



PBSE and AADL in ASSERT

Jean-François Tilman
Axlog ingénierie

`Jean-Francois.Tilman@axlog.fr`



Agenda

- Introduction
- The ASSERT project
- PBSE and TRDF
- AADL for this purpose
- Prospects



ASSERT



ASSERT objectives

- ASSERT = "Automated proof-based System and Software Engineering for Real-Time applications"
- "Software crisis": current empirical practices and increasing complexity.
- Purpose: improvement of the development process and quality of operational systems by:
 - following a PBSE method during the whole development cycle;
 - using an ADL to write specifications exchanged across lifecycle phases;
 - ensure the continuity with the SW development.



ASSERT clusters and leaders

PBSE
(Inria, Axlog)

Dependability
Distribution
Hard Real Time
(LAAS, ENST)

Development
Verification Tools
(CS, Verimag,
TNI, ETH)

Multi-domain Advanced Available Automated System pilot project
(EADS-ST, Astrium, Dassault Aviation)

Highly Reliable Infrastructure pilot project
(Alcatel, Alenia)

Open pilot project

Process & Standardisation (Synspace)

Openness & Exploitation (University of Padua)



PBSE and TRDF



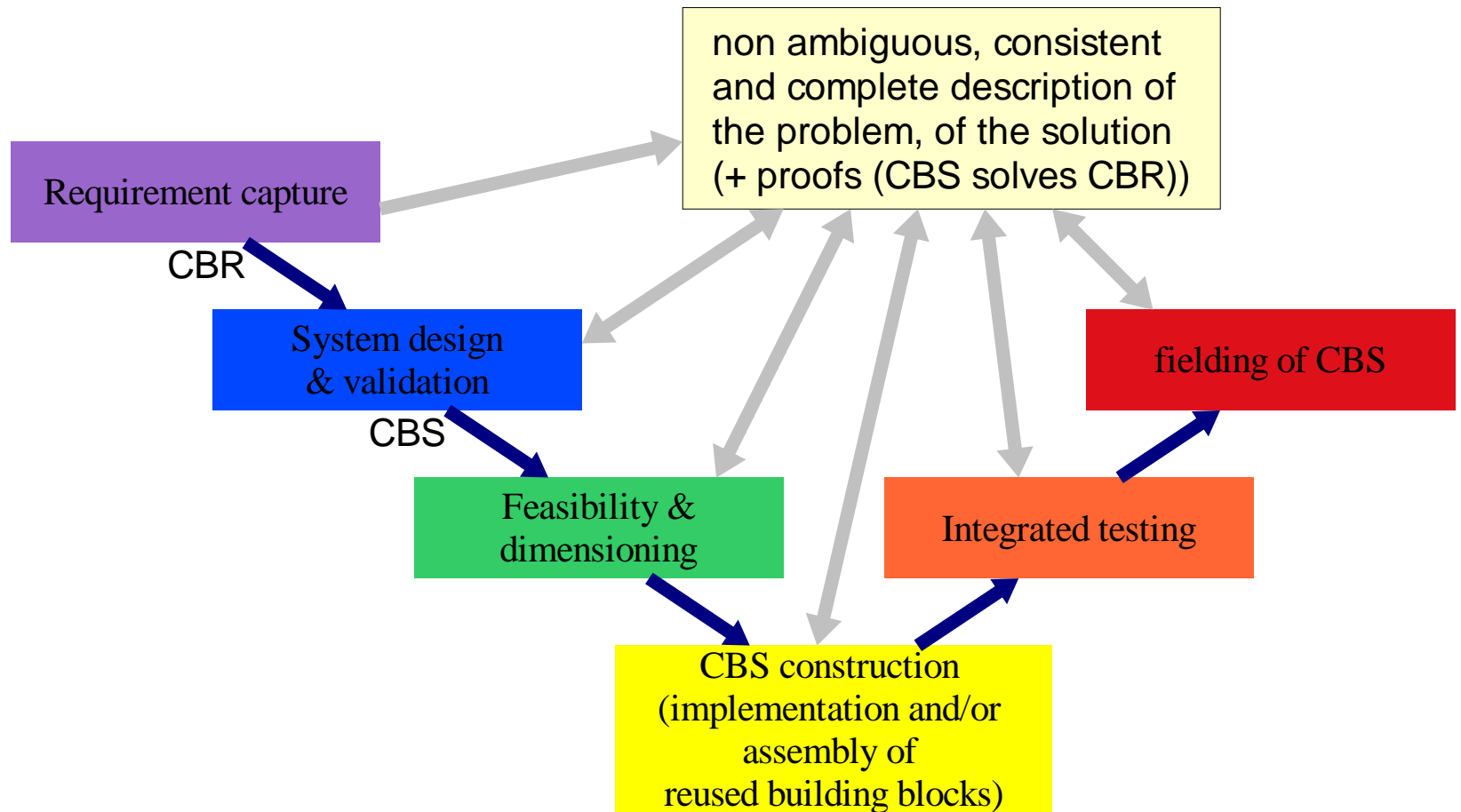
The PBSE approach

- PBSE = "Proof-Based System Engineering"
- Aim: a system engineering method for computer-based systems aimed at eliminating faults in the early phases of lifecycles.
 - => rigorous specifications & fulfillment of proof obligations.

