

# Industrial Validator Whitepaper Airbus

<b>Nature:</b>	Report		
<b>Dissemination Level:</b>	Public		
	<b>Distribution to Participants</b>	<b>Additional Distribution</b>	
	ALL INTERESTED companies represented	<a href="mailto:interested_all@interested-ip.eu">interested_all@interested-ip.eu</a> <a href="http://www.interested-ip.eu">www.interested-ip.eu</a>	
<b>DocID:</b>	INTERESTED_whitepaper_Airbus	<b>Creation Date:</b>	17/5/2011

<b>Project</b>	INTERESTED	<b>Contract Number</b>	214889
<b>Author</b>	Cedrik Besseyre	<b>Organisation</b>	AIRBUS OPERATIONS S.A.S.

<b>Project</b>	INTERESTED	<b>Contract Number</b>	214889
<b>Internal Reviewers</b>	Francois-Xavier Dormoy Eric Bantegnie	<b>Organisation</b>	ESTEREL TECHNOLOGIES

# 1 Table of Contents

<b>1</b>	<b>Table of Contents .....</b>	<b>2</b>
<b>2</b>	<b>Motivation for the Airbus Validator .....</b>	<b>3</b>
2.1	Overview.....	3
2.2	Market Requirements.....	3
2.3	Cost-Benefits Estimation.....	3
2.4	Process and Tools used by Validator.....	4
<b>3</b>	<b>Technical implementation of the Airbus Validator.....</b>	<b>5</b>
3.1	Current Approach .....	5
3.1.1	Weaknesses of the Current Approach .....	5
3.2	INTERESTED Approach.....	5
3.2.1	Evaluating the use of SCADE System (MDT Papyrus / SCADE Suite tight integration .	6
3.2.2	Evaluating the use of SCADE Suite (Monitor) .....	6
3.2.3	Evaluating the use of ADA (Spark) code generator.....	6
3.3	Metrics.....	7
3.3.1	Effort.....	7
3.3.2	Support of new industrial work breakdown between several stakeholders .....	8
3.3.3	New dissimilar solution .....	8
<b>4</b>	<b>Appendix A – FWS application presentation.....</b>	<b>9</b>
4.1	Architecture .....	9
4.2	Display software .....	9
4.3	Logic and computation software .....	10

## 2 Motivation for the Airbus Validator

### 2.1 Overview

The aim of this document is to show how Airbus can use the INTERESTED workflow, what the expected benefits are, and how Airbus measured the improvement that the INTERESTED tool chain will provide to the entire system to software development process. In order to be able to provide such answer an industrial use case has been implemented using INTERESTED Tool chain.

A part of a Flight Warning System (FWS) has been implemented from System description down to integration on Target.

The Flight Warning System (FWS) manages faults and warnings coming from other systems and displays messages, recovery procedures, checklists (visual and audio) to the pilots. The application is made of several components (interactive, computation, logic, display) developed by several distinct teams.

HMI is subject to safety and cockpit ergonomics constraints and is changing a lot during the development phase.

The aim of the experiment was to show how to go further in the model driven engineering approach: to reduce the volume of textual specification, and thus enhance the early validation of the specification through simulation, and to extend the part of the application done by automatic code generation to ease the integration phase.

### 2.2 Market Requirements

Today, FWS development is involving several teams using different technologies. SCADE Suite is used for the logical part, SCADE Display is used for the Display part and manual coding is used for the Monitor part.

Strong links between all these components and the high rate of requirement updates during development phase leads to difficulties during integration and efficiency issues.

The main improvements provided by INTERESTED that need to be assessed are:

- Provide a global model based approach to the full application that federate all developments
- Formalize exchange between System and Software team and ease integration
- Use SCADE Suite improvements in Version 6 (iterators and State Machine) to design and code the Monitor Part
- Use SCADE Display improvements to design and code the HMI (Engine Warning Display, Engine Control Panel, for instance)
- Provide early host co-simulation capabilities to test and validate the application
- Ease changes in Monitor behavior requirements thanks to a modular and components-based architecture
- Investigate ADA code generation as a possible dissimilar second chain.

### 2.3 Cost-Benefits Estimation

The principal metrics used to assess the benefits adopting INTERESTED tool chain is the savings (in term of effort) made during Software Requirements, Detailed Design, Coding and Integration.

The benefit is estimated comparing the data collected during the Validator against the historical data collected by Airbus during previous developments, and applying factors to take into account the part addressed during the experiment.

Other metrics are secondary benefits like better support of new industrial work breakdown between several stakeholders and better sustainability of the development environment.

