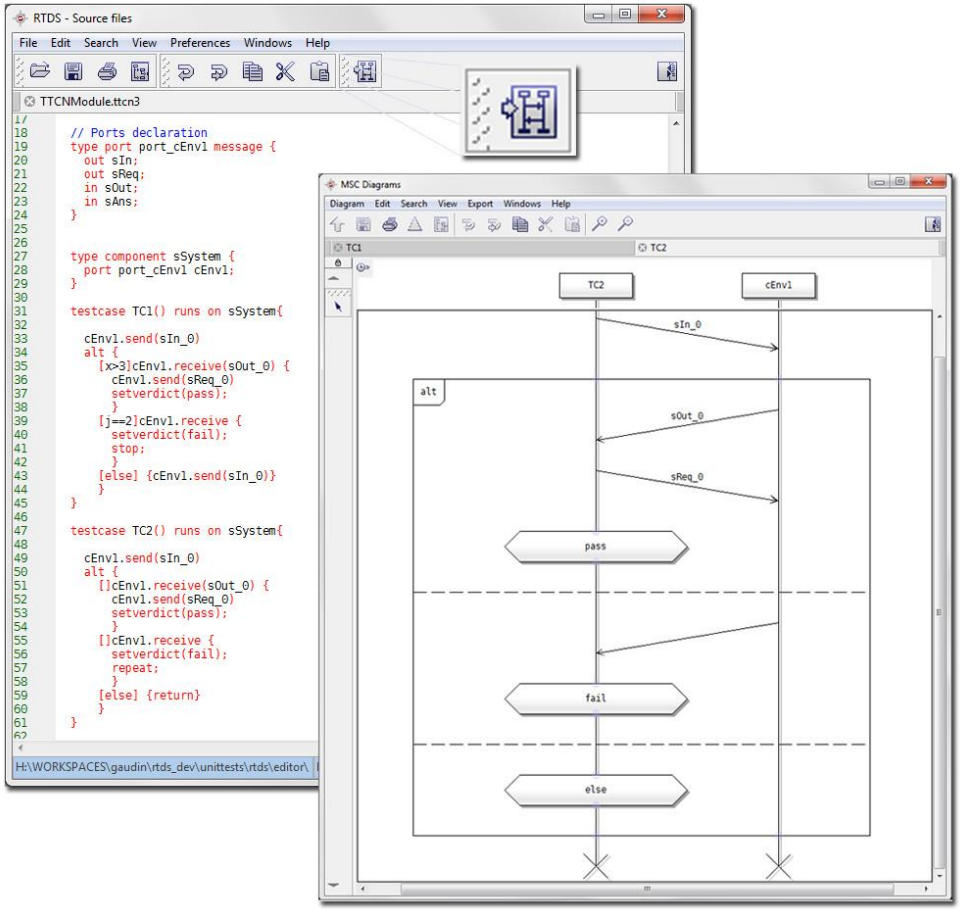
ETSI
World Class Standards

ITU International Telecommunication Union

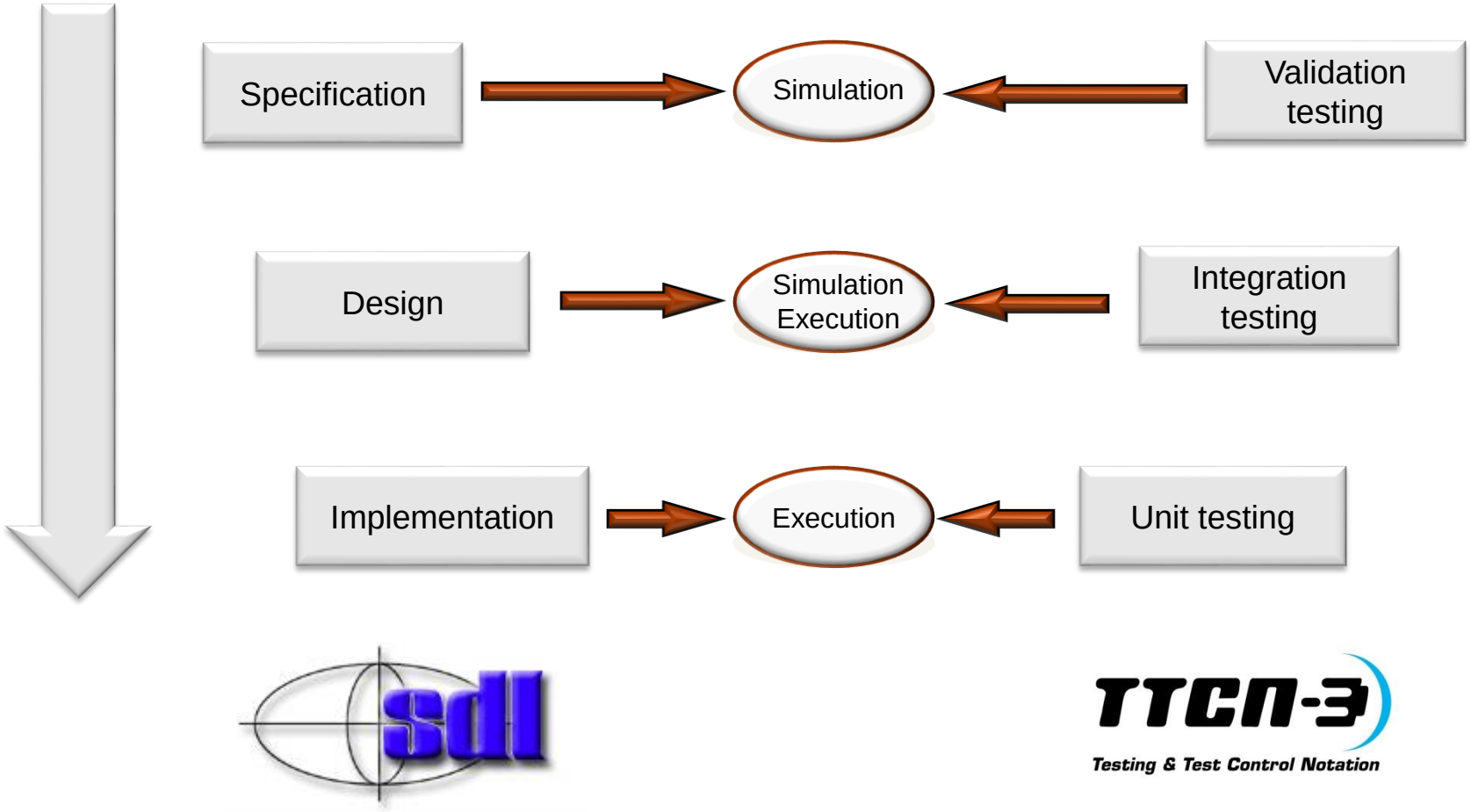
TTCN-3 support

- Textual editor
- TTCN-3 to MSC generation
- Simulator including a Test manager
- C++ code generator

- MSC to TTCN-3 generation
- TTCN-3 generation from a property on the model (Verimag)
- TTCN-3 generation based on model coverage (CEA List)



Same level of abstraction



Model analysis technologies

- Partnership with specialized labs:
  - Exhaustive simulation,
  - **Symbolic resolution**.
- Properties:
  - **Model coverage**,
  - Static or dynamic property:
    - Property verification,
    - Test objectives.