Phases	Contents/outputs
Requirement capture (RC)	description of the application/user problem specification of the computer-based system problem (CBR)
System design & validation (SDV)	specification of a computer-based system solution (CBS)
Feasibility & dimensioning (FD)	building block composability checking and tailoring
Integrated testing (IT)	generation of the complete suite of tests needed to check the implemented CBS with respect to worst-case (load & failure) scenarios



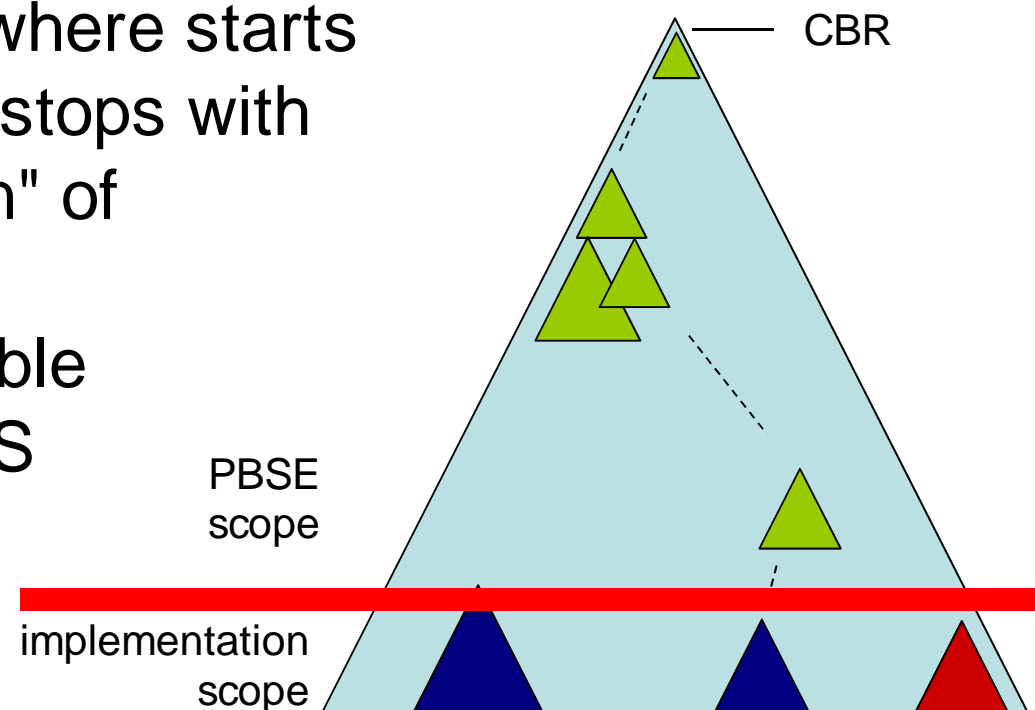
The PBSE life-cycle





Building an architecture

- A building-block is any concept which can be used from the upper specification down to the final implementation
- PBSE concern stops where starts implementation, but it stops with a "proven specification" of what must provide the terminal BB to be eligible to participate to a CBS





TRDF

- TRDF = *Temps-Réel, Traitement Distribué, Tolérance aux Fautes*
(Real-Time, Distributed Treatments, Fault Tolerance)
- TRDF is a PBSE method based on deterministic approaches/solutions.
- Developed at INRIA, and is today the only existing PBSE method, tried successfully in various domains.
- See [GLL1998] for more information.



Success stories of PBSE/TRDF

- P1 (1995-97): Modular avionics (air combat) for Dassault Aviation and French MoD => validation of the full TRDF method (from RC to CBS implementation).
- P2 (1996-97): For IPSN (French Atomic Energy Authority): how to tell whether COTS can be used in safety critical systems?
- P3 (1996-98): For French Ministry of Research and DA => PBSE solves problems that cannot be solved with (formal) SW engineering methods - which may explain the "SW crisis".
- P4 (1997): Analysis of the Ariane 5 flight 501 failure => Satellite launcher explosion has been caused by 1 fault in the RC phase – SW is not the "culprit" (contrary conclusions in the official IB report).
- P5 (2001-03): Space-borne systems, for ESA and EADS Astrium => proven building-blocks for distributed dependable middleware (consensus/coordination in the presence of failures).