Until now, no global model based approach was used and an important part of the application is done in a traditional/manual coding process because design tools were not suited to model such application.

After applying INTERESTED workflow, usage of global model based approach together with formalized software-architecture model and early use of co-simulation saves half (precisely 52,3%) of the effort for specification, development and integration.

This estimate already takes into account the additional effort required for model-based design but doesn't take into account possible additional savings during verification phase (manual coding review) and the impact of late changes at system specification level.

This 52,3% saving is thus a conservative estimate of the possible real total saving.

Additional benefits will be presented. These additional benefits show a better support of new industrial work breakdown between several stakeholders and a possible solution for code generation dissimilarity.

## 2.4 Process and Tools used by Validator

An improved process and the corresponding tools have been exercised by the Validator to support the System to Software flow in a Model Based approach avoiding redundancies, inconsistencies while preserving Intellectual properties.

The integral process steps that have been exercised are:

- 1) Initial system-level model for function architecture (e.g. using MDT Papyrus). The model based tool supports the refinement into a software architecture model.
- 2) Export of part of the design allocated to software (preserve IP for the other part)
- 3) Import in SCADE Suite the contract defined by the System team and initialize synchronization mechanism between software design and System architecture and contract.
- 4) Design in SCADE Suite in sync with System architecture and contract
- 5) Design in SCADE Display in sync with System architecture and contract
- 6) Design links between software for co-simulation and integration
- 7) Co-simulate the complete system for testing or validation with user (Pilot)
- 8) ADA Code Generation (Spark)
- 9) Integration on target

## 3 Technical implementation of the Airbus Validator

### 3.1 Current Approach

Starting from high level architecture where components are identified, textual requirements are defined for each components. For most components (95%) representing more than 2 millions of lines of C code, a model based approach with formal requirements (using SCADE suite) is followed then coding using qualified code generator is used.

For some other components (5%) representing more than 1 hundred thousand lines of C code, a traditional process is followed with a global and refined software specification step using natural language (English), detailed design with SADT approach and then manual coding in C code.

#### 3.1.1 Weaknesses of the Current Approach

The main weaknesses of the current approach are:

- Lack of global model based approach for System architecture and software part to formally check consistency,
- Traditional/manual and costly process for some part of the application with a lot of modifications.
- Late co-simulation capabilities that delay the interaction with customer (Aircraft Pilot)
- Late performance assessment on target that may cause important re-design very late in the process
- Integration is painful because interfaces generally do not match from component to component.

### 3.2 INTERESTED Approach

The INTERESTED process flow will differ from the existing one according to the following guidelines:

- The architecture is designed using a model-based design tool; specifically Eclipse tool MDT Papyrus was evaluated for modelling the Architecture system specifications in SysML.
- The software architecture with the interfaces and the types is derived from the architecture using MDT Papyrus/SCADE Suite tight integration provided in SCADE System.
- Software description of the monitor is designed using SCADE Suite 6 with a modular approach that eases reuse and updates.
- Software description of the HMI is designed using SCADE Display 6.
- Co-simulation of the complete system or part of it is done to start testing and verification very early in the process
- ADA Code generation chain is explored as a solution for dissimilarity.

#### Benefits:

- 1) Provide global overview of the system
- 2) Capability to export a component contract/interface from a complete system to a dedicated team while preserving IP.
- 3) Capability to check the consistency of the interfaces between system and Software using automatic tools
- 4) Capability to check if the software implementation meets the system-level (SysML) description
- 5) Capability to predict performance of the application (WCET)
- 6) Capability to perform co-simulation on host very early in the process development.
- 7) Capability to ease requirement tuning with the customer (Pilot) thanks to early demonstrators

#### Penalties:

- 1) extra effort in terms of the capture of the Architecture design
- 2) extra effort in term of formalizing the monitor design (using SCADE Suite)
- 3) need to train the system team to a new tool and language
- 4) need to train the Monitor design team to a new tool and language
- 5) ADA code generation chain has (currently) no certification credit within Airbus

### 3.2.1 Evaluating the use of SCADE System (MDT Papyrus / SCADE Suite tight integration)

One goal is to verify if SCADE System can be used to capture architectural description and message data types in a convenient way.

From that formalized description of the architecture the goal is to keep consistent with implementation in order to prepare and ease integration. This is why, we expect that the automatic verification and consistency checks that are enabled by the adoption of this SysML approach will provide a big improvement during integration compared to traditional approach.