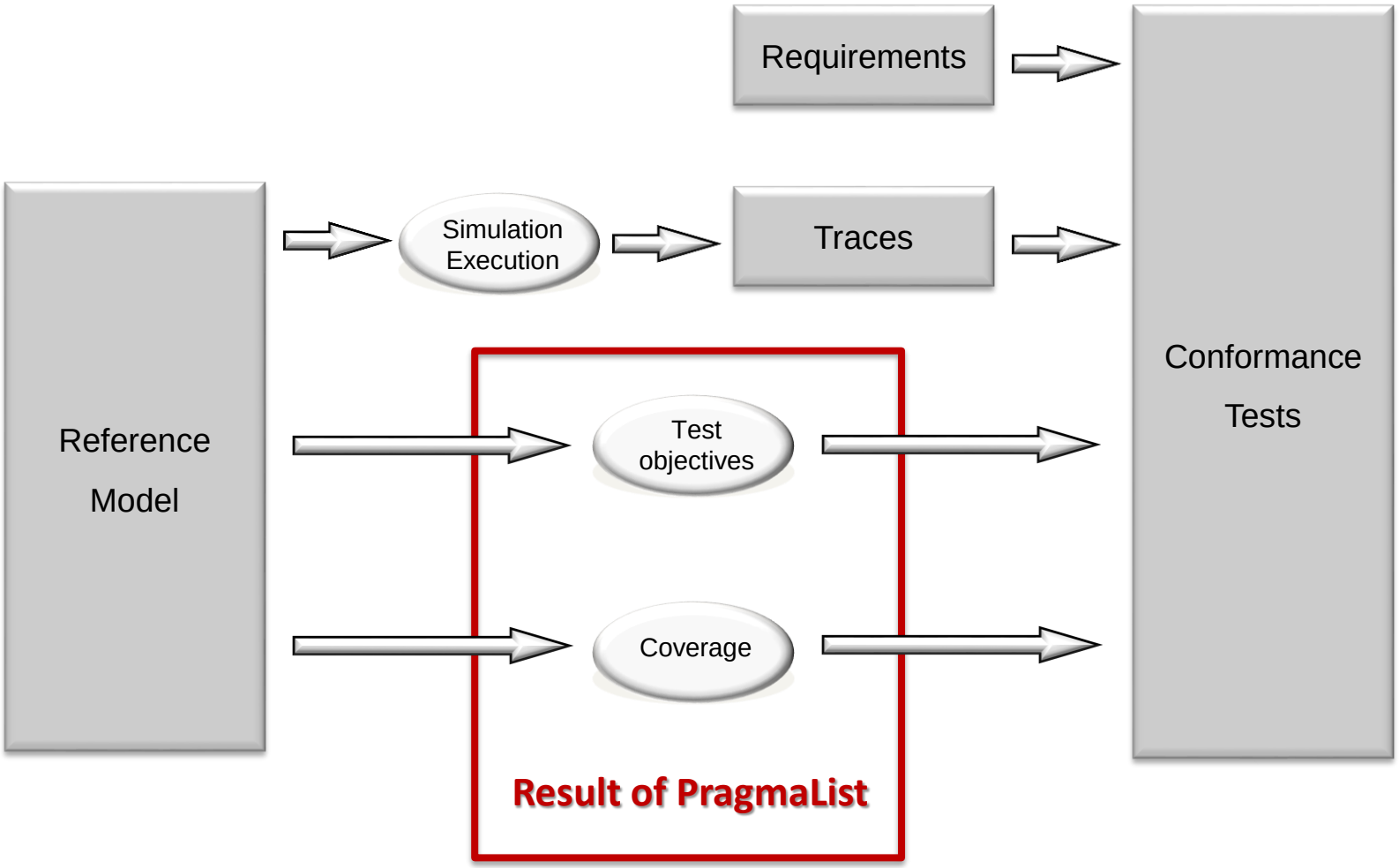
PRAGMAList

cea list

LAAS

Verimag

