THALES
Building a future we can all trust

Where to Start with MBSE when Thousands of System Requirements Are Already Defined

Laura Mallon, Peter Havenga

Talk content

- Project & context
- Modelling goals
- System of System model overview
- Engineering Environment
- Best practices

OneSKY Australia Program

Complex Transformation of Air Traffic Management in Australia

- Centralised Collaborative Command and Control
- Broad Utilisation of Workforce
- Optimised Network and Flight Efficiency

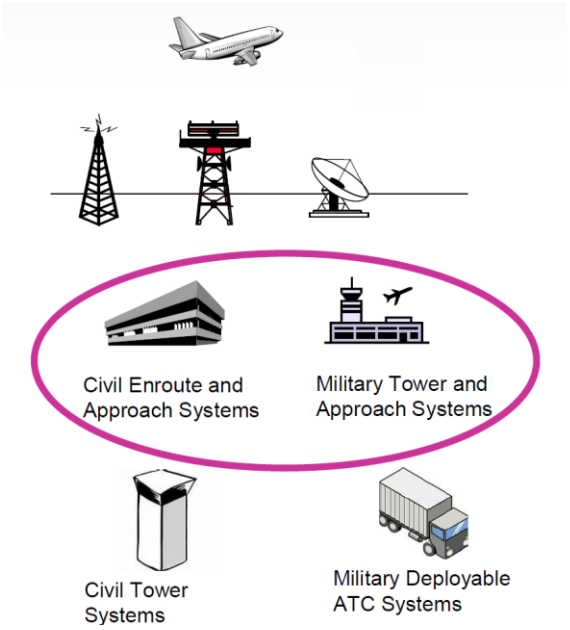
Replace existing air traffic management systems with an advanced integrated Civil and Military Air Traffic Management System (CMATS)

PLM Reference 0000024497 Revision 004 Revision Date 21.05.2021
Template: 87211168-DOC-GRP-EN-006

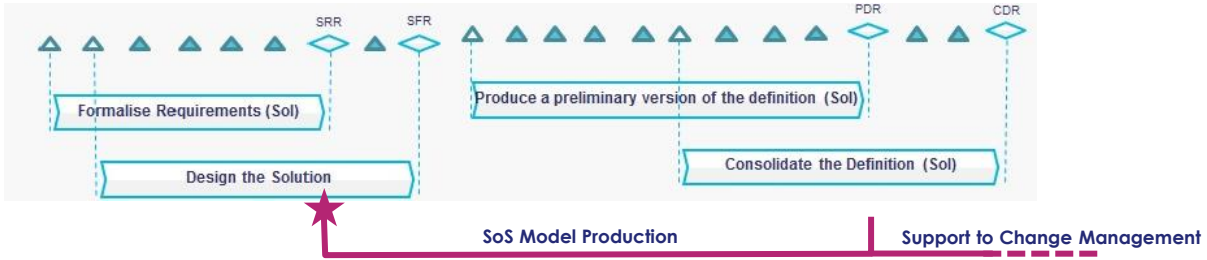
OPEN

CMATS Overview – Complex System

- Deployed at 12 Sites
- 10+ Subsystems
- 50+ External Interfaces to legacy Systems e.g. Surveillance, MET Data
- Inter Site Communication
- Rolled out over 3 incremental phases



Starting Point for MBSE



Starting point for MBSE post Solution Requirement Review (SRR)

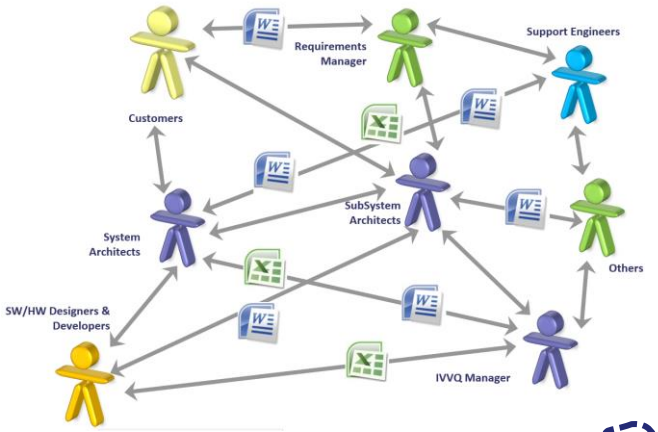
- 6000+ solution requirements identified and managed in DOORS
- Allocation of requirements to subsystem breakdown defined
- Interface Control Documents (ICDs)definition process agreed

CMATS is a complex system that must be clearly defined and agreed by a large number of stakeholders.

PLM Reference 0000024497 Revision 004 Revision Date 21.05.2021
Template: 87211168-DOC-GRP-EN-006

OPEN

MBSE Transformation

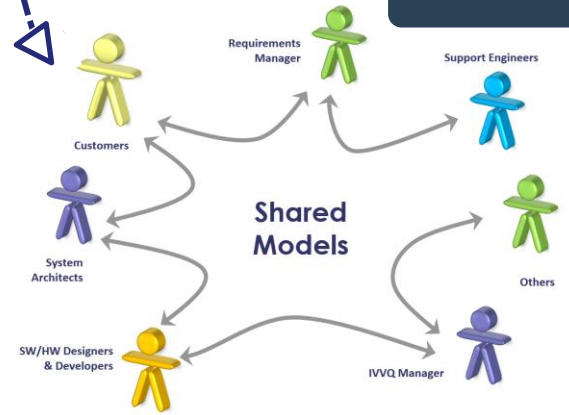


Standard practices

Standalone documents and models

Incoherent reference and decisions

Poor continuity between engineering levels



MBSE

Shared models with multiple views

Collaborative workflow and unique reference

Common unified language/concepts

System Modelling Goals

Provide an accurate, referenced and available architecture definition that provides:

