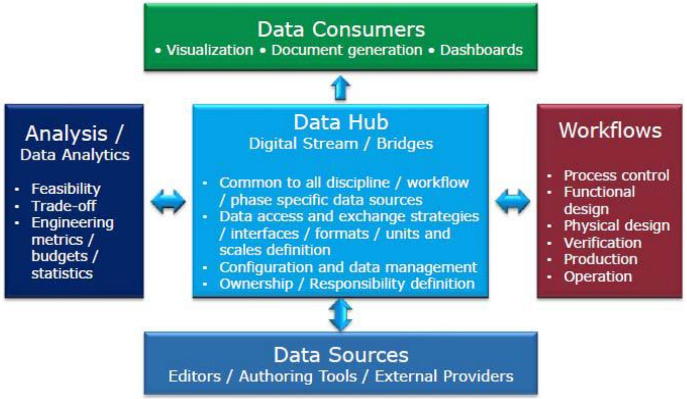
Model Based Engineering Hub

Alex Vorobiev – MBSE 2022 Session 5

Project and Objectives

Objective

To scope the MBSE hub, provide a technical implementation and demonstrate its operation on use-cases involving OPS-ENG and RAMS-ENG exchanges.



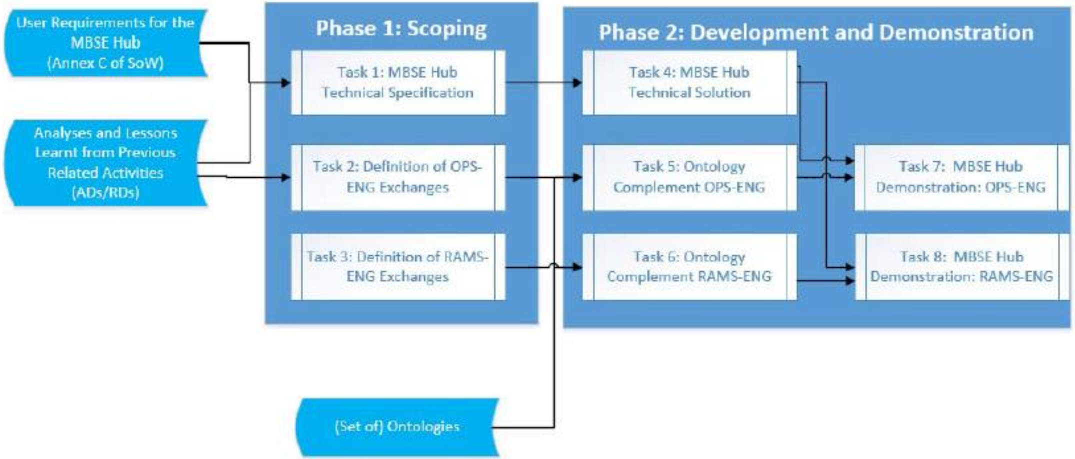
Data Consumers
• Visualization • Document generation • Dashboards

Analysis /
Data Analytics
- Feasibility
- Trade-off
- Engineering metrics / budgets / statistics

Data Hub
Digital Stream / Bridges
- Common to all discipline / workflow / phase specific data sources
- Data access and exchange strategies / interfaces / formats / units and scales definition
- Configuration and data management
- Ownership / Responsibility definition

Workflows
- Process control
- Functional design
- Physical design
- Verification
- Production
- Operation

Data Sources
Editors / Authoring Tools / External Providers



User Requirements for the MBSE Hub (Annex C of SoW)

Analyses and Lessons Learnt from Previous Related Activities (ADs/RDs)

(Set of) Ontologies

Phase 1: Scoping
- Task 1: MBSE Hub Technical Specification
- Task 2: Definition of OPS-ENG Exchanges
- Task 3: Definition of RAMS-ENG Exchanges