Reference testing

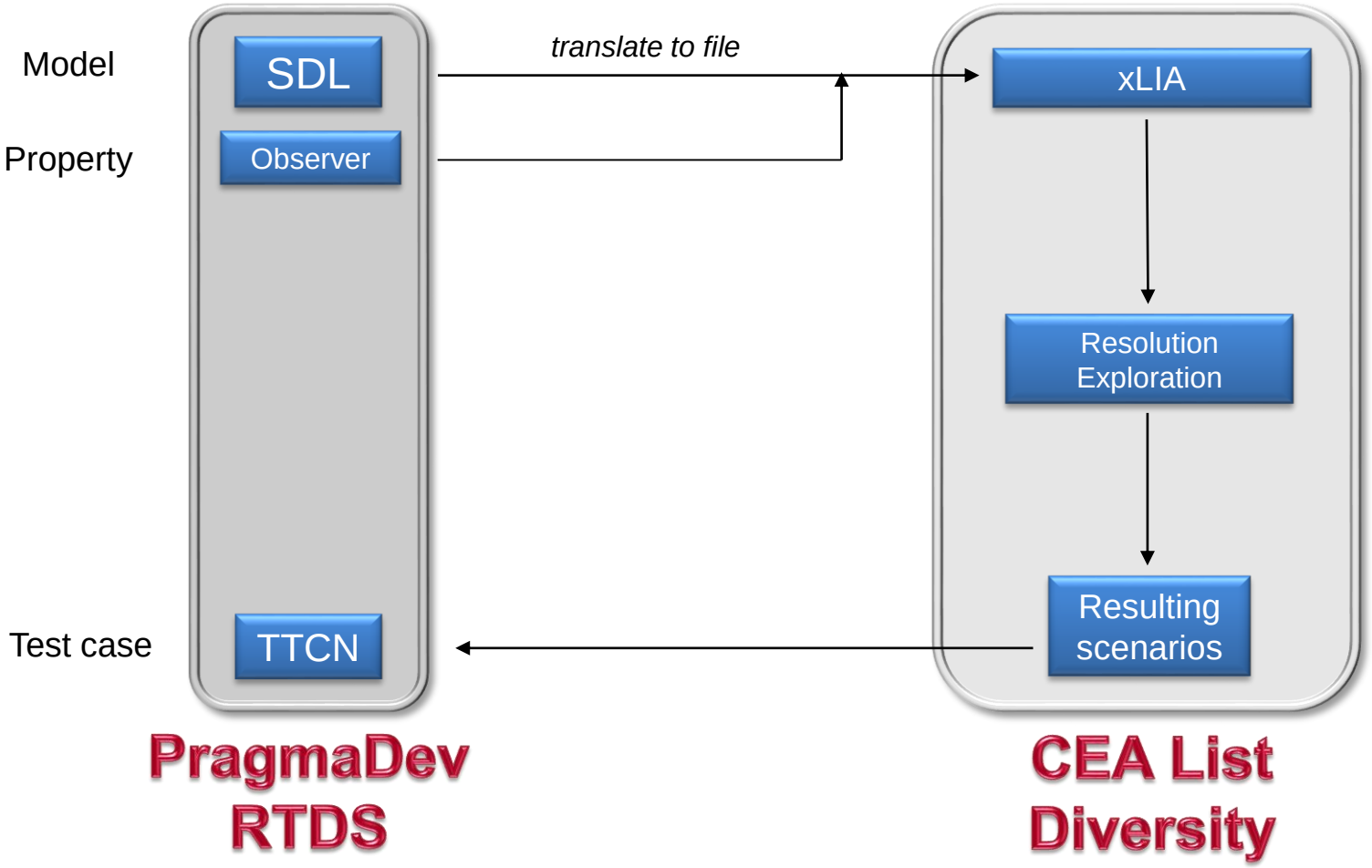


The project in four steps.

