From Rigorous Requirements Engineering to Formal System Design of Safety-Critical Systems

by Christophe Ponsard, Philippe Massonet, Gautier Dallons

The majority of systems that control our daily lives rely on software. Often this software is embedded in the device and remains invisible to users. While the consequences of a failure may be minor, such as failing to deliver part of the requested service, they may also be dramatic, possibly causing human injury or even fatalities (eg car crashes or train collisions). In recent years, CETIC (Centre d'Excellence en Technologies de l'Information et de la Communication) has devoted significant effort to the development of industrial methods and tools that target safety-critical software, with a specific focus on the early phase of system development.

At this level, the consequences of design flaws are the most dramatic and costly to correct. Large studies like Chaos (Standish Group) have also shown that these flaws remain the principal reason for project failure. To improve this, the following topics, illustrated in Figure 1 using the V reference model, are being investigated from a model-driven engineering perspective.

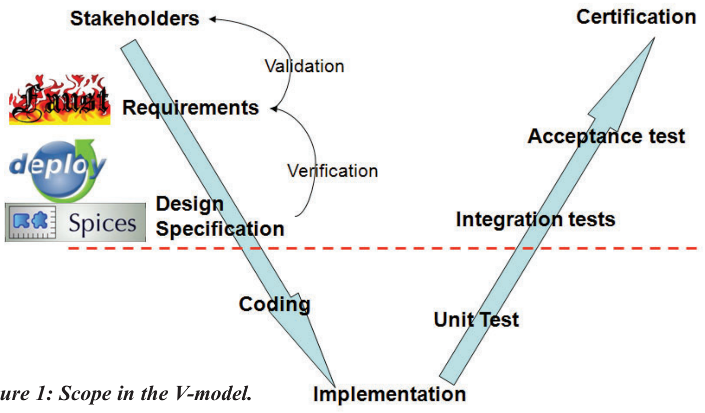


Figure 1: Scope in the V-model.

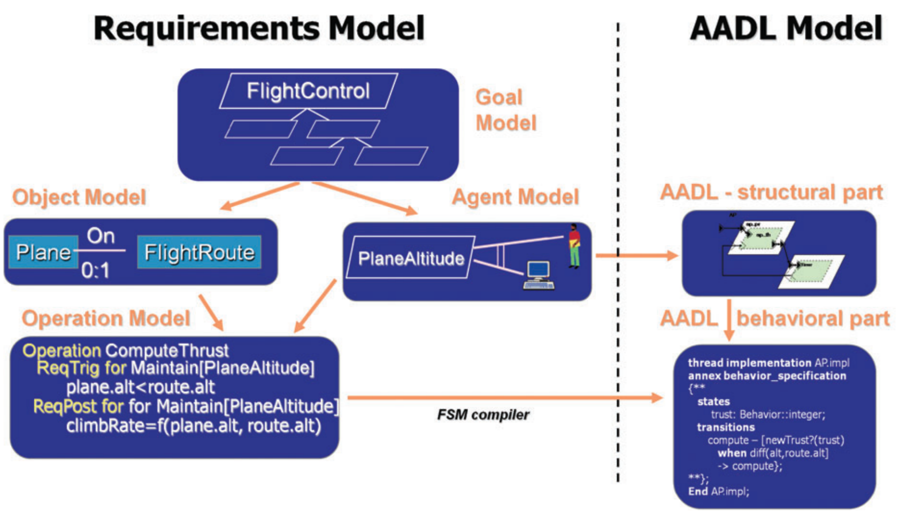
At the start of the life cycle, rigorous engineering of requirements aims to provide adequate methods and tools to capture, formalize and perform early verification and validation of critical requirements.

Immediately following this, a strong connection with industrial development methodologies is required. Good candidates are Event-B/B (a generic proof-oriented formal language) and AADL (architecture description language). At the other end of the life cycle, the certification process should support adequate techniques for a high level of assurance.