#### **Benefits:**

- 1) The Software architecture is described in a standard way, a software architecture description document is derived directly from the model
- 2) The Software is generated out of the architecture (Contract part) and consistency is checked during all project development.
- 3) The consistency verification between the architecture and the related SW implementation is performed with an automatic tool

#### **Penalties:**

- 1) a new artefact needs to be delivered and managed, extra effort must be planned
- 2) additional skills and competencies are required from the software architect engineer

### 3.2.2 Evaluating the use of SCADE Suite (Monitor)

The goal is to assess the usability of SCADE Suite (version 6) to design the monitor part of the FWS. With SCADE Suite version 5 and before it was not possible (scalable) to design the monitor because this design is using a lot of regular loop (iterator) computations with imperative (decision diagram and state machine) style design.

The design is very modular in order to allow reuse and allow limited impact on design when requirement are updated.

For instance changing from a 18 lines FWS display (EWD) present in Airbus 380 to a 15 lines FWS display present in A400M, is now limited to a SCADE suite Constant update.

#### **Benefits:**

- 1) Allow automatic code generation (saving)
- 2) Ease update when requirements are updated thank to a modular approach
- 3) Allow reuse from Aircraft to Aircraft
- 4) Allow performance estimation (WCET) very early in the process
- 5) Allow early execution on host to get customer feedback (Pilot).

#### **Penalties:**

- 1) Additional skills are required from software design engineers

### 3.2.3 Evaluating the use of ADA (Spark) code generator

The goal is to assess the usage of ADA code chain as a possible dissimilar chain.

**Benefits:**

- 1) Provide a dissimilar solution

**Penalties:**

- 1) No ADA compiler certification credit is existing at Airbus
- 2) Additional skills are required from software embedded engineers
- 3) Runtime error analyser available on Ada need to be found and experimented
- 4) WCET analyser needs to be further experimented and qualified (as verification tool) prior to industrial usage

**3.3 Metrics**

Airbus, in order to estimate cost/benefit in adopting the INTERESTED tool chain uses the **main metrics** (number of days spent in design and coding) and as **secondary metrics** provide rationale how INTERESTED tool chain provides competitive advantage to Airbus for critical embedded software in conformity with the DO-178 B standard.

**3.3.1 Effort**

The evaluation of the savings and improvements on the SW development process requires the extraction of measured data on development costs/efforts and comparison with similar activities done during the Validator.



Only the requirements, design, coding and integration tasks and only a subset of the manual part of FWS is considered.

Reference effort collection has been done on current projects (ie before application of the Interested process).

An estimation of the effort using the INTERESTED process has been made considering the time used for the validator itself and the portion of the entire design that has been addressed in the Validator.

The table below summarizes the estimation of the savings

**FWS Interested Savings**  
April 20th, 2011

Breakdown	Reference Effort <sup>1</sup> Breakdown (Current Process)	Interested Effort Breakdown	Gain	Gain (%)	
Software Requirements	375	0,0	375,0	100,0%	Full Saving
Detailed Design	90	266,7	-176,7	-196,3%	
Coding	75	6,7	68,3	91,1%	
Integration	75	20,0	55,0	73,3%	
<b>TOTAL</b>	<b>615</b>	<b>293</b>	<b>322</b>	<b>52,3%</b>	

<sup>1</sup> All efforts are expressed in days

The table above shows that the use of INTERESTED tools and especially their interoperability provides an estimated 52% gain in term of effort.

This gain is mainly due to the benefit linked to model-based processes and Automatic Code generation, completed with the combined use of SCADE and SysML to guarantee the consistency of the data between System and Software teams..

It is expected that the application of this workflow will reduce integration efforts and the rework of specification, design and code related currently to coherence issues that are revealed late in the integration process.

The gain due to the reduction of effort needs also to take into account the frequent changes in the specification that was not measured and should increase again the expected gain.

### 3.3.2 Support of new industrial work breakdown between several stakeholders

Airbus is more and more acting as an Aircraft Systems Integrator, and manages its Suppliers as true Partners through the Extended Enterprise concept. Airbus aims at being an Architect and Integrator and establishes partnership with Suppliers to master the technologies.

The Extended Enterprise means sharing Airbus internal processes with main partners use the same common tools and follow the same processes but also preserve Intellectual Properties.

In that Area, Interested brings important improvements:

- Export System subcomponent with all but only the needed artefact needed for the component.
- Diff capabilities at system level to highlight in a document that can be support of contractual relationship, the differences between 2 versions of the system architecture.

### 3.3.3 New dissimilar solution

For some Safety critical systems such as FCS (Flight Control System) many redundancies are defined (several sensors, several computers) and dual dissimilar architecture is selected with FCS Prim (Primary) and FCS Sec (Secondary).