Phase 2: Development and Demonstration
- Task 4: MBSE Hub Technical Solution
- Task 5: Ontology Complement OPS-ENG
- Task 6: Ontology Complement RAMS-ENG
- Task 7: MBSE Hub Demonstration: OPS-ENG
- Task 8: MBSE Hub Demonstration: RAMS-ENG

Partnership and Roles

RHEA GROUP

Design and Development of Implementation, Conceptual Modelling

OHB

OPS Experts

ThalesAlenia Space

a Thales / Leonardo company

RAMS Experts

«DigitalEngineering»

DEKonsult

Consultation

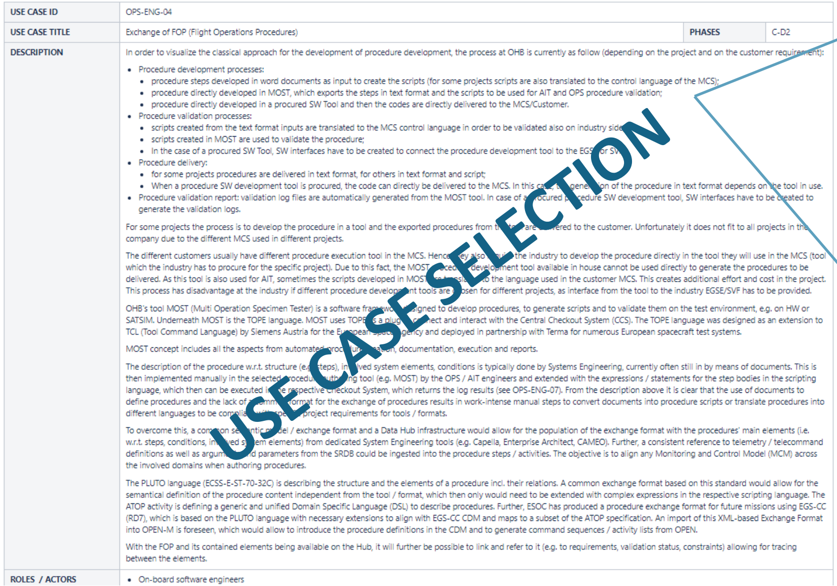
Approach

Definition of Exchanges

- Identification of Actors
- Identification of various use cases for both OPS and RAMS
- Mapping of use cases to processes such as ones defined in ESA-OPS-QMS-GSEG-PR-1000 and ECSS-E-ST-70C.
- Identification of exchanges
- Selection and scoping of a single use case to be used for demonstration
- Detailing the use case

Definition of Exchanges

Exchange of FOP (Flight Operations Procedures)



