Why are ARCADIA and Capella relevant for MBSE?
Introduction: Pascal Roques

- Senior consultant, 25+ years of modeling experience
  - SADT, OMT, UML, SysML, ARCADIA/Capella

- UML2 and SysML Certified by the OMG

- Co-founder of the SysML

- Trainer for Thales on ARCADIA / Capella
  - 130+ sessions, 1500+ trainees
  - Part of Clarity

- Author of UML/SysML best-sellers in France

- … and of the first Capella book soon!
Agenda

1. Tooled-Up Modeling Method
2. Systems Engineering Continuity
3. User-Driven Modeling Tool
4. Open-Source!
MBSE pillars (SysML)

Language

Tool

Method (TBD!)
MBSE pillars (ARCADIA / Capella)
## ARCADIA Summary

### Need

**Customer Operational Need Analysis**
- What the users of the system need to accomplish
  - Define operational capabilities
  - Perform operational need analysis

**System/SW/HW Need Analysis**
- What the system has to accomplish for the users
  - Perform capability trade-off analysis
  - Perform a functional and non-functional analysis
  - Formalise and consolidate requirements

### Solutions

**Logical Architecture Design**
- How the system will work so as to fulfil expectations
  - Define architecture drivers and viewpoints
  - Build candidate architectural breakdowns in components
  - Select best compromise architecture

**Physical Architecture Design**
- How the system will be developed & built
  - Define architectural patterns
  - Consider reuse of existing assets design a physical architecture
  - Design a physical reference architecture
  - Validate and check it

### Concepts

- Operational capabilities
  - Actors, operational entities
  - Actor activities
  - Interactions between activities & actors
  - Information used in activities & interactions
  - Operational processes sequencing activities
  - Scenarios for dynamic behaviour

- Actors and system capabilities
  - Functions of system & actors
  - Dataflow exchanges between functions
  - Functional chains traversing dataflow
  - Information used in functions & exchanges, data models
  - Scenarios for dynamic behaviour
  - Modes & states

### Description Means

**Dataflow**
- Functions, op. activities interactions & exchanges

**Scenarios**
- Actors, system, components interactions & exchanges

**Functional chains**
- Operational processes through functions & op. activities

**Modes & states**
- States of actors, system, components

**Breakdown of functions & components**

**Data model**
- Dataflow & scenario contents, definition & justification of interfaces

**Component wiring**
- All kinds of components

**Allocation**
- Op. activities to actors, functions to components, impl-components to impl-components, dataflow to interfaces, of elements to configuration items
Methodological Guidance

▪ Activity Browser

- Define Stakeholder Needs and Environment
  - Capture and consolidate operational needs from stakeholders
  - Define what the users of the system have to accomplish
  - Identify entities, actors, roles, activities, concepts

- Formalize System Requirements
  - Identify the boundary of the system, consolidate requirements
  - Define what the system has to accomplish for the users
  - Model functional dataflows and dynamic behaviour

- Develop System Logical Architecture
  - See the system as a white box, define how the system will work so as to perform a first trade-off analysis

- Develop System Physical Architecture
  - How the system will be developed and built
  - Software vs. hardware allocation, specification of interfaces, deployment configurations, trade-off analysis

- Formalize Component Requirements
  - Manage industrial criteria and integration strategy: what is expected from each designer/sub-contractor
  - Specify requirements and interfaces of all configuration items

- Transition From Operational Activities
  - [SFB] Create a new Functional Breakdown diagram

- Define Actors, Missions and Capabilities
  - [SDFB] Create a new Functional Dataflow Blank diagram

- Refine System Functions, describe Functional Exchanges
  - [FS] Create a new Functional Scenario

- Allocate System Functions to System and Actors
  - [SAB] Create a new System Architecture diagram
  - [ES] Create a new Exchange Scenario
Methodological Guidance

- Semantic Browser
Methodological Guidance

- Model Checking
Methodological Guidance

- Semantic Color Map

  - Functional analysis = Green
  - Components engineering = Blue
  - Data models and interfaces = Pink
Agenda

1. Tooled-Up Modeling Method

2. Systems Engineering Continuity

3. User-Driven Modeling Tool

4. Open-Source!
ARCADIA: Global View

What the users of the system need to accomplish

What the system has to accomplish for the users

How the system will work to fulfill expectations

How the system will be developed and built
Automated Transitions

- Functions / Actors / etc.

Logical Architecture:
- System Analysis
- Logical Architecture
- Physical Architecture

- Transition from System Functions
- Define Logical Components and Actors
- Allocate Logical Functions to Logical Components
- Delegate System Interfaces and create Logical Interfaces
- Enrich Logical Scenarios
- Transverse Modeling

Logical Architecture:
- System Analysis
- Logical Architecture
- Physical Architecture

- Transition from System Functions
- Define Logical Components and Actors
- Allocate Logical Functions to Logical Components
- Delegate System Interfaces and create Logical Interfaces
- Enrich Logical Scenarios
- Transverse Modeling

- Perform an automated transition of System Functions
- Create Traceability Matrix
Automated Transitions

- Even Scenario Transition!
Functions / Components / Data

- Components
- Functions
- Exchange Items
- Types

Diagram showing relationships between components, functions, exchange items, and types.
System/Subsystem Transition
Replicable Elements and Libraries
Architecture Evaluation

- Viewpoints
Capella Studio

- Integration of Capella in Kitalpha
- Definition of a Capella Target Application
- Integration of Capella-specific generators
- Extensions of the textual editors
- Customization of the html documentation generation
Architecture Evaluation

- Coupling with External Tools
  - Example: Safety Architect

Feared event added to Capella dataflows (viewpoint)

In Safety Architect, analysis of block local failure conditions

Functional Hazard Analysis (FHA)

In Capella, visualization of fault trees as critical functional chains

In Safety Architect, automated generation of fault-trees
True MBSE!
Agenda

1. Tooled-Up Modeling Method
2. Systems Engineering Continuity
3. User-Driven Modeling Tool
4. Open-Source!
Computed Links
Advanced Diagram Management
Powerful Accelerators!

- Initialization from existing diagram:
  - Initializes the current diagram according to an existing diagram defined in the previous architecture. It adds realizing elements for each element from the source diagram and preserves layout between diagrams. This tool does not modify the semantic model.

- Selection Wizard:
  - Select existing diagram for initialization.
  - Select a name to find
    - ? = any character, * = any string
  - type filter text
  - Available options:
    - [SAB] EOLE
    - [SAB] EOLE External View
    - [SAB] EOLE Functional View
    - [SAB] EOLE Synthetic View

- Image of a diagram showing a logical system with various components such as Measurement Engineer, Launch Date Acquisition, Air Temperature, Air Humidity, and Foreach Weather.
Model Diff / Merge
Agenda

1. Tooled-Up Modeling Method
2. Systems Engineering Continuity
3. User-Driven Modeling Tool
4. Open-Source!
Capella Ecosystem

Initial 3-year (French) collaborative project

Larger industry consortium currently being initiated
Web Sites:

- www.polarsys.org/capella
- //wiki.polarsys.org/Capella
- //polarsys.org/forums/index.php/f/13/

- www.obeo.fr/en/capella-professional-offer

- www.prfc.fr

- www.clarity-se.org/