ERTS 2014 session 3B.3 - February 5th

Efficient modelling of avionics systems: combining standard language and custom editor

Presented by Raphaël Faudou (Samares Engineering)
Prepared by Raphaël Faudou, Laurent Duffau (Airbus)
Patrick Farail (Airbus) and Camille Louge (Atos)

Agenda

- Introduction
- Different strategies on MBSE tooling at Airbus during last decade
- Two modelling experiences above standard editor
  - TOPHOO
  - FAST
- Conclusions and perspectives

Introduction

- **Feedback from industry**
  - Critical embedded systems more and more subjected to high safety requirements whatever the domain
  - More and more difficult for industrial actors to specify such systems with high quality
- **MBSE helps…**
  - Requirement errors or unconsistencies are detected earlier (thanks to formalizing of requirements into model elements)
- **… but is not enough for wide dissemination in operations**
  - We now need good tooling to support efficient modelling !

Context N° 1 :
software development at Airbus internal entity for avionic equipment

Context N° 1 (SW) : 10 years ago

- **First approach = UML, in 2003**
  - Diversity in modelling tools,
  - Benefit from a large panel of engineers already trained
  - Lot of training material (books) and consulting if needed
  - Unified and unambiguous semantics

- **Usage context**
  - A380 equipment
  - Main motivation = improve communication between teams with UML pragmatic approach

  ➔ Not all UML 1.3 diagrams were used: mainly UC, sequence and classes
  - Commercial tool

Context N° 1 (SW) : first lessons

- **Reduced textual part of specification and architecture documents**
- **Still tedious and sometimes inefficient and error prone**
  - too many concepts,
  - hard to ensure consistency between diagrams and for the whole model.

Context N° 1 (SW) – 7 years ago

- **The Topcased initiative, from 2005**
  - An open source solution as it was considered the best strategy to ensure long term availability of key tooling
  - A model based engineering toolkit to detect errors earlier than with traditional document approach and reduce verification efforts
  - An eclipse-based platform to ensure modularity and extensibility (plug-ins)

- **Airbus learnt a lot (techniques technologies) about modelling editors and changed their minds (software level)**

  "We can specify and develop our own modelling editors at low cost"

- **Birth of « SAM », a DSML based on SA-RT method**
  - Functional software decomposition,
  - Associated flows (control, data and message)
  - Behaviour (through automata)..

Context N° 1 (SW) – other lessons

- **Good points**
  - SAM modelling editor was simple to use
  - Good adoption at software level and even in other departments including design office

- **Issues**
  - Lack of extensibility: not so easy to update the meta model when models already exist in operational context…
  - Not standard → difficult to find support → finally yet another editor to maintain…

Context N° 1 (SW) - 2 years ago

- **The TOPHOO initiative**
  - Opportunity to develop a new modelling tool supporting Embedded Software design activity: component breakdown structure to prepare the coding phase

- **Functional requirements**
  - Manage components (called "Machines" in Airbus terminology) that can define following elements
    - Services (operations) with some that are exported (made public)..
    - Types
    - Constants and resources
  - Decompose root machine (the equipment or subset to address) into several sub machines and terminal machines (at lowest level)

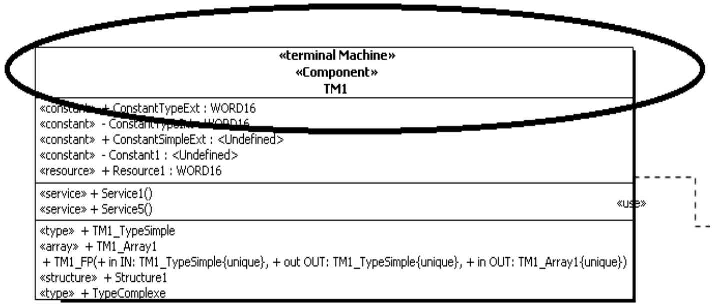
Context N° 1 (SW) - 2 years ago

- **Key criteria for the solution**
  1. Use UML language for standard conformance and all associated benefits…
  2. …but with strong customization to provide "simple" modelling editors adapted to end user vocabulary and process
  3. Open source solution (ensure long term availability)
  4. Based on EMF (Eclipse Modeling Framework) to benefit from powerful API
  5. Reuse existing modelling editor to benefit from Topcased efforts
  6. Ability to integrate Topcased mature components around the modelling editor : GenDoc, Tpc Req, OCL, UML2EC