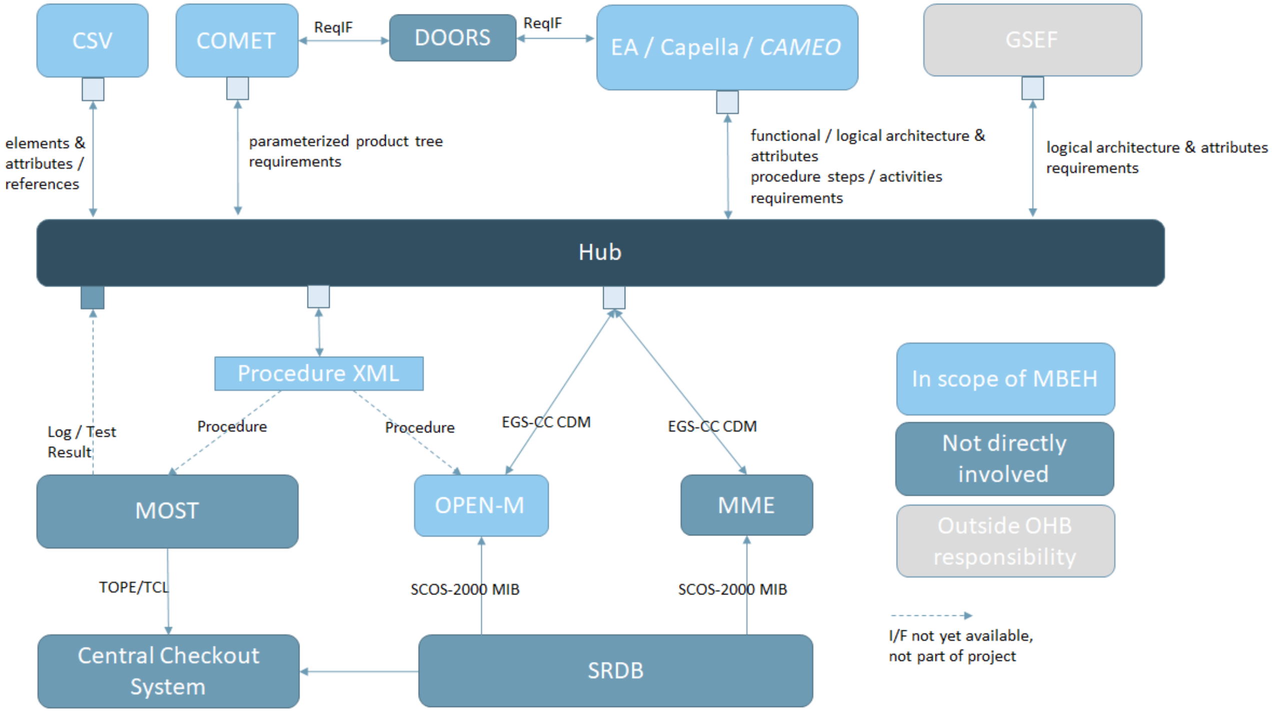
USE CASE ID	OPS-ENG-04		
USE CASE TITLE	Exchange of FOP (Flight Operations Procedures)	**PHASES**	C-D2
DESCRIPTION	In order to visualize the classical approach for the development of procedure development, the process at OHB is currently as follow (depending on the project and on the customer requirement): • Procedure development processes: • procedure steps developed in word documents as input to create the scripts (for some projects scripts are also translated to the control language of the MCS); • procedure directly developed in MOST, which exports the steps in text format and the scripts to be used for AIT and OPS procedure validation; • procedure directly developed in a procured SW Tool and then the codes are directly delivered to the MCS/Customer. • Procedure validation processes: • scripts created from the text format inputs are translated to the MCS control language in order to be validated also on industry side; • scripts created in MOST are used to validate the procedure; • In the case of a procured SW Tool, SW interfaces have to be created to connect the procedure development tool to the EGSE or SVF. • Procedure delivery: • for some projects procedures are delivered in text format, for others in text format and script; • When a procedure SW development tool is procured, the code can directly be delivered to the MCS. In this case, the generation of the procedure in text format depends on the tool in use. • Procedure validation report: validation log files are automatically generated from the MOST tool. In case of a procured procedure SW development tool, SW interfaces have to be created to generate the validation logs. For some projects the process is to develop the procedure in a tool and the exported procedures from the tool are delivered to the customer. Unfortunately it does not fit to all projects in the company due to the different MCS used in different projects. The different customers usually have different procedure execution tool in the MCS. Hence they also require the industry to develop the procedure directly in the tool they will use in the MCS (tool which the industry has to procure for the specific project). Due to this fact, the MOST procedure development tool available in house cannot be used directly to generate the procedures to be delivered. As this tool is also used for AIT, sometimes the scripts developed in MOST are translated to the language used in the customer MCS. This creates additional effort and cost in the project. This process has disadvantage at the industry if different procedure development tools are chosen for different projects, as interface from the tool to the industry EGSE/SVF has to be provided. OHB's tool MOST (Multi Operation Specimen Tester) is a software framework designed to develop procedures, to generate scripts and to validate them on the test environment, e.g. on HW or SATSIM. Underneath MOST is the TOPE language. MOST uses TOPE as a plugin to connect and interact with