For a reliable and scientific approach in system and software engineering

TASTE: The ASSERT Set of Tools for Engineering

The TASTE Toolset

assert-project - Dr Eric Conquet, Maxime Perrotin, Marie-Aude Esteve – European Space Agency
Statement of the problem

- Targeted domain: Real-Time, mission critical and embedded systems
- SW dominant systems (but expandable to HW)
- Current development approach is mostly based on paper work
  - Few models,
  - Tools: mainly MS office!
- Suppliers face difficulties to master design before testing
- Customers find reviews not efficient enough
- Needs:
  - Capture system model with associated properties,
  - Verify early and continuously during design,
  - Smoothly handle system heterogeneity,
  - Use automatic code generation.

Those were addressed in ASSERT 2004/2007
The ASSERT Requirement baseline

- System families: from market segments to property oriented design
- Proofs: from an empirical to a scientific approach,
- New development process: from nice concepts to actual steps
- Tools: From paper to models
- Heterogeneity handling: from multiple models to one single and integrated SW
- Case studies: from toy examples to real applications.

**ASSERT = an ambitious and pragmatic approach to develop critical SW dominant RT systems.**
Future of ASSERT: a strategy around three main axis.

- **Extend the process and toolset: ongoing studies**
  - Link with system modelling
  - Integration of HW components
  - Integration of space specific components (PUS, Spacewire, ...)
  - Connection to system simulators and schedulability analyser

- **Market the toolset**
  - Create an ecosystem gathering innovative SMEs and research centres into a network
  - Each network node will lead a technical domain
  - The network can be dynamically configured to address a specific need (virtual company)
  - To have ESA coordinating the network and protecting the community investment

- **Disseminate the technology**
  - Case study on formation flying experiment with space industry
  - Preparation of case studies with space and non space industry
From the ASSERT process to TASTE.

The TASTE Toolset

assert-project - Dr Eric Conquet, Maxime Perrotin, Marie-Aude Esteve – European Space Agency
What is the Assert process?

- **The ASSERT process is based on simple observations**
  - a system – ANY system – is made of *heterogeneous components*, that have to live and communicate together
  - system builders have other concerns than *software implementation details*,
  - and good software engineers are unhappy when they have to develop application code: their skills are misused

- **The Assert process proposes solutions to**
  - capture a system using user-friendly (yet formal) modeling techniques
  - automate repetitive and error-prone software activities
  - build an homogeneous system having heterogeneous components

- **The toolset has been specified, designed, and implemented by ESA together with some Assert partners.**

- **It is unique on the market.**
Capture of the system: architecture, behaviour, data, real-time attributes, and hardware platform.

AADL and ASN.1 are combined to provide a formal, precise, and complete description of the system architecture and data.
What ASSERT tools do with the models

1. Generate “application skeletons” in Simulink, SDL, C, and Ada
2. Generate a software real-time architecture (in AADL)
3. Generate glue code to put everything together on a real-time operating system
ASSERT technology details

- **The ASN.1 Space Certifiable Compiler**
  - Generation of compact binary encoders and decoders for ASN.1 modules
  - Integrated in the ASSERT tool-chain, can be used standalone as well
  - Targets embedded systems: no dynamic memory allocation, no system calls

- **The “virtual machine” (assert VM) – two options**
  - A real-time operating system based on the Ada runtime with the Ravenscar profile and PolyORB-HI middleware.
  - RTEMS standard operating system with PolyORB-HI-C
  - Supported platforms: Leon, Native (Linux on Intel)
  - Supported networks: Ethernet (for native platforms), Spacewire, serial link
In addition - bonuses

- **Rapid prototyping:** the toolchain generates GUIs to quickly test the system under development

- **Simulation and Analysis:** Data can be monitored using real-time plotting.