Papyrus, a good starting point

SAMARES ENGINEERING
Accelerate Systems Design

- Can define a UML profile to map Airbus vocabulary on UML concepts
  - Machine->component,
  - Service->operation,
  - Resource and Constant->Property
  - …
- Can define easily a palette matching UML profile
- Can customize property view for specific layout

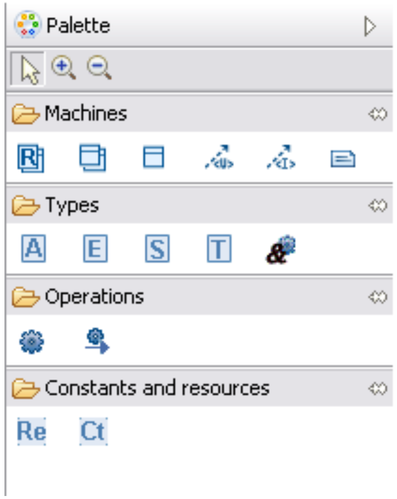


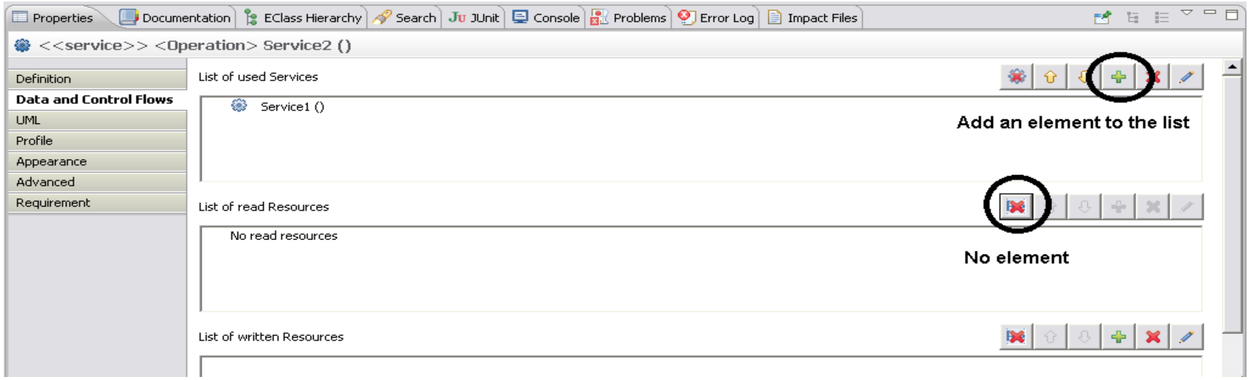
«terminal Machine»
«Component»
TM1

«constant» + ConstantTypeExt : WORD16
«constant» - ConstantTypeInt : WORD16
«constant» + ConstantSimpleExt : <Undefined>
«constant» - Constant1 : <Undefined>
«resource» + Resource1 : WORD16

«service» + Service1()
«service» + Service5()

«type» + TM1_TypeSimple
«array» + TM1_Array1
+ TM1_FP(+ in IN: TM1_TypeSimple{unique}, + out OUT: TM1_TypeSimple{unique}, + in OUT: TM1_Array1{unique})
«structure» + Structure1
«type» + TypeComplexe





<<service>> <Operation> Service2 ()

List of used Services

Service1 ()