the Central Checkout System (CCS). The TOPE language was designed as an extension to TCL (Tool Command Language) by Siemens Austria for the European Space Agency and deployed in partnership with Terma for numerous European spacecraft test systems. MOST concept includes all the aspects from automated procedure generation, documentation, execution and reports. The description of the procedure w.r.t. structure (e.g. steps), involved system elements, conditions is typically done by Systems Engineering, currently often still in by means of documents. This is then implemented manually in the selected procedure authoring tool (e.g. MOST) by the OPS / AIT engineers and extended with the expressions / statements for the step bodies in the scripting language, which then can be executed in the respective Checkout System, which returns the log results (see OPS-ENG-07). From the description above it is clear that the use of documents to define procedures and the lack of a common format for the exchange of procedures results in work-intense manual steps to convert documents into procedure scripts or translate procedures into different languages to be compliant with specific project requirements for tools / formats. To overcome this, a common semantic model / exchange format and a Data Hub infrastructure would allow for the population of the exchange format with the procedures' main elements (i.e. w.r.t. steps, conditions, involved system elements) from dedicated System Engineering tools (e.g. Capella, Enterprise Architect, CAMEO). Further, a consistent reference to telemetry / telecommand definitions as well as arguments and parameters from the SRDB could be ingested into the procedure steps / activities. The objective is to align any Monitoring and Control Model (MCM) across the involved domains when authoring procedures. The PLUTO language (ECSS-E-ST-70-32C) is describing the structure and the elements of a procedure incl. their relations. A common exchange format based on this standard would allow for the semantical definition of the procedure content independent from the tool / format, which then only would need to be extended with complex expressions in the respective scripting language. The ATOP activity is defining a generic and unified Domain Specific Language (DSL) to describe procedures. Further, ESOC has produced a procedure exchange format for future missions using EGS-CC (RD7), which is based on the PLUTO language with necessary extensions to align with EGS-CC CDM and maps to a subset of the ATOP specification. An import of this XML-based Exchange Format into OPEN-M is foreseen, which would allow to introduce the procedure definitions in the CDM and to generate command sequences / activity lists from OPEN. With the FOP and its contained elements being available on the Hub, it will further be possible to link and refer to it (e.g. to requirements, validation status, constraints) allowing for tracing between the elements.		
ROLES / ACTORS	• On-board software engineers		