For these two systems, a dissimilar development process is defined.

With Ada as a new target language for KCG, new improved dissimilarity solution can now be explored.

On one hand (Prim): Dedicated Team A can design SCADE 5 Dataflow Model A, KCG 5 can be used to generate C Code, C cross compiler used.

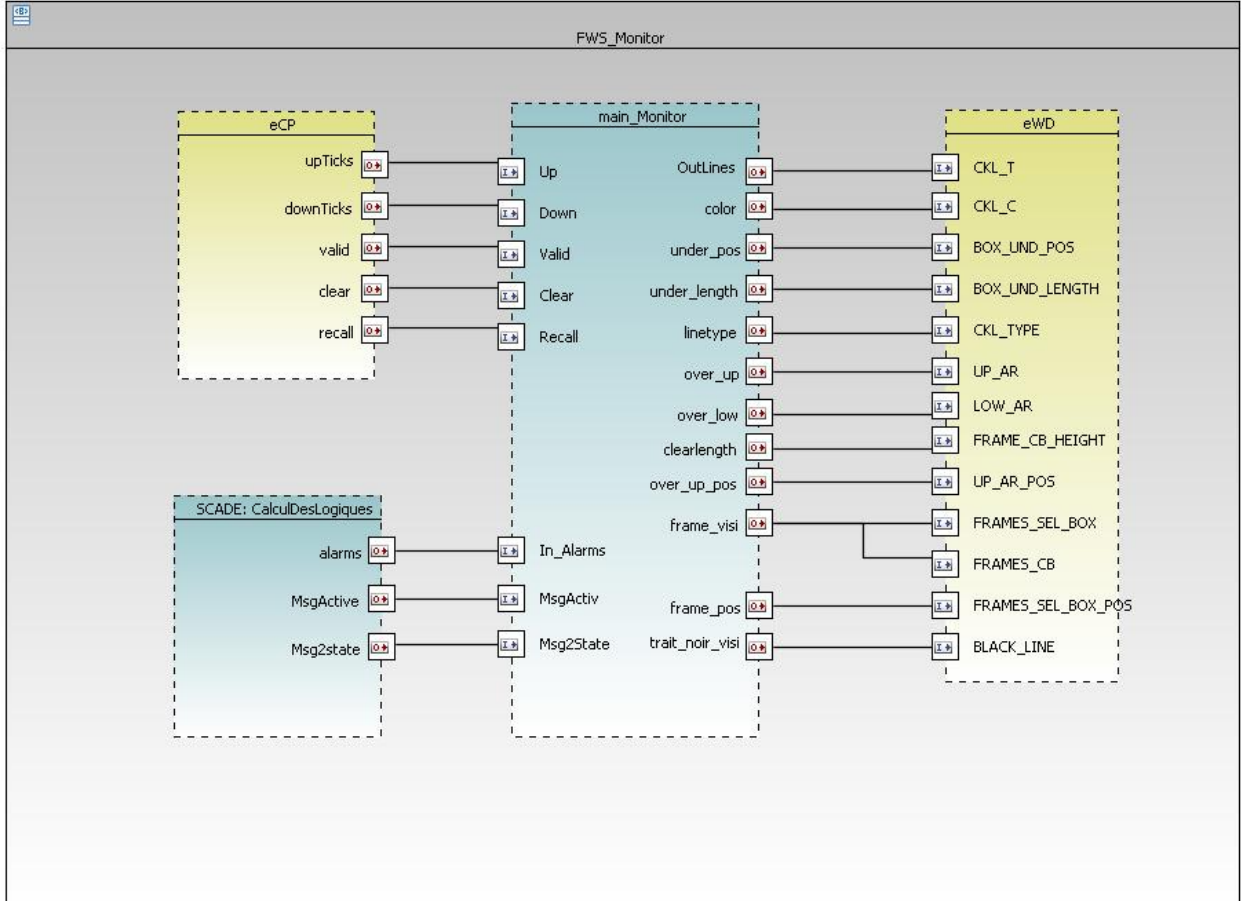
On the other hand (Sec): Dedicated Team B can design SCADE 6 State Machine/iterators Model B, KCG 6 can be used to generate ADA Code, ADA cross compiler used.

It is important to notice that KCG 5 and KCG 6 are completely different code generators using different architecture and coding language (C vs Caml) with Zero commonalities.

## 4 Appendix A – FWS application presentation

### 4.1 Architecture

Here is a description of the complete FWS architecture in SCADE System/Papyrus.