Add an element to the list

List of read Resources

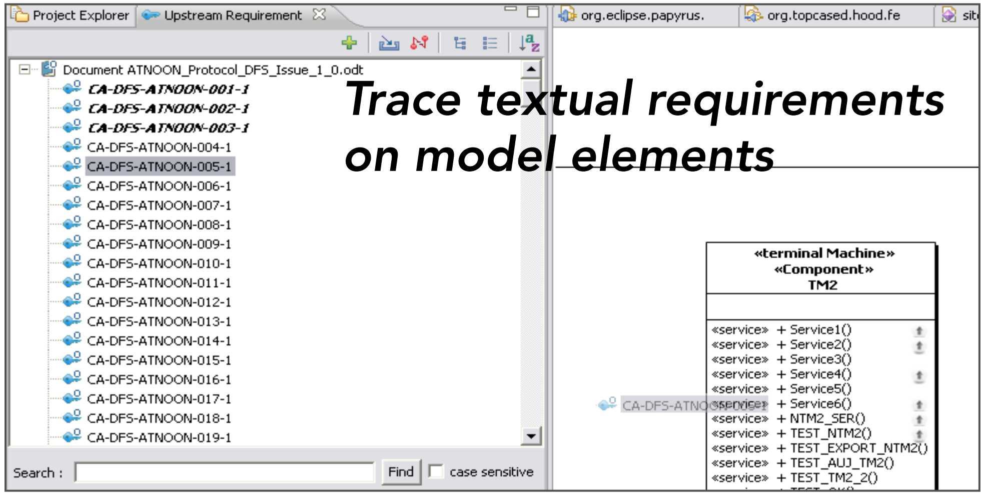
No read resources

No element

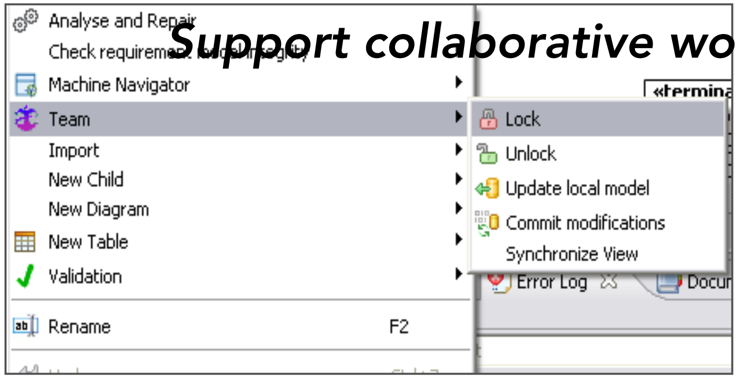
List of written Resources

Papyrus + Topcased is better !

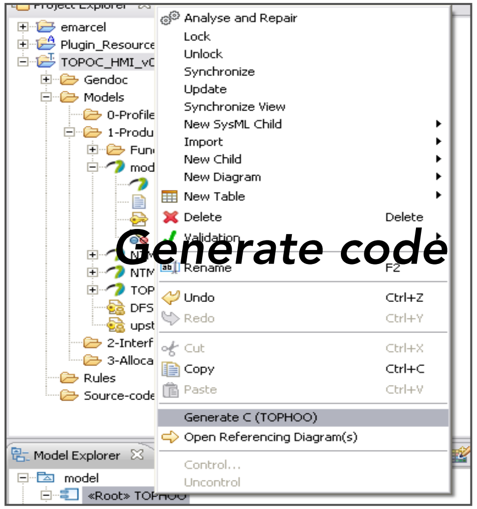
SAMARES ENGINEERING
Accelerate Systems Design



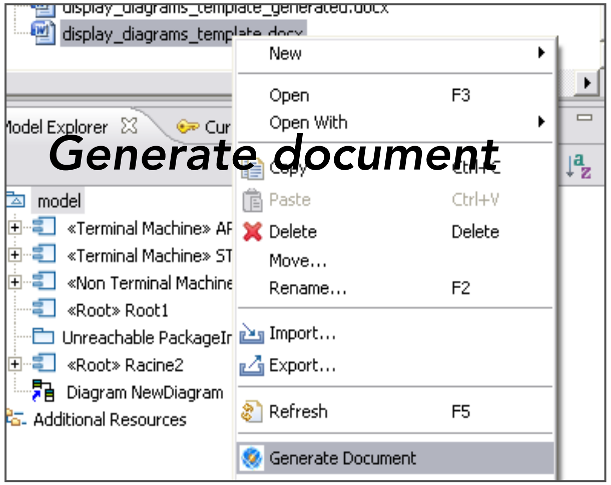
Trace textual requirements on model elements