USE CASE SELECTION

OPS: Exchange of Flight Operations Procedures

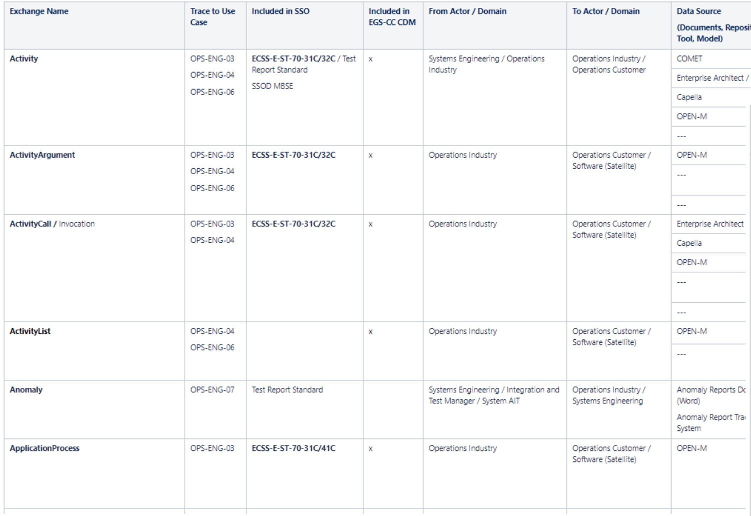
RAMS: Link between FMEA + FDIR + Operational procedures