- **Step 1 : SDL to xLIA translation rules :**
  - Write the translation rules to convert SDL to xLIA.
- **Step 2 : SDL to xLIA translator :**
  - Write the xLIA generator from an SDL model.
- **Step 3 : Diversity adaptation to support SDL semantic :**
  - Work on SDL communication semantic,
  - Work on SDL timer semantic.
- **Step 4 : TTCN-3 formats output generation :**
  - TTCN-3 test cases formatting to be supported by RTDS.

xLIA is the CEA List Diversity file format to describe the model

Architecture



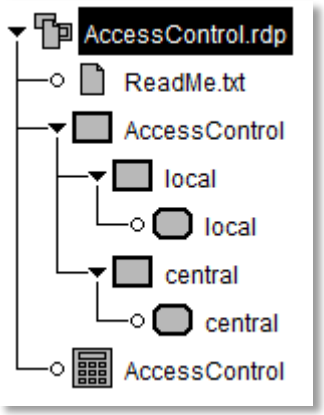
Four types of targets

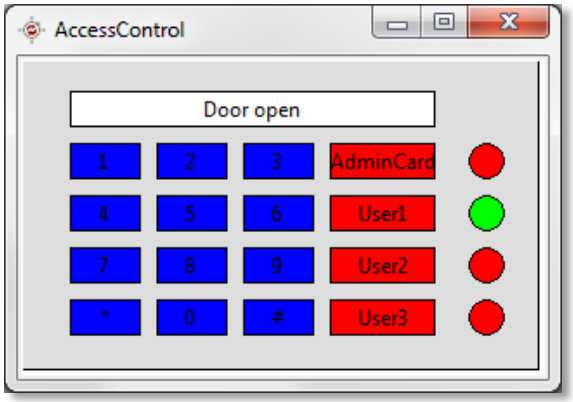
- **Code coverage :**
  - To generate the minimum number of test cases that cover all transitions.
- **Transition :**
  - To generate a test case that covers a specific transition in the SDL model.
- **Property :**
  - To generate the test cases verifying a static property (process state, variable value, …).
- **Observer :**
  - To generate the test cases verifying a dynamic property (succession of action or temporal rules). A dynamic property is defined as a state machine called observer.