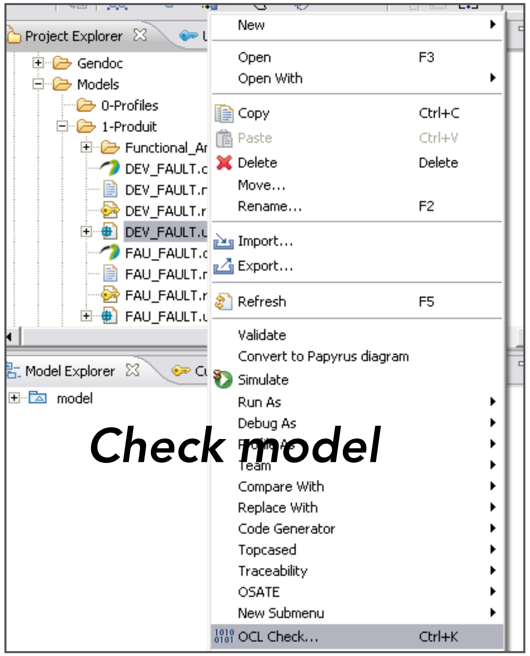
Support collaborative work



Generate code



Generate document



Check model

Context N° 2:
system architecture modelling - operational and functional views

Context N° 2 – Last decade

- **10 years ago at design office…**
  - No modelling standard to describe architecture
  - Use of Microsoft Visio for communication…

- **R&T studies underlined interest for top/down functional approach**
  - Introduction of Vitech CORE to describe operational scenarii and simulate resource consumption (timing diagrams)
  - IBM Rhapsody to describe functions and their decomposition and simulate scenarii (sequence diagrams)

Context N° 2 – Lessons learnt

- **Good points**
  - Vitech Core appreciated for simplicity and usability
    - "systems engineer" oriented, no long list of UML properties,…
  - IBM Rhapsody useful to generate simulation traces
- **Issues with Vitech Core**
  - Performance bottleneck to access database
  - Migration issues because of database customization (not standard)
- **Issues with IBM Rhapsody**
  - considered as too complex because it "sticks" to SysML notation…

    …and SysML remains complex (many concepts and diagrams …)

    … with too generic concepts: "block" instead of "function"

Context N° 2 – 2 years ago

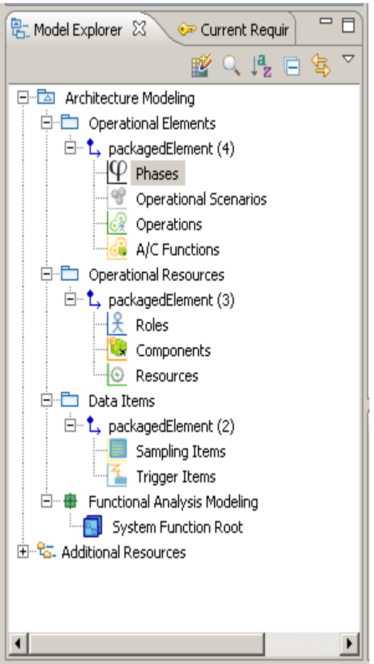
- **A need**
  - Find or develop a modelling tool able to support a generic Model Based methodology for Systems Architecture

- **Key decision criteria**
  - Intuitive and customizable (no need to learn another language)
  - Standard (OMG SysML)
  - Do not reinvent the wheel (reuse as much as possible, TOPHOO?)