- A solid ground for design decisions and for the descriptions of the system and subsystems
- A definition of how the CMATS solution will be delivered along its lifecycle
- A definition of how the deployment of subsystems varies between sites.
- Site Connectivity
- Support to engineering change requests

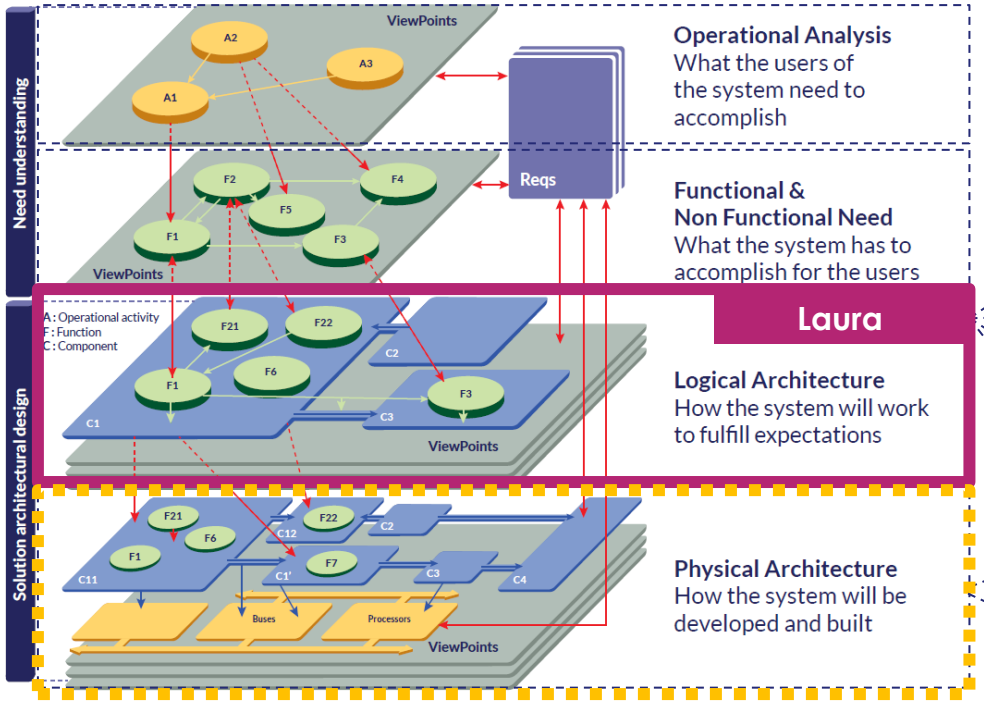
Demonstrate how the SoS Design satisfies the Requirements

- For those requirements that are fulfilled by more than one subsystem – demonstrate the role of each subsystem in that requirement. (Functional Chains)
- Ensure the definition of internal interfaces between subsystems are understood and complete

Key Input into System Design Document (SDD)

- Document Generation

System of Systems Architecture System Model Content overview



- System and Subsystem Definition "Black Box"
- External and Internal Interface Definition
- Functional Chain Definition
- Coupling to system requirements
- A definition of how the deployment of subsystems varies between sites.
- Site Connectivity

PLM Reference 0000024497 Revision 004 Revision Date 21.05.2021
Template: 87211168-DOC-GRP-EN-006