OPS Demonstration Scenario


CSV
COMET
ReqIF
DOORS
ReqIF
EA / Capella / *CAMEO*
GSEF
elements & attributes / references
parameterized product tree requirements
functional / logical architecture & attributes
procedure steps / activities
requirements
logical architecture & attributes requirements
Hub
Procedure XML
Log / Test Result
Procedure
Procedure
EGS-CC CDM
EGS-CC CDM
In scope of MBEH
Not directly involved
Outside OHB responsibility
MOST
OPEN-M
MME
TOPE/TCL
SCOS-2000 MIB
SCOS-2000 MIB
I/F not yet available, not part of project
Central Checkout System
SRDB

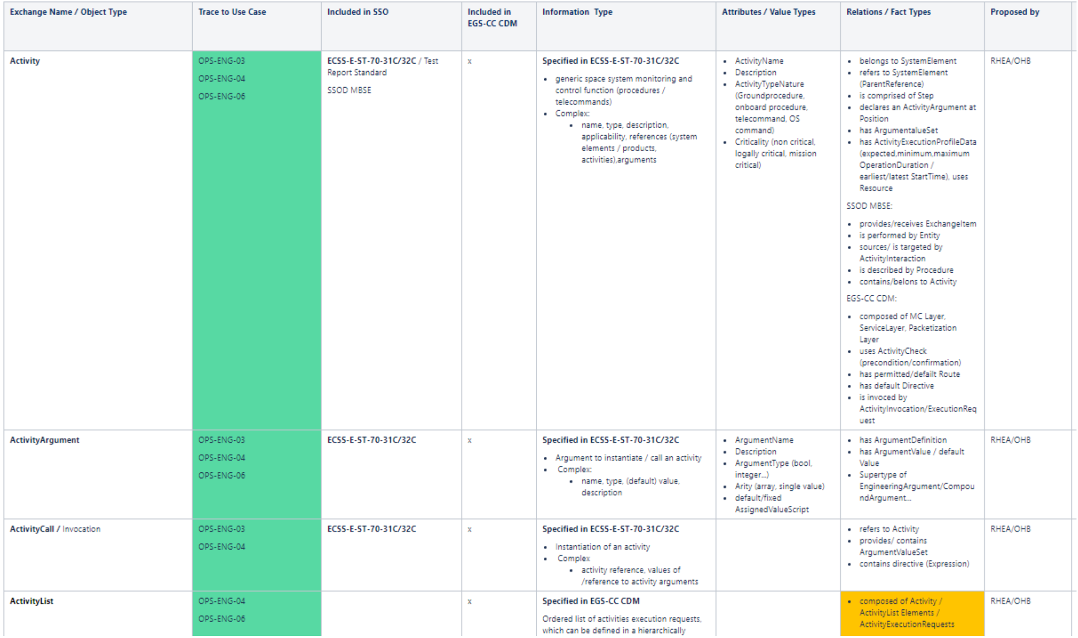
Towards the Ontology Definition

Exchange Identification



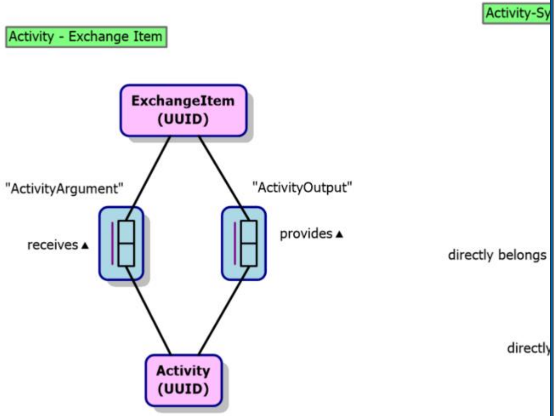
Exchange Name	Trace to Use Case	Included in SSO	Included in EGS-CC CDM	From Actor / Domain	To Actor / Domain	Data Source (Documents, Repository, Tool, Model)	Data Format (Standards & Schemas) / Data model	Information Type
Activity	OPS-ENG-03 OPS-ENG-04 OPS-ENG-06	**ECSS-E-ST-70-31C/32C** / Test Report Standard SSOD MBSE	x	Systems Engineering / Operations Industry	Operations Industry / Operations Customer	COMET Enterprise Architect / CAMEO Capella OPEN-M ---	E-TM-10-25 SysML Capella Meta Model (ECORE)	**Specified in ECSS-E-ST-70-31C/32C** • generic space system monitoring and control function (procedures / telecommands) • Complex:
ActivityArgument	OPS-ENG-03 OPS-ENG-04 OPS-ENG-06	**ECSS-E-ST-70-31C/32C**	x	Operations Industry	Operations Customer / Software (Satellite)	OPEN-M --- ---		
ActivityCall / Invocation	OPS-ENG-03 OPS-ENG-04	**ECSS-E-ST-70-31C/32C**	x	Operations Industry	Operations Customer / Software (Satellite)	Enterprise Architect Capella OPEN-M --- ---		
ActivityList	OPS-ENG-04 OPS-ENG-06		x	Operations Industry	Operations Customer / Software (Satellite)	OPEN-M ---		
Anomaly	OPS-ENG-07	Test Report Standard		Systems Engineering / Integration and Test Manager / System AIT	Operations Industry / Systems Engineering	Anomaly Reports Dc (Word) Anomaly Report Tra System		
ApplicationProcess	OPS-ENG-03	**ECSS-E-ST-70-31C/41C**	x	Operations Industry	Operations Customer / Software (Satellite)	OPEN-M		

Concept Definition



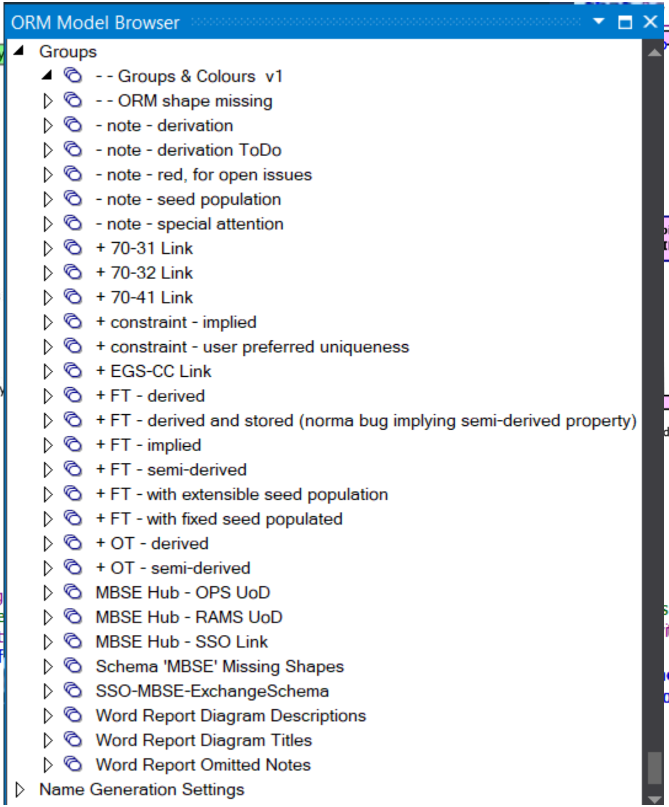
Exchange Name / Object Type	Trace to Use Case	Included in SSO	Included in EGS-CC CDM	Information Type	Attributes / Value Types	Relations / Fact Types	Proposed by