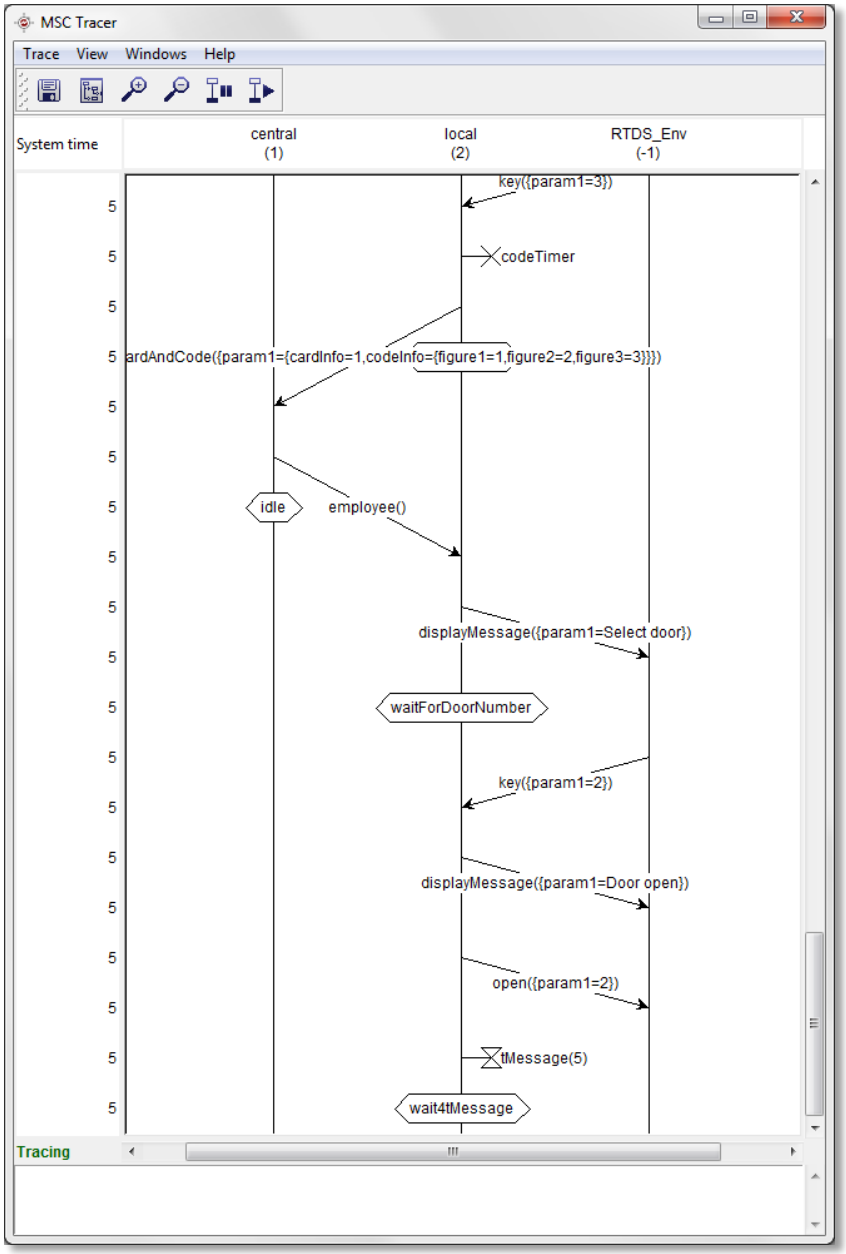
Demonstration

An Access Control System:

- 2 state machines
- A card input with a 0..65535 integer as a parameter
- A key input with a 0..11 integer as a parameter