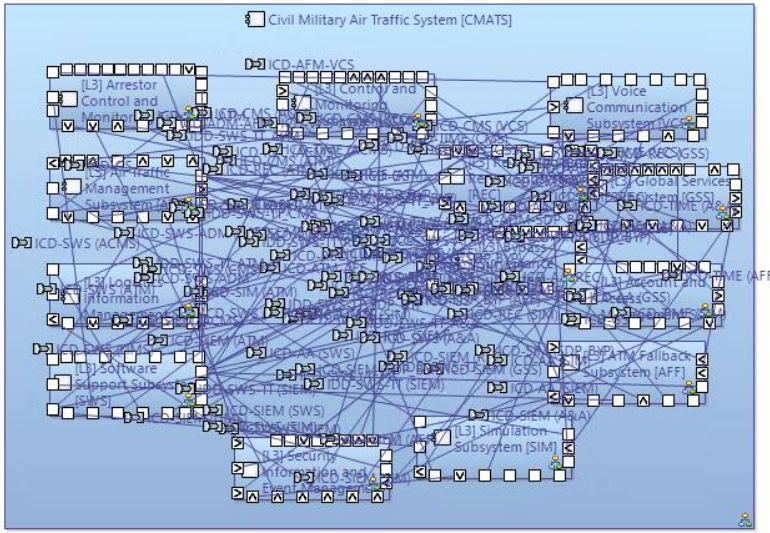
OPEN

THALES
Building a future we can all trust

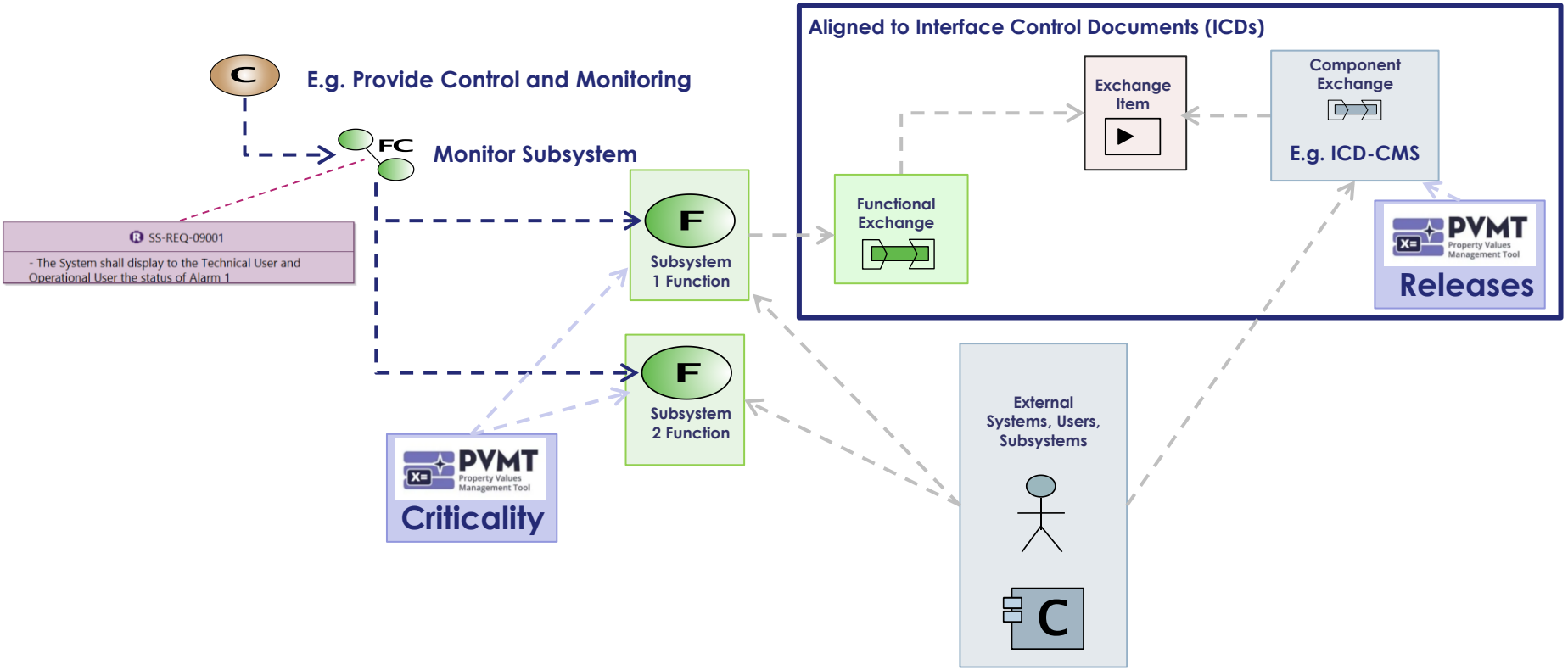
Logical Architecture - What is complete?

Stopping Criteria

- All internal and external interfaces and exchange items modelled and descriptions provided(60+ Interfaces)
- All subsystems modelled as a black box with top level functions only
- All requirements that are fulfilled by more than one subsystem allocated to a functional chain (In Capella) and described.



Logical Architecture Overview

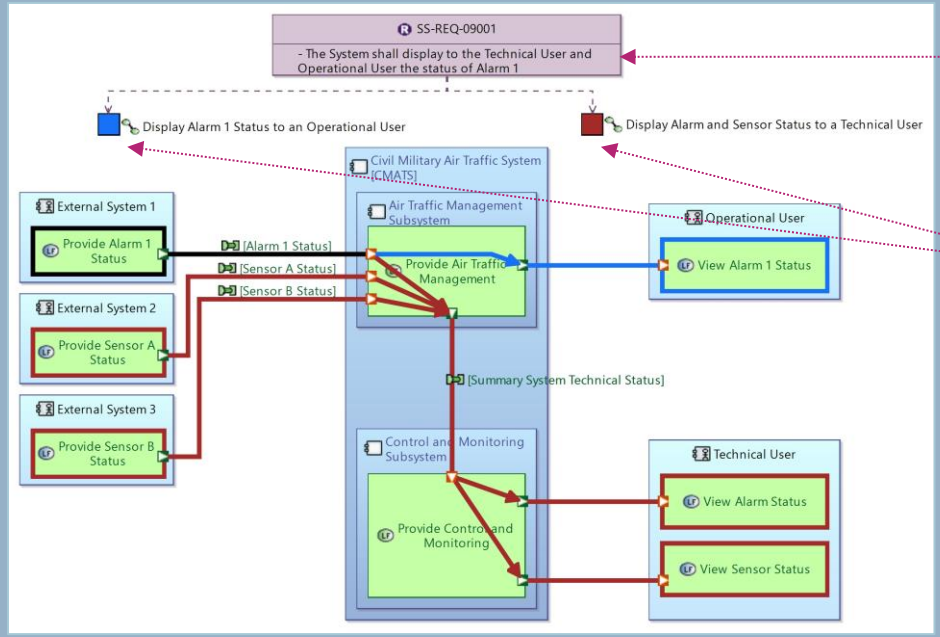


Functional Chains - Clarifying Design

Requirements Management

IE PUID	CMATS Req Title	CMATS System Specification	CMATS System	REQAM Release
SS-REQ-09001	Monitor Alarm 1	The System shall display to the Technical User and Operational User the status of Alarm 1	ATM CMS	Rz R1 R2

CMATS System-of-Systems Team Model



Subsystem allocation in Doors

DOORS Specification
Included into architecture model

Functional Chains
- Demonstrate how the SoS Design satisfies the requirements through defining the role of a subsystem in each requirement
- Highlight Dependencies on Internal and External Interfaces to support System Integration.