Activity	OPS-ENG-03 OPS-ENG-04 OPS-ENG-06	**ECSS-E-ST-70-31C/32C** / Test Report Standard SSOD MBSE	x	**Specified in ECSS-E-ST-70-31C/32C** • generic space system monitoring and control function (procedures / telecommands) • Complex: • name, type, description, applicability, references (system elements / products, activities),arguments	• ActivityName • Description • ActivityTypeNature (Groundprocedure, onboard procedure, telecommand, OS command) • Criticality (non critical, logally critical, mission critical)	• belongs to SystemElement • refers to SystemElement (ParentReference) • is comprised of Step • declares an ActivityArgument at Position • has ArgumentalueSet • has ActivityExecutionProfileData (expected,minimum,maximum OperationDuration / earliest/latest StartTime), uses Resource SSOD MBSE: • provides/receives ExchangeItem • is performed by Entity • sources/ is targeted by ActivityInteraction • is described by Procedure • contains/belons to Activity EGS-CC CDM: • composed of MC Layer, ServiceLayer, Packetization Layer • uses ActivityCheck (precondition/confirmation) • has permitted/defailt Route • has default Directive • is invoced by ActivityInvocation/ExecutionRequest	RHEA/OHB
ActivityArgument	OPS-ENG-03 OPS-ENG-04 OPS-ENG-06	**ECSS-E-ST-70-31C/32C**	x	**Specified in ECSS-E-ST-70-31C/32C** • Argument to instantiate / call an activity • Complex: • name, type, (default) value, description	• ArgumentName • Description • ArgumentType (bool, integer…) • Arity (array, single value) • default/fixed AssignedValueScript	• has ArgumentDefinition • has ArgumentValue / default Value • Supertype of EngineeringArgument/CompoundArgument…	RHEA/OHB
ActivityCall / Invocation	OPS-ENG-03 OPS-ENG-04	**ECSS-E-ST-70-31C/32C**	x	**Specified in ECSS-E-ST-70-31C/32C** • Instantiation of an activity • Complex • activity reference, values of /reference to activity arguments		• refers to Activity • provides/ contains ArgumentValueSet • contains directive (Expression)	RHEA/OHB
ActivityList	OPS-ENG-04 OPS-ENG-06		x	**Specified in EGS-CC CDM** Ordered list of activities execution requests, which can be defined in a hierarchically		• composed of Activity / ActivityList Elements / ActivityExecutionRequests	RHEA/OHB