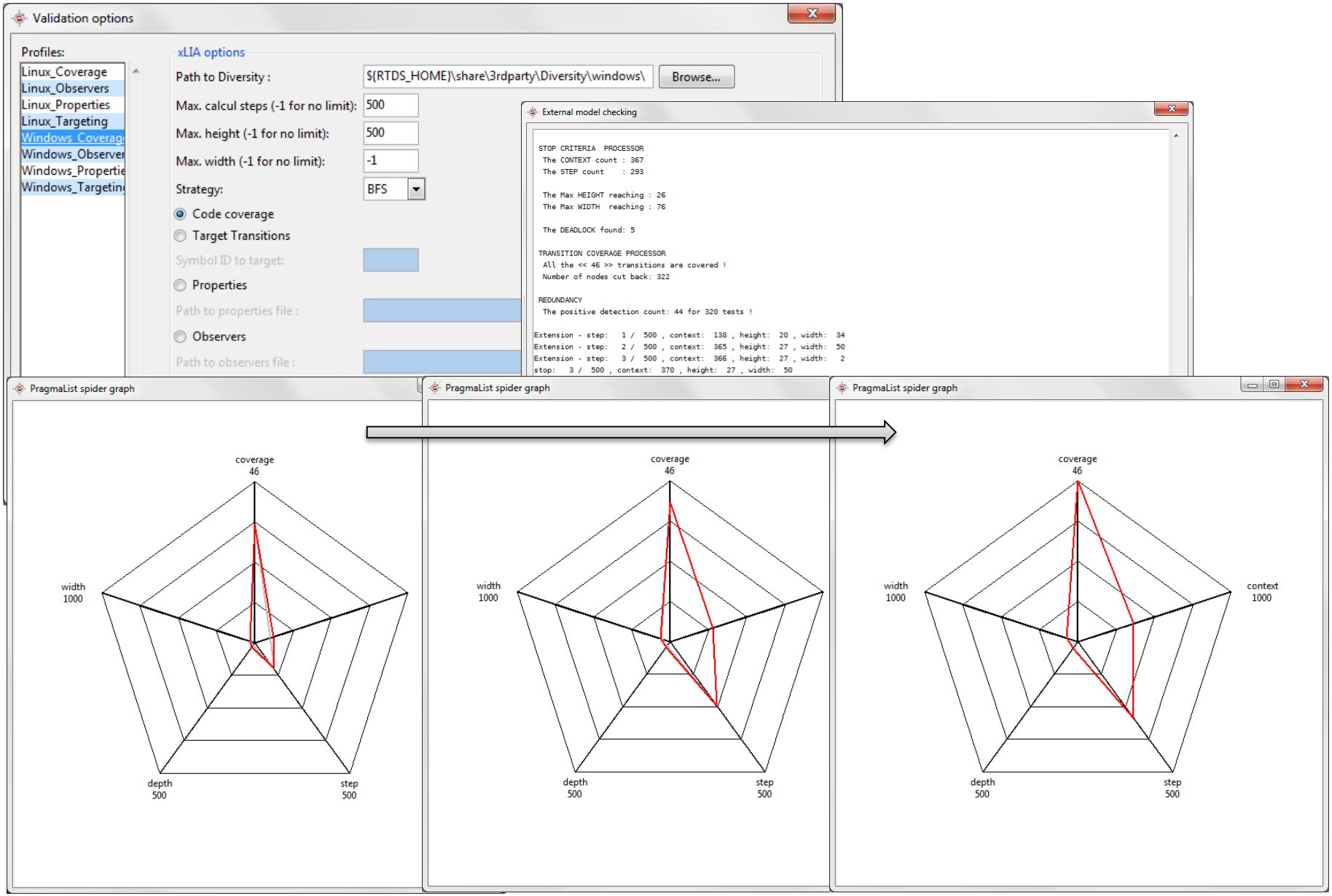




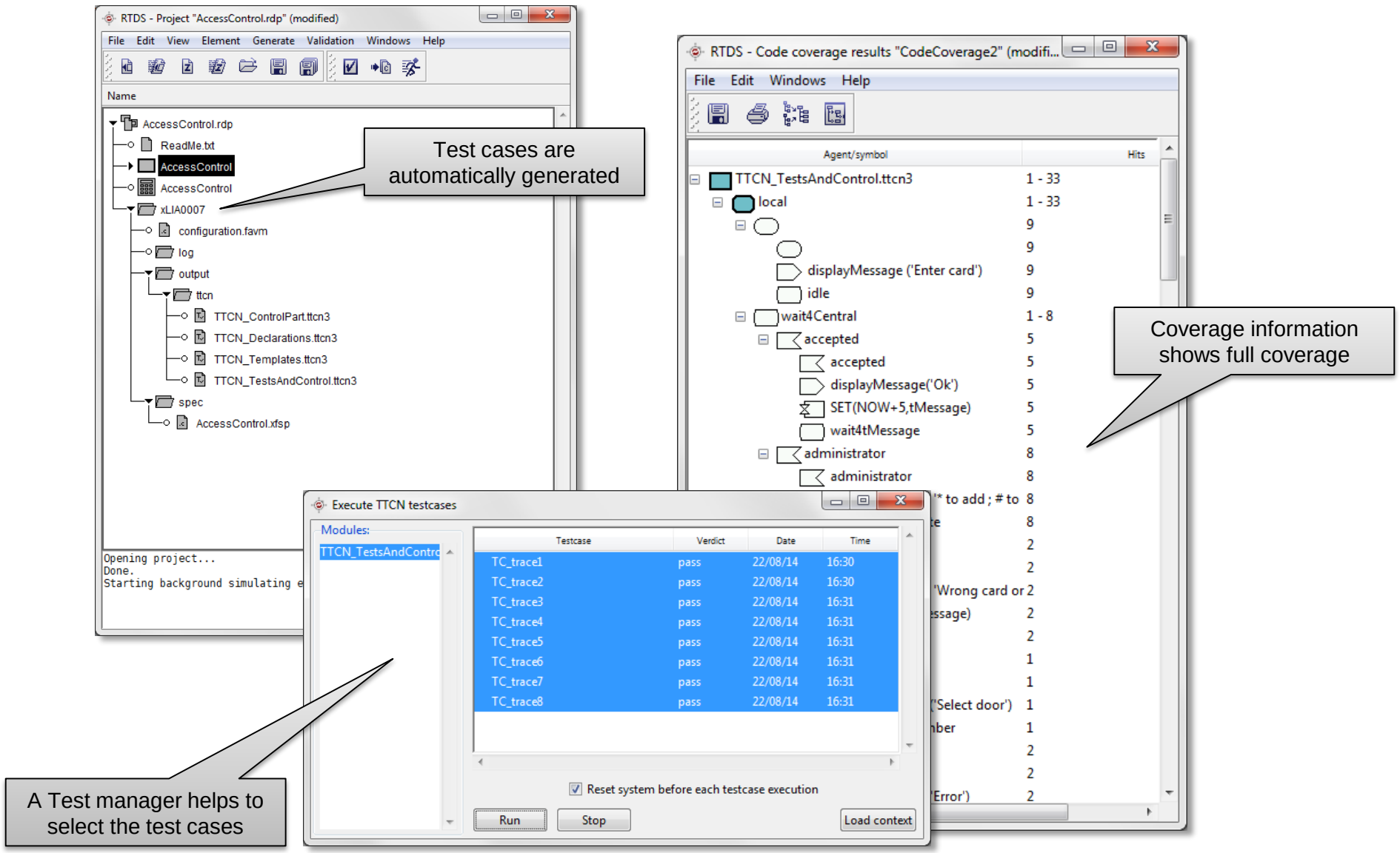


PRAGMAList

Example



Example


RTDS - Project "AccessControl.rdp" (modified)
File Edit View Element Generate Validation Windows Help
Name
AccessControl.rdp
ReadMe.txt
AccessControl
AccessControl
xLIA0007
configuration.favm
log
output
ttcn
TTCN_ControlPart.ttcn3
TTCN_Declarations.ttcn3
TTCN_Templates.ttcn3
TTCN_TestsAndControl.ttcn3
spec
AccessControl.xfsp
Opening project...
Done.
Starting background simulating e
Test cases are automatically generated
RTDS - Code coverage results "CodeCoverage2" (modifi...
File Edit Windows Help
Agent/symbol
Hits
TTCN_TestsAndControl.ttcn3 1 - 33
local 1 - 33
9
9
displayMessage ('Enter card') 9
idle 9
wait4Central 1 - 8
accepted 5
accepted 5
displayMessage('Ok') 5
SET(NOW+5,tMessage) 5
wait4tMessage 5
administrator 8
administrator 8
Coverage information shows full coverage
Execute TTCN testcases
Modules:
TTCN_TestsAndContro
Testcase Verdict Date Time
TC_trace1 pass 22/08/14 16:30
TC_trace2 pass 22/08/14 16:30
TC_trace3 pass 22/08/14 16:31
TC_trace4 pass 22/08/14 16:31
TC_trace5 pass 22/08/14 16:31
TC_trace6 pass 22/08/14 16:31
TC_trace7 pass 22/08/14 16:31
TC_trace8 pass 22/08/14 16:31
Reset system before each testcase execution
Run
Stop
Load context
A Test manager helps to select the test cases

CEA List - Diversity

- Exploration time is always the same (10 secondes) whatever are the message parameter ranges.

Verimag - IF toolbox

- Exhaustive exploration
- Exploration time depends on message parameter range.

Card range \ Digit range	0..1	0..2	0..3
0..1	13	126	721
0..2	38	316	2169
0..3	64	650	28234

Time to explore the model in seconds

On-going use cases

- SNCF: Radio Block Center (RBC)
- Alstom Belgium: Radio Gateway
- Alstom France: Passenger exchange
- Airbus: Air Traffic Control (ATC)
- Other: Secure transactions

Model Based Testing solution

- Integrated tool chain
- Non dedicated model
- Efficient symbolic kernel
  - Test automation
  - Reduce the number of test cases
  - Early in the development process