OPEN

Outputs – SoS Architecture Reference

Provide targeted views

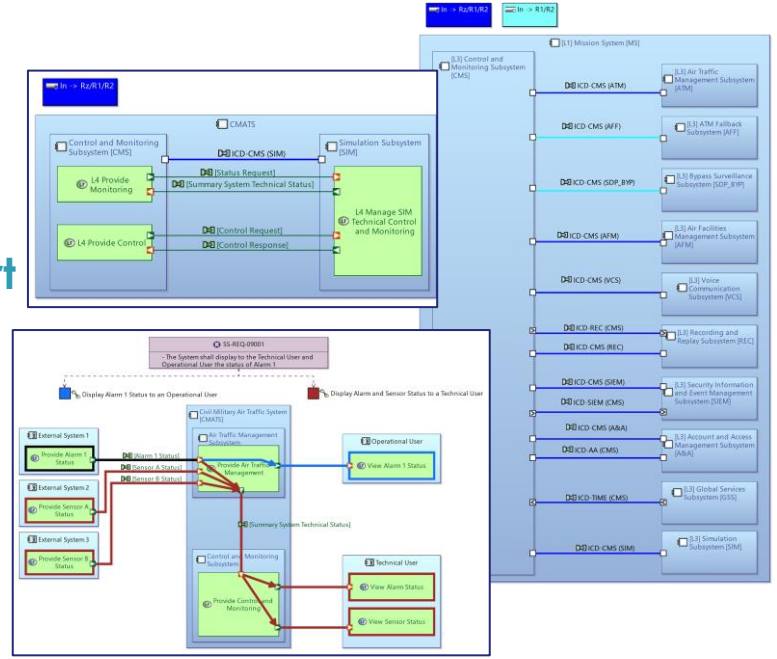
- System Context Diagrams (Release Management)
- High Level Interface Definitions

Provide Defined and Agreed framework to support change management

Support Gateway Design

IVVA – Functional Chains

On-boarding activities - HTML



PLM Reference 0000024497 Revision 004 Revision Date 21.05.2021
Template: 87211168-DOC-GRP-EN-006

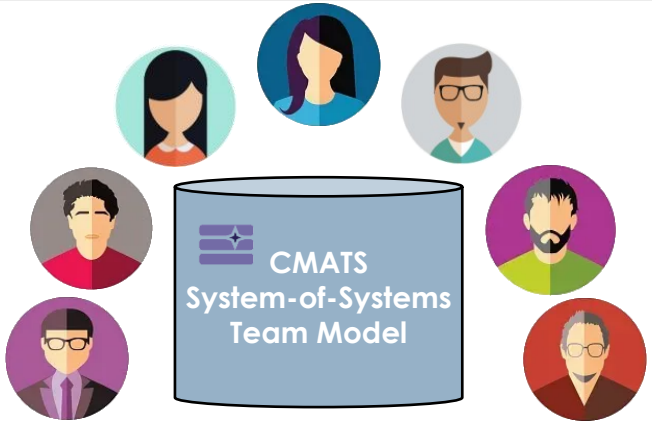
OPEN

Multiple engineers: One team model

Capella Team

Managing Simultaneous design work

- Team Approach
  - Define clear boundaries of responsibility
  - Communication
- Capella Team
  - Validation Rules
  - Commit History
  - Integration with Change Management Tools
  - Daily backups