- **Documentation:** ICDs are generated automatically with a description of the data binary encodings (ASN.1 uPER Encodings)
A few screenshots of the toolset (1)

- Interface and deployment view editors
A few screenshots of the toolset (2)

- **Generation of Simulink “skeletons”**
A few screenshots of the toolset (3)

- Generation of SDL “skeletons”

```
system basic_fv

USE Datamodel;

SIGNAL basictotc (T_TM);
SIGNAL tcommand (T_HLTC_PLUS);
SIGNAL basictocontrol (T_CONTROL_IN);
SIGNAL controldowntobasic (T_CONTROL_DOWN_OUT);
SIGNAL controluptobasic (T_CONTROL_UP_OUT);
SIGNAL cyclicactivationimplementation;

procedure aplc_basic_op COMMENT '#c_predef':FPAR
IN thrusters_opening T_THRUSTERS_OPENING,
IN pfs_hw_m_aruming_relay_status on T_PFS_HW_M_ARMING_RELAY_STATUS_ON,
IN pfs_hw_m_hltc_red_button_is_on T_PFS_HW_M_HLTTC_RED_BUTTON_IS_ON,
IN msu_id T_MSU_ID,
IN pfs_ew_m_msuy Msux hs T_PFS_EWM_MSU_MSU_Hs,
IN ftcp_health_status T_FTCP_HEALTH_STATUS,
IN pfs_ew_m_dtg12_msu T_PFS_EWM_DTG12_MSU,
IN hltc T_HLTC,
IN end_boost_is_reached T_END_BOOST_IS_REACHED,
IN sun_is aimed T_SUN_IS_AIMED,
INOUT pfs_ewc_msu_pde T_PFS_EWC_MSU_PDE,T,
INOUT pde_cmd_a T_PDE_CMD_A,
INOUT dpu_cmd T_DPU_CMD,
INOUT set_pfs_ewc_msu_dtg_mode_coarse T_ON_OFF_CMD,
INOUT hltm T_HLTM,
INOUT pfs_ew_m_msuy Msuy hs T_PFS_EWM_MSU_MSU_Hs,
INOUT cam_mode T_CAM_MODE,
INOUT controller_to_be_activated T_CONTROLLER_TO_BE_ACTIVATED,
INOUT navigation_output T_NAVIGATION_OUTPUT;
EXTERNAL:

procedure mysimulink COMMENT '#c_predef':FPAR
IN my_in T_FOR_SIMULINK_IN,
IN my_in2 T_control_in,
INOUT my_out T_FOR_SIMULINK_OUT,
INOUT my_out2 T_Control_in;
EXTERNAL;
```
A few screenshots of the toolset (4)

- C "skeletons"
A few screenshots of the toolset (5)

- **Ada “skeletons”**
A few screenshots of the toolset (5) - Data types in Ada

```ada
package ASN1_Types is
  type Destination_T is (displayer, other_dest);
  for Destination_T use
    (displayer => 0, other_dest => 1);
  for Destination_T'Size use Integer'Size;

  type Display_T_inner is
    array (0..254) of Character;

  type Display_T is
    record
      length : Integer;
      data : Display_T_inner;
    end record;

  type Action_T is
    record
      choice : selection;
    end record;

  type Action_T'choice is
    (action => 0, display => 1);

  type TC_T is
    record
      destination : Destination_T;
      action : Action_T;
    end record;

  type TM_T is new Display_T;
```

The TASTE Toolset

assert-project - Dr Eric Conquet, Maxime Perrotin, Marie-Aude Esteve – European Space Agency
The after TASTE story

- **Extensions to TASTE:**
  - HW/SW co engineering: study in progress with GMV (Spain) to extend TASTE with HW (FPGA) components handling.
  - System modelling: study in progress with ASTRIUM (France) to extend TASTE with system modelling (SysML) and Spark Ada generation.

- **Demonstrators:**
  - FDIR (Failure Detection Isolation and Recovery) modelling: study almost finished with SpaceBel (Belgium) to model FDIR layers for formation flying.
  - Galileo/GPS receiver on a satellite platform: study with M3S (France) with THALES and ASTRIUM (includes a prototype to support HWSW co-design developed by SEMANTIX (Greece)).
The TASTE Toolset

assert-project

Automated proof-based System and Software Engineering for Real-Time systems

assert-project

For a reliable and scientific approach in system and software engineering

http://www.assert-project.net