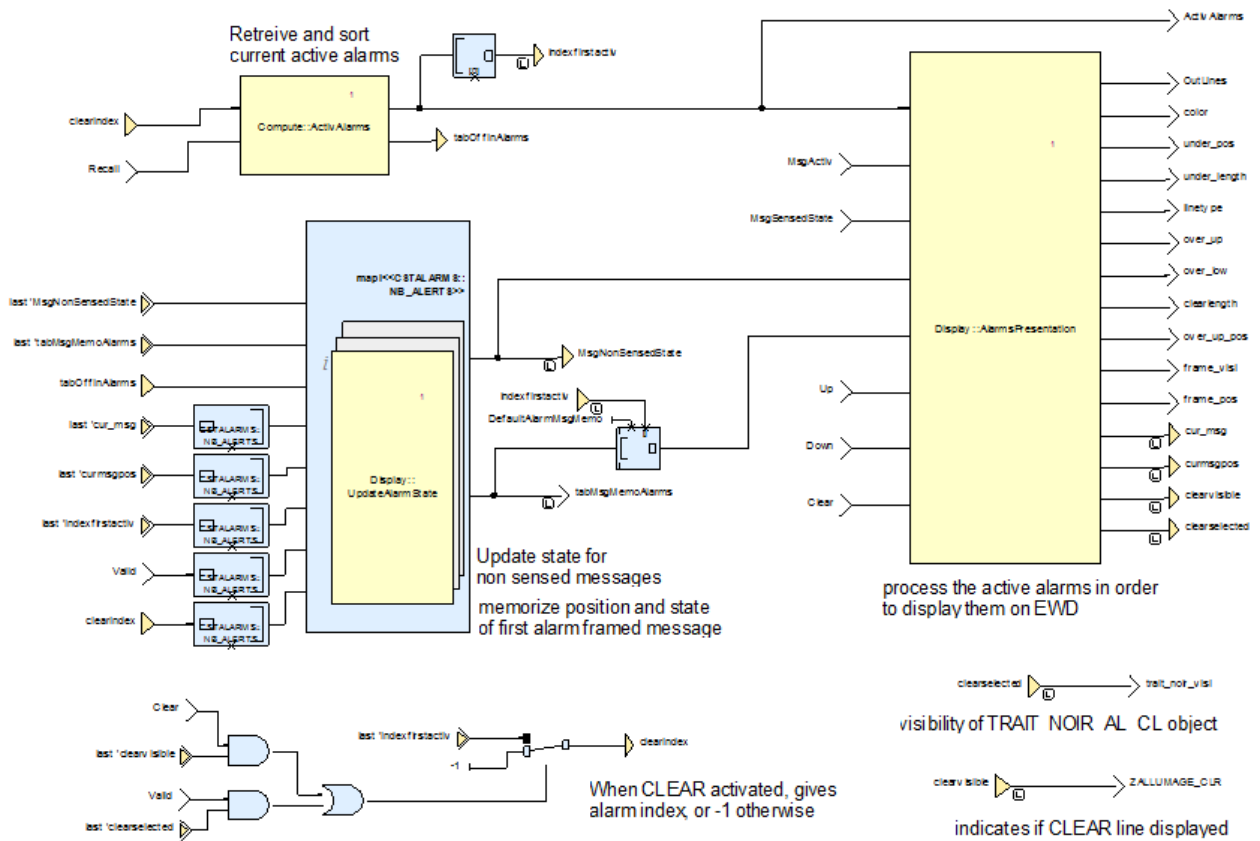
### 4.2 Display software

SCADE Display 6 is used to model the EWD part. An example is shown below.

```
Graphical Panel - EWD
F/CTL_SINGLE ELEVATOR FAULT
.WHEN ACFT STABLE :
□ CONFIRM ACFT STABLE
□ -L OUTR ELEVR POSITION .....CHECK
□ -R OUTR ELEVR POSITION .....CHECK
  .IF L OUTR ELEVR POS NOT AT 0° :
□ CONFIRM L OUTR ELEVR POS NOT AT 0°
  .IF R OUTR ELEVR POS NOT AT 0° :
□ CONFIRM R OUTR ELEVR POS NOT AT 0°
□ MAX SPEED : 200 KT
□ -STABILIZE FLIGHT PATH FOR MONITORING
□ CONFIRM FLIGHT PATH STABLE
...
  .IF L OUTR ELEVR POS NOT AT 0° :
□ MAX SPEED : 200 KT
```

### 4.3 Logic and computation software

SCADE Suite 6 is used to model the Monitor part, an example is show below



Note : SCADE Suite was also used to model the “Calculdeslogiques” part of the FWS.