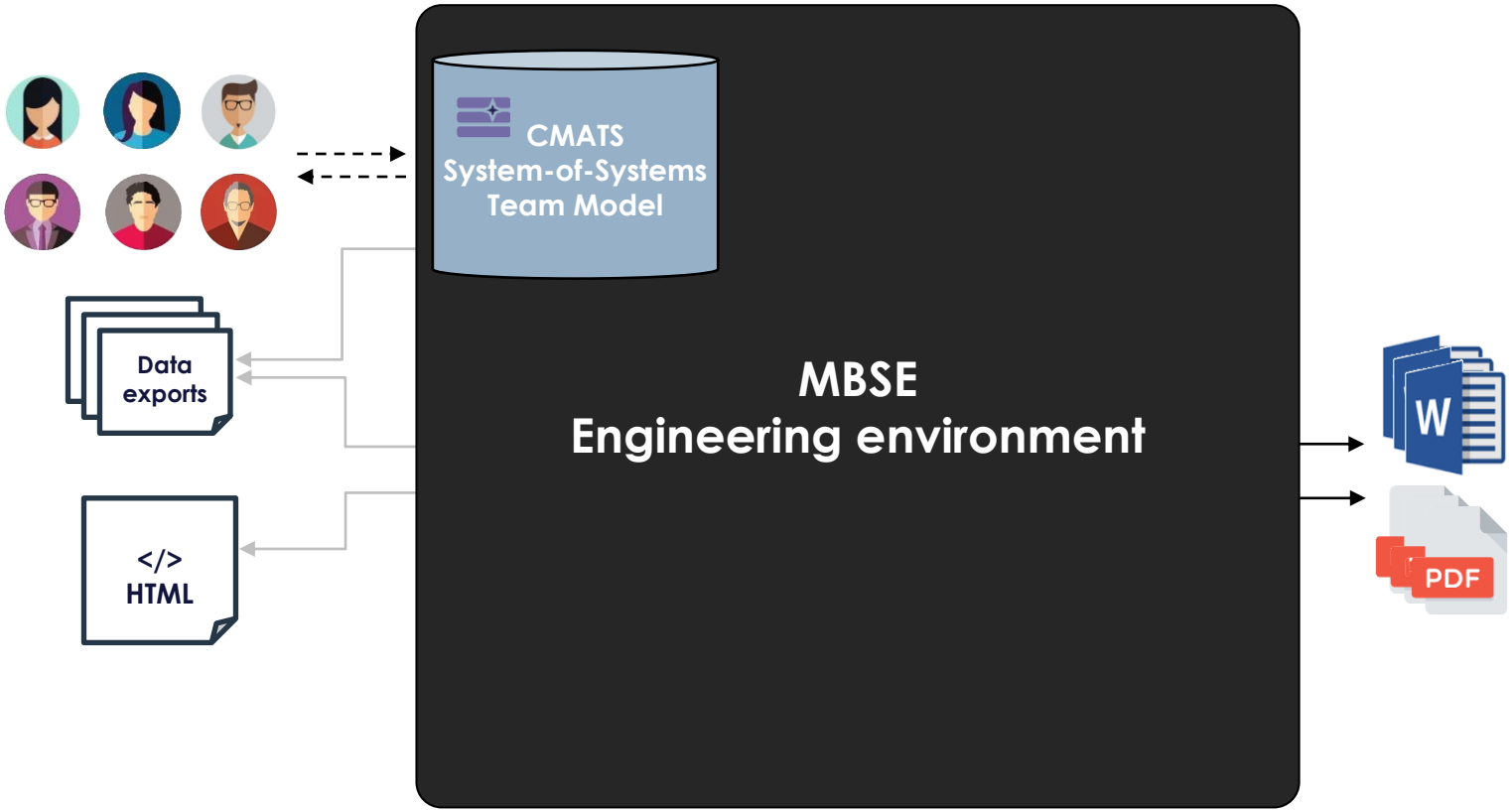


- **10+ Capability definitions**
- **10+ Subsystems**
- **50+ External Interface definitions**
- **10+ Internal Interface definitions**
- **300+ Exchange items**

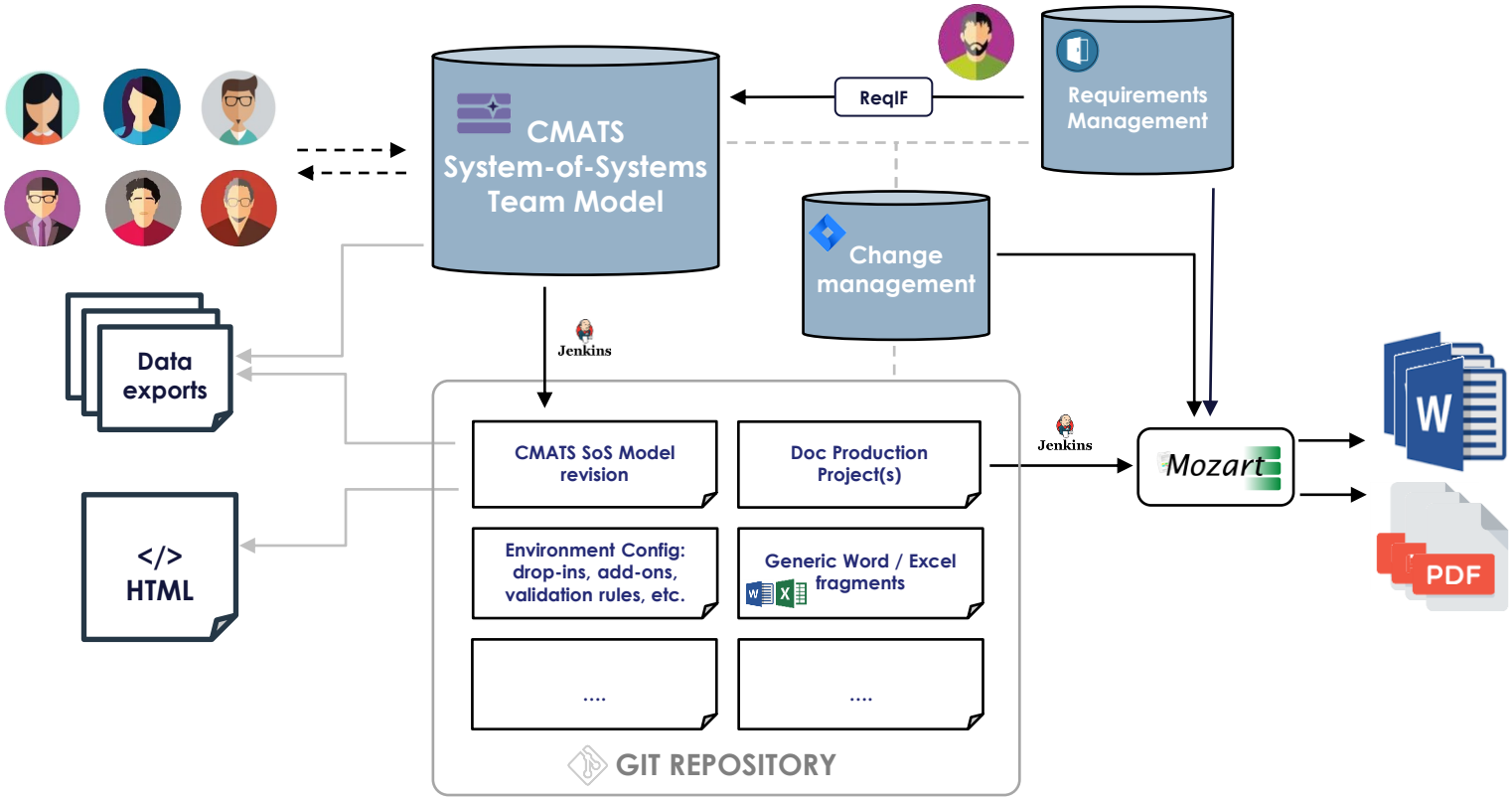
PLM Reference 0000024497 Revision 004 Revision Date 21.05.2021
Template: 87211168-DOC-GRP-EN-006