AADL for this purpose



AADL in ASSERT

- ASSERT needs an ADL to handle non-ambiguous, consistent and complete descriptions of applications/user problems.
- The most mature ADL in this domain is AADL;
=> Axlog has promoted AADL, which has been chosen as a base for the project.
- AADL has not been initially designed for this specific purpose.
=> the studies will reveal complements and extensions to support the needs of ASSERT.



AADL extensions

- Several kinds of extensions may be proposed for AADL:
 - PBSE extensions: to support the proof obligations at any level of the description;
 - DDHRT extension: the needs of the dependability and distribution cluster are probably closer with the current AADL (e.g., connectors).
- => collaboration with the AADL standardisation committee.



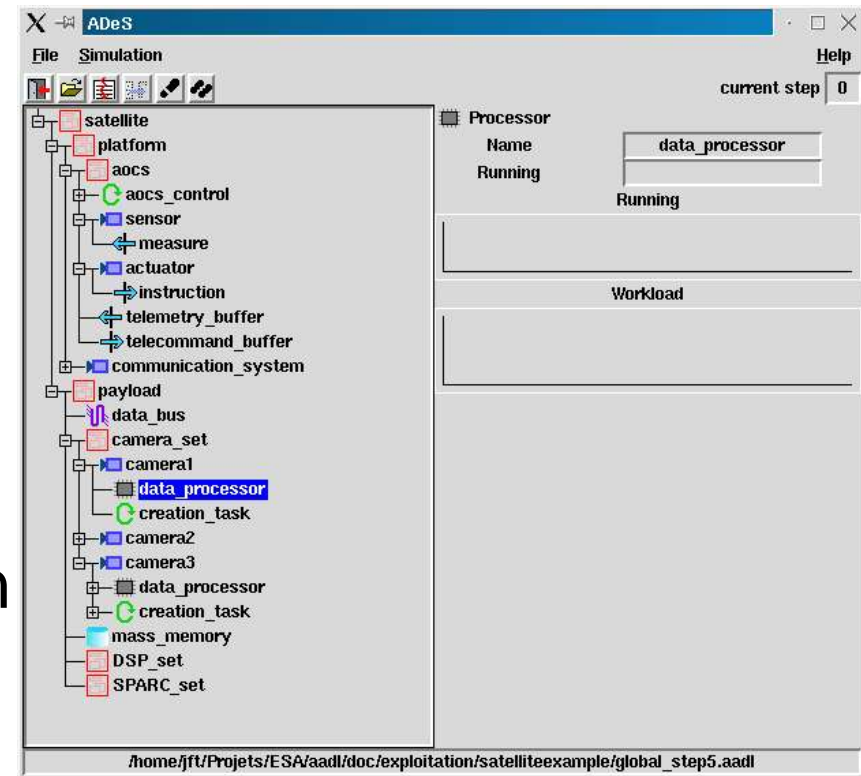
PBSE tools

- In ASSERT, several prototype tools will be developed to support the process. The aim is to have an expertise of AADL to demonstrate the feasibility of these tools:
 - RC tool: to support the requirement capture phase (questionnaires, dictionaries);
 - SDV tools: to support system design and validation (system design "assistants");
 - FD tool: Oracles to support feasibility and system dimensioning;
 - IT tool: Oracles to support (final) integrated testing.



ADeS

- ADeS is an example of a possible SDV tool.
- It has been developed during the AADL study led by Axlog for ESA.
- Its purpose is the simulation of the behaviour of an architecture described with AADL.
- The results of a simulation are not proofs, but they may help the designer in its work.





Prospects

- Two major goals of ASSERT are to check:
 - is it possible to build PBSE tools?
 - can such tools be used by the "average" engineer, in the course of a real project?
- Timescale: by mid-2006, prototypes of PBSE tools will be available and assessed.
- Rationale: the adoption of PBSE methods appears to be the best strategy to achieve the "faster, cheaper, better" goal.



References

- [GLL1998] Gérard Le Lann, "Proof-Based System Engineering and Embedded Systems", invited paper, European School on Embedded Systems (Veldhoven, NL, Nov1996), in Lecture Notes in Computer Science n° 1494, Springer-Verlag Pub., Oct 1998, pp. 208-248. (http://www-rocq.inria.fr/reflecs/publications_fr.html)