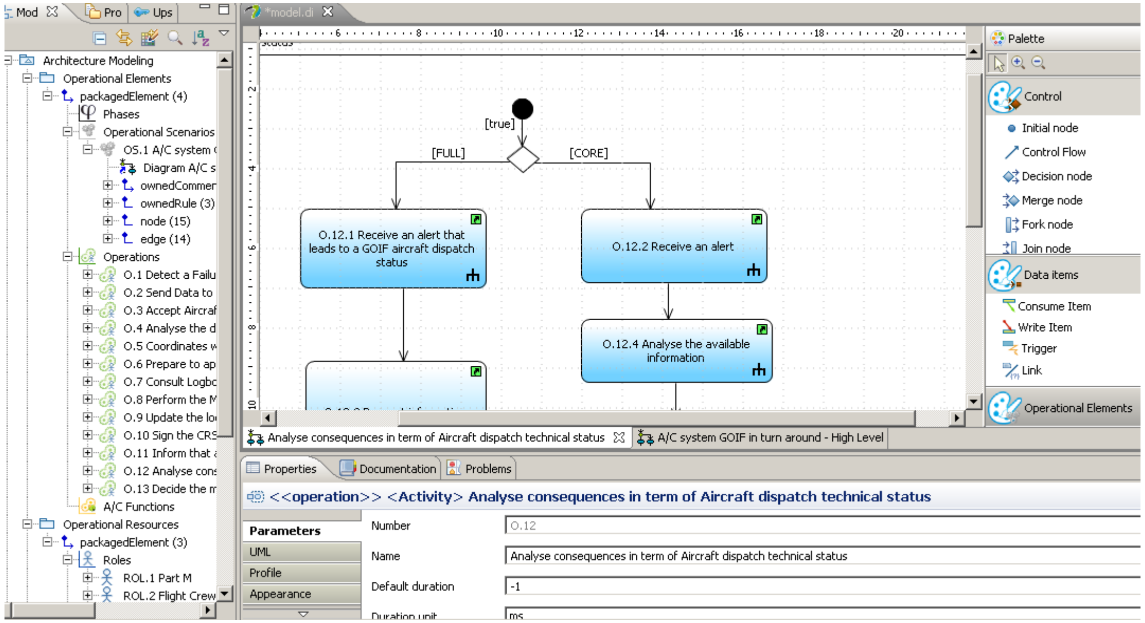
- **There were several contacts with TOPHOO team to learn from their experiment on Papyrus customization**

Following TOPHOO…

- **FAST = Airbus experiment to support a generic Model Based methodology for Systems Architecture operational and functional views with customized SysML**
  - Papyrus considered as best approach (like TOPHOO)…
  - … but with customization on top of SysML (which is already a profile…)



- Architecture Modeling
  - Operational Elements
    - packagedElement (4)
      - Phases
      - Operational Scenarios
      - Operations
      - A/C Functions
  - Operational Resources
    - packagedElement (3)
      - Roles
      - Components
      - Resources
  - Data Items
    - packagedElement (2)
      - Sampling Items
      - Trigger Items
  - Functional Analysis Modeling
    - System Function Root
- Additional Resources



- Architecture Modeling
  - Operational Elements
    - packagedElement (4)
      - Phases
      - Operational Scenarios
        - OS.1 A/C system
          - Diagram A/C s
          - ownedCommer
          - ownedRule (3)
          - node (15)
          - edge (14)
      - Operations
        - O.1 Detect a Failu
        - O.2 Send Data to
        - O.3 Accept Aircraf
        - O.4 Analyse the d
        - O.5 Coordinates w
        - O.6 Prepare to ap
        - O.7 Consult Logbc
        - O.8 Perform the M
        - O.9 Update the lo
        - O.10 Sign the CRS
        - O.11 Inform that
        - O.12 Analyse cons
        - O.13 Decide the m
      - A/C Functions
  - Operational Resources
    - packagedElement (3)
      - Roles
        - ROL.1 Part M
        - ROL.2 Flight Crew

<<operation>> <Activity> Analyse consequences in term of Aircraft dispatch technical status