Rigorous Requirements Engineering

The aim of requirements engineering is to capture the intended behaviour of a system (including its safety properties) and the characteristics of its environment of operation. CETIC has adopted KAOS, a major goal-oriented methodology, which combines two description levels: an informal/graphical level for optimal communication and a formal layer enabling powerful reasoning about the requirements. A toolset called FAUST (Formal Analysis Using Specification Tools), which extends the semi-formal Objectiver tool, was developed to support the formal level. It provides the following main tools:

- validation of requirements, based on an animator able to generate state-based animation that can be displayed in graphical representations from the domain
- verification of the requirements, including completeness and consistency criteria, based on model-checking technology providing explanatory counter-examples
- acceptance test generation, based on the coverage of the system properties, especially those most critical.



```
Operation ComputeThrust
  ReqTrig for Maintain[PlaneAltitude]
      plane.alt<route.alt
  ReqPost for for Maintain[PlaneAltitude]
      climbRate=f(plane.alt, route.alt)
```

```
thread implementation AP.impl
annex behavior_specification
{**
  states
     trust: Behavior::integer;
  transitions
     compute – [newTrust?(trust)
        when diff(alt,route.alt]
        -> compute};
**};
End AP.impl;
```

Figure 2: Requirements (KAOS) to AADL mapping.

Providing Connections with Industrial Development

Ensuring full traceability is critical in order that changes can be tracked and their impact assessed. It is therefore necessary to connect requirements models with more refined models at system and specification levels. While this connection can be loose, using simple traceability techniques, richer models enable a more elaborate and automated connection, or even the derivation of some design artefacts from the requirements level, hence reducing the effort required to produce and maintain them. This is currently being investigated in two projects with quite different approaches.

First, the FP7 DEPLOY (industrial deployment of system engineering methods providing high dependability and productivity) project aims at the industrial deployment of the Event-B method, a system-level variant of the B method. This method is quite generic and is being applied in sectors such as automotive, train, space and even e-business. Like B, it is mainly proof-based. It is supported by the RODIN (Rigorous Open Development Environment for Complex Systems) proof environment which is Eclipse-based and includes plug-ins for model-checking (ProB) and animation. Bridging the gap between requirements and Event-B models has been identified as a major topic.

Second, the ITEA2 SPICES (on Support for Predictable Integration of mission Critical Embedded Systems) project is domain-specific: it targets embedded systems and relies on a formal architectural description language called AADL. Verifications are oriented towards real-time properties and resources management. The connection with requirements models is currently developed both for the structural and dynamic parts of AADL, as shown in Figure 2. Again, Eclipse-based tools are available (TOPCASED).

Supporting Certification at High Assurances Levels

Systematic certification is required for safety-critical systems. A generic standard such as IEC-61508 defines safety integrity levels and guidelines. These generic guidelines are refined in more sector-specific standards such as space (ECSS - European Cooperation on Space Standardization), aeronautics (DO-178B) or railways (CENELEC 50126/8/9 - European Committee for Electrotechnical Standardization). Most of these also define a number of assurance levels in direct connection with risk (probability and impact). Higher assurance levels require a more formal demonstration of correctness. Formal models, which support the rigorous development of the system, can also directly support the certification process. This means not only that the effort required for certification is not increased for such systems, but that the perception of the certification process is also improved, since it is seen as being better integrated with the system products and not as additional work. Our current work mainly targets the aeronautics domain.

Links:
KAOS: http://www.info.ucl.ac.be/~avl/ReqEng.html
FAUST: http://faust.cetic.be
SPICES: http://www.spices-itea.org
DEPLOY:
http://www.deploy-project.eu
RODIN/Event-B:
http://www.event-b.org/
TELECOM:
http://www.skywin-telecom.be

Please contact:
Christophe Ponsard
CETIC, Belgium
Tel: +32 71 490 743
E-mail: christophe.ponsard@cetic.be

Modelling the Role of Software in the Propagation of Failures across National Critical Infrastructures

by Chris W. Johnson

In recent years, terrorist attacks, system failures and natural disasters have revealed the problems that many countries face in preparing for national civil contingencies. The diversity of critical infrastructures and the interconnections between different systems make it difficult for planners to anticipate everything. For example, the loss of power distribution networks can disrupt rail and road transportation systems. Knock-on effects can also be felt across telecommunications infrastructures as the uninterruptible power supplies (UPS) that protect mobile phone base stations fail over time. In addition, domestic water supplies are affected when pumping and treatment centres lose power.

It is difficult to underestimate the safety implications of these interdependencies. For example, Pironi, Spinucci and Paganelli describe how the Italian blackout of 2003 affected patients who relied on home parenteral nutrition systems. These individuals used electronic pumps for the overnight infusion of nutritional solutions, and the loss of power disrupted their treatment. Different devices responded in different ways, with some generating alarms and others reverting to battery power. Patients also responded in different ways, as they became worried about whether or not their systems had sufficient power to complete their treatment for that night. The blackout lasted several days across many areas of Italy. This created further problems, as stores of parenteral solution needed to be kept frozen. Other patients were placed at risk when the loss of power began to affect water treatment centres; for instance, it