OPEN

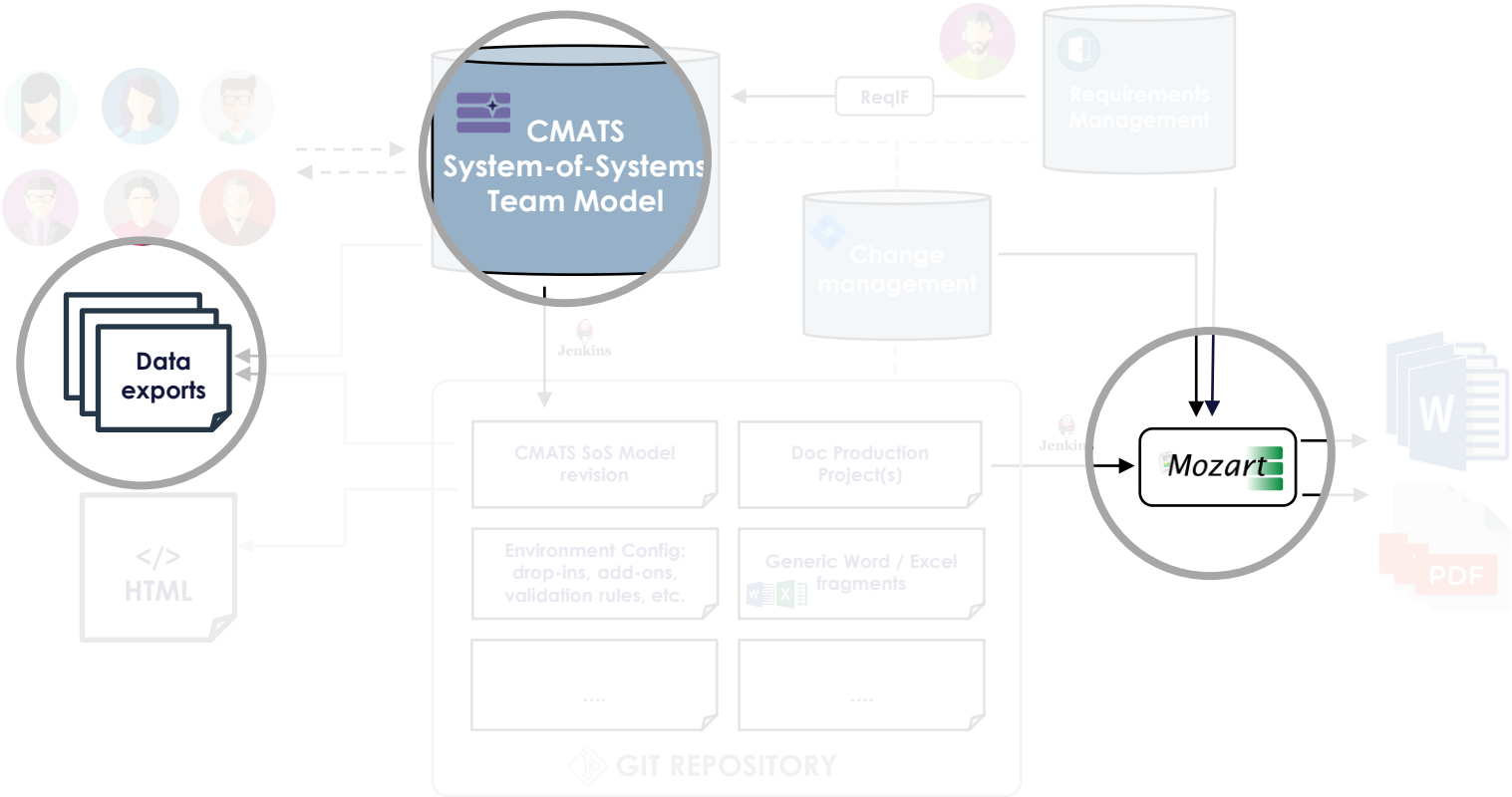
MBSE Engineering Environment introduction



MBSE Engineering Environment detailed overview



MBSE Engineering Environment 3 highlights

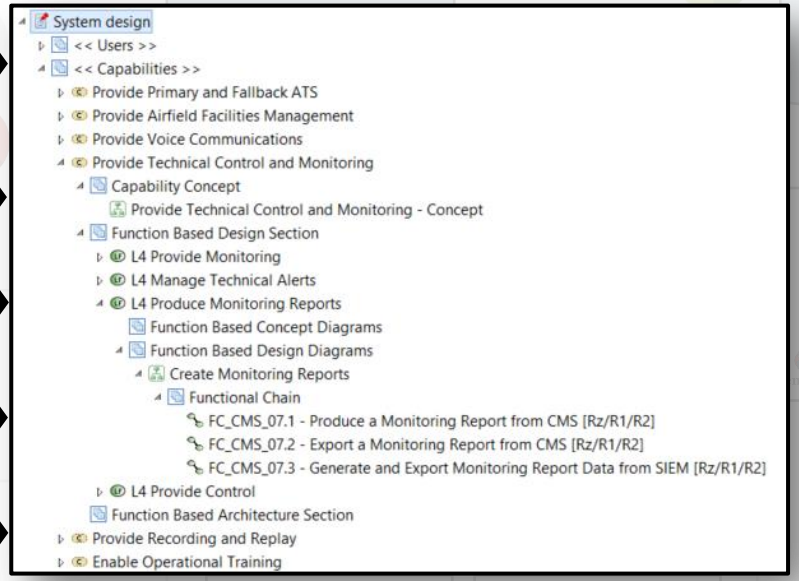


1. Document Production

DOCUMENT INPUTS

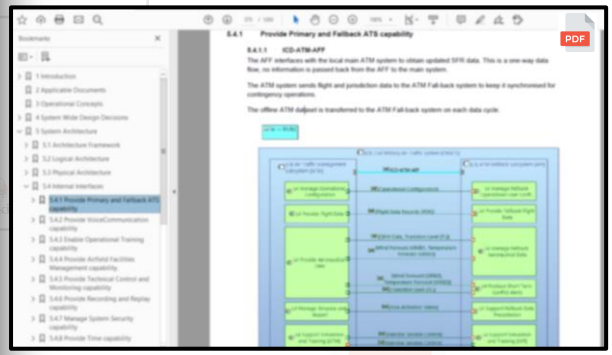
- CMATS SYSTEM-OF-SYSTEM MODEL
- DOCUMENT TEMPLATE (*.DOT)
- REQUIREMENTS DATA
- CHANGE MANAGEMENT DATA
- GENERIC DOC TEXTS /TABLES