Number: O.12

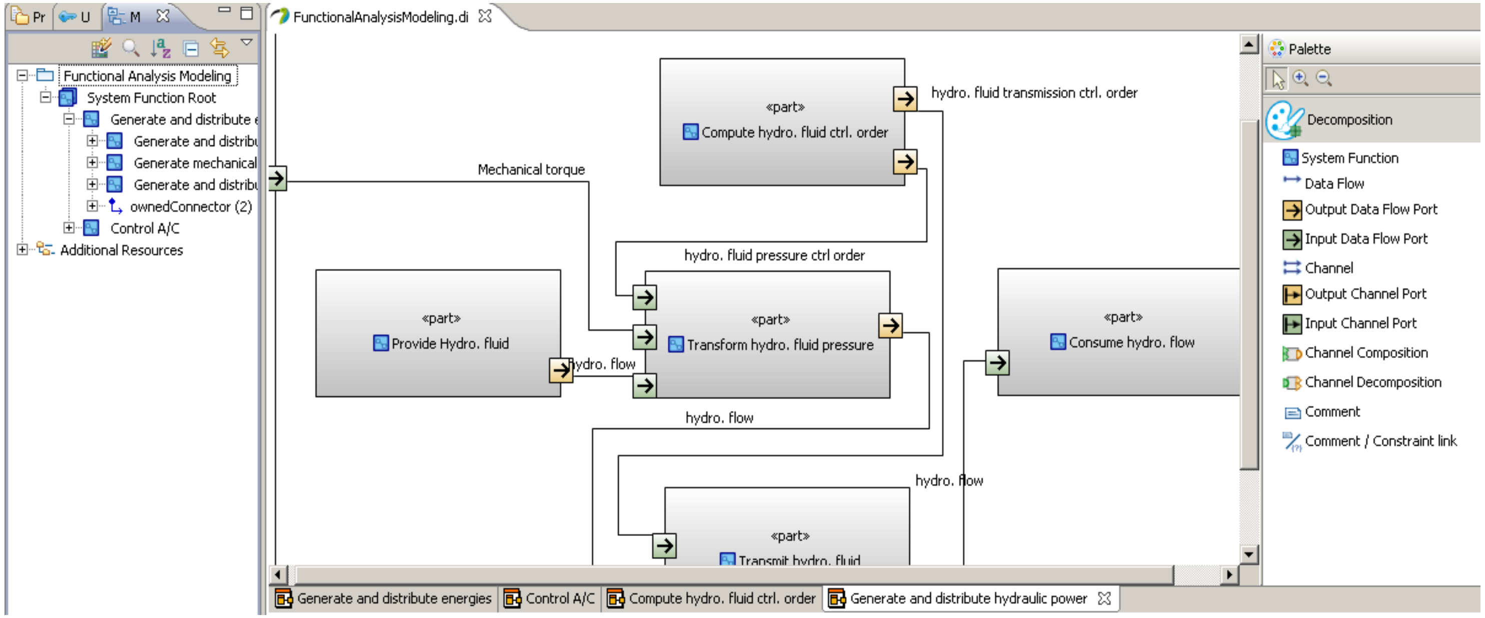
Name: Analyse consequences in term of Aircraft dispatch technical status

Default duration: -1

Duration unit: ms

FAST, a Papyrus SysML customization

- **No need to learn SysML notation…**
  - Useful system concepts are available in the palette and only them



- **… but produced model conforms to SysML !**
  - Can be consumed by SysML tools: model checking, documentation generation, simulation…

SAMARES ENGINEERING
Accelerate Systems Design

Conclusions and perspectives (both contexts)

Conclusions - TOPHOO

- **TOPHOO solution has been industrialized and is used in production on a software equipment of A400M aircraft program (safety level = C)**

- **Good points**
  - Good operational feedback
  - Performance and functional improvments on Papyrus editor
- **Warnings**
  - Deployment impact of several papyrus customizations : Improve modularity of customization (through plug-ins)
- **Rooms for improvement**
  - Customisation flexibility at user level to reduce customization development costs

Conclusions - FAST

- **FAST is currently used by system designers on R&T programs**

- **Good points**
  - Good operational feedback because of the simple HMI (system designers only manipulate the needed business objects = system functions, data flows,…)
  - Efficiency: 3 times faster to create the same model compared to not customized SysML tool

- **Rooms of improvments**
  - Complete model checking rules
  - Better integration between operational and functional views

Perspectives

- **TOPHOO**
  - Real-time architecture above MARTE
  - Specification of complex behaviour : protocols, HMI …
  - Migration to Eclipse PolarSys industrial working group
- **FAST**
  - Integration of requirement traceability management
  - Interoperability with Scade System for Software implementation
  - Migration to Eclipse PolarSys industrial working group

Thank you for your attention

SAMARES ENGINEERING
Accelerate Systems Design

- Any question?