DOCUMENT ASSEMBLY



Jenkins

FINAL DOCUMENT



Doc(s) structured using Capella/Arcadia relations:

For example: Capability -> Involved Functions -> Diagrams (using 'Element of Interest')

2. Capella PVMT example

CONFIGURATION:

Name	Type	Default Value
Performance		
Physical Node Attributes		
Releases		
Safety		
Security		
Subsystem Security		
Critical Operations Functions		
Scope	[LOGICAL]	
Rationale	stringProperty	
Critical Operation Function	booleanProperty	false
Subsystem Encryption		
Interface Encryption		
Site Allocation		
Site Release Information		
WP Attributes		
WP Site Allocation		

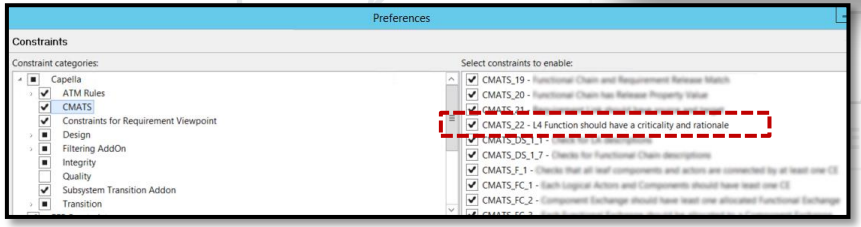
ACTUAL USE:

- SoS_Model_CMATS
  - FBL_glossary.csv
  - SoS_Model_CMATS.afm
  - SoS_Model_CMATS.aird
    - SoS_Model_CMATS
      - OperationalAnalysis
      - System Analysis
      - Logical Architecture
        - Logical Functions
          - Root Logical Function
            - L0 CMATS
              - L1 Provide Mission System Functions [MS]
                - L2 Provide Air Traffic Services [ATS]
                - L3 Provide Air Traffic Management Subsystem Functions [ATM]
                - L4 Provide Flight Data
                - L4 Assist Control Tower
                - L4 Provide Surveillance Data
                - Security.Critical Operations Functions
                - Rationale = An information exchange function that provide controllers
                - Critical Operation Function = TRUE
                - L4 Couple Surveillance and Flight Data
                - Security.Critical Operations Functions
                - Rationale = An efficiency functions that couples surveillance tracks with
                - Critical Operation Function = TRUE
                - L4 Raise Safety Nets Alerts
                - Security.Critical Operations Functions
                - Rationale = A safety function that generates operational alters to operat
                - Critical Operation Function = TRUE

USE IN DOCUMENTS:

L4 Function	Critical Operation Function	Rationale
		[redacted]
[redacted]	True	[redacted]
[redacted]	True	[redacted]
[redacted]	False	[redacted]
[redacted]	True	[redacted]
[redacted]	True	[redacted]

VALIDATION:

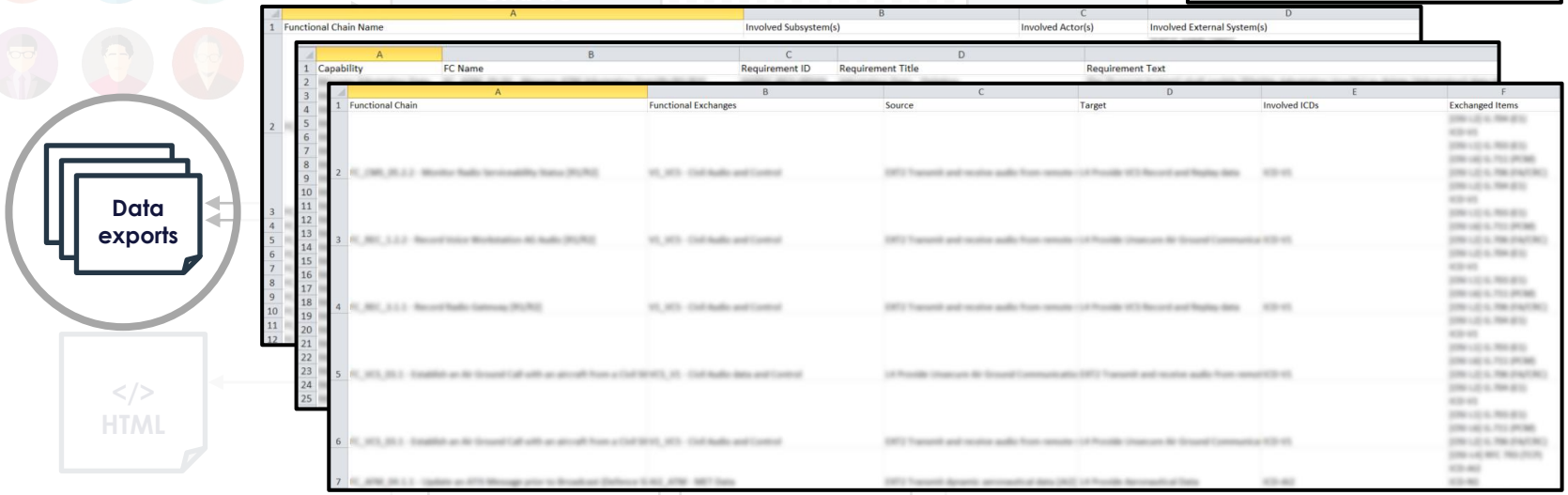


OPEN

3. Model Data Exports

Lightweight Java drop-in adding menu action:

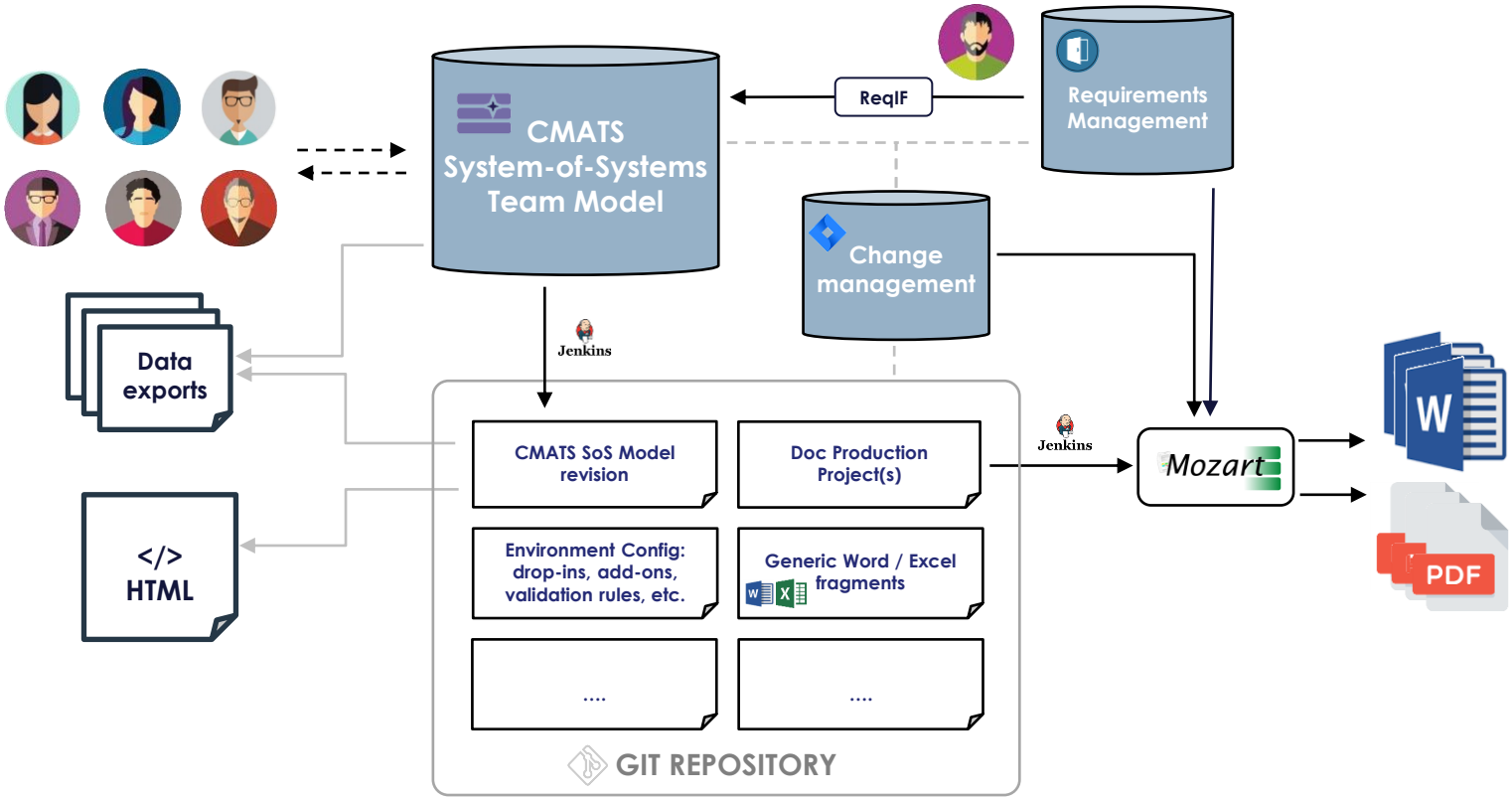
CSV Export Examples:



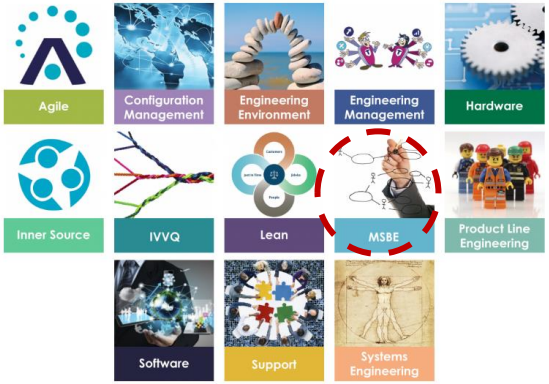
Main usage:

- **Analysis (Architecture Team/Security/Safety/HF)**
- **Integration/Verification/Validation activities**
- **Document delivery (tables)**

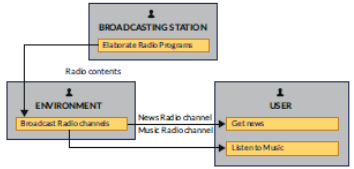
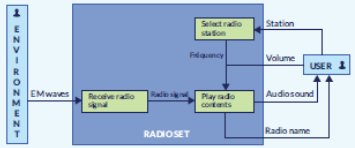
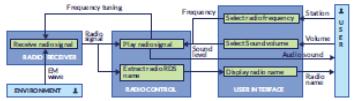
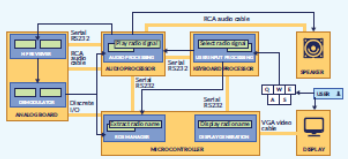
MBSE Engineering Environment overview



Questions?



OPEN

METHOD STEPS	TASKS	SAMPLE MODEL
NEED		
Customer Operational Need Analysis What the users of the system need to accomplish	✓ Define operational capabilities ✓ Perform an operational need analysis	
System/ SW/HW Need Analysis What the system has to accomplish for the Users	✓ Perform a 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 ✓ Design a physical reference architecture ✓ Validate and check it	
Development Contracts What is expected from each designer/ sub-contractor	✓ Define a components IVVQ strategy ✓ Define & enforce a PBS and component integration contract