Conceptual Modeling



Activity receives ExchangeItem.
It is possible that some Activity receives **more than one** Exchang
and that for some ExchangeItem, **more than one** Activity receive
In each population of Activity receives ExchangeItem, **each** Activit
This association with Activity, ExchangeItem **provides the pref**

Used by derivations:
Activity supports to exchange ExchangeItem in FlowDirection.

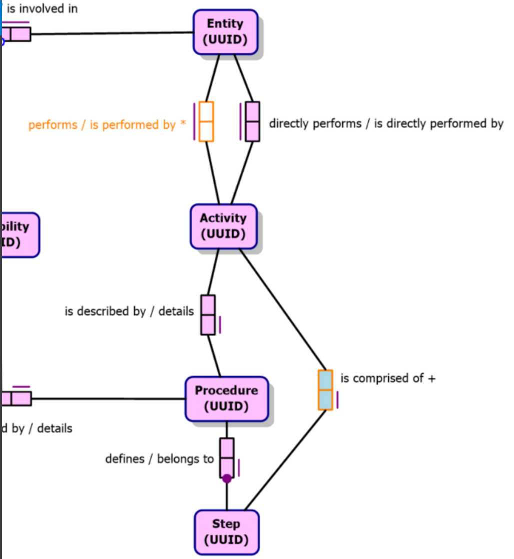


ORM Model Browser

◢ Groups
- ◢ - - Groups & Colours v1
- ▷ - - ORM shape missing
- ▷ - note - derivation
- ▷ - note - derivation ToDo
- ▷ - note - red, for open issues
- ▷ - note - seed population
- ▷ - note - special attention
- ▷ + 70-31 Link
- ▷ + 70-32 Link
- ▷ + 70-41 Link
- ▷ + constraint - implied
- ▷ + constraint - user preferred uniqueness
- ▷ + EGS-CC Link
- ▷ + FT - derived
- ▷ + FT - derived and stored (norma bug implying semi-derived property)
- ▷ + FT - implied
- ▷ + FT - semi-derived
- ▷ + FT - with extensible seed population
- ▷ + FT - with fixed seed populated
- ▷ + OT - derived
- ▷ + OT - semi-derived
- ▷ MBSE Hub - OPS UoD
- ▷ MBSE Hub - RAMS UoD
- ▷ MBSE Hub - SSO Link
- ▷ Schema 'MBSE' Missing Shapes
- ▷ SSO-MBSE-ExchangeSchema
- ▷ Word Report Diagram Descriptions
- ▷ Word Report Diagram Titles
- ▷ Word Report Omitted Notes

▷ Name Generation Settings

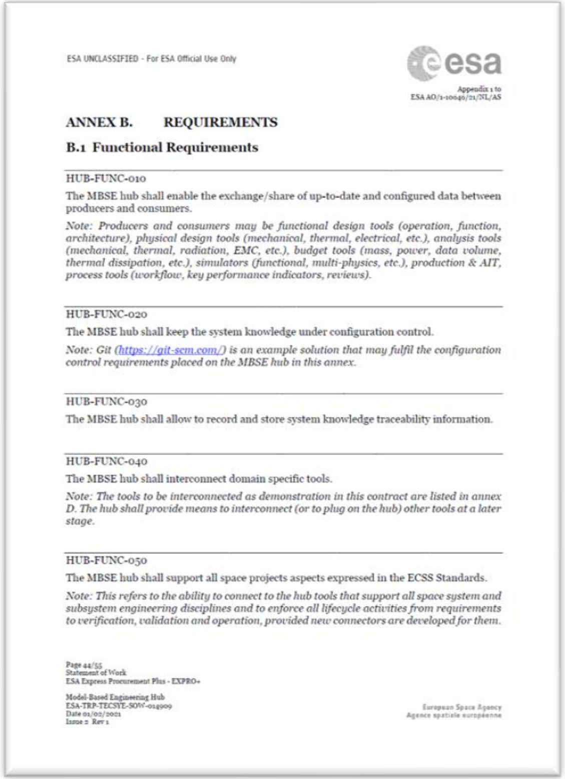
Activity-Procedure-Step



s comprised of Step **if**
ity is described by **some** Procedure **that** defines **that** Step.

ch Step, **at most one** Activity is comprised of **that** Step.
ossible that some Activity is comprised of **more than one** Step.

Specification



ESA UNCLASSIFIED - For ESA Official Use Only

Appendix 1 to
ESA AO/1-10640/21/NL/AS

ANNEX B. REQUIREMENTS

B.1 Functional Requirements

HUB-FUNC-010

The MBSE hub shall enable the exchange/share of up-to-date and configured data between producers and consumers.

Note: Producers and consumers may be functional design tools (operation, function, architecture), physical design tools (mechanical, thermal, electrical, etc.), analysis tools (mechanical, thermal, radiation, EMC, etc.), budget tools (mass, power, data volume, thermal dissipation, etc.), simulators (functional, multi-physics, etc.), production & AIT, process tools (workflow, key performance indicators, reviews).

HUB-FUNC-020

The MBSE hub shall keep the system knowledge under configuration control.

Note: Git (https://git-scm.com/) is an example solution that may fulfil the configuration control requirements placed on the MBSE hub in this annex.

HUB-FUNC-030

The MBSE hub shall allow to record and store system knowledge traceability information.

HUB-FUNC-040

The MBSE hub shall interconnect domain specific tools.

Note: The tools to be interconnected as demonstration in this contract are listed in annex D. The hub shall provide means to interconnect (or to plug on the hub) other tools at a later stage.

HUB-FUNC-050

The MBSE hub shall support all space projects aspects expressed in the ECSS Standards.

Note: This refers to the ability to connect to the hub tools that support all space system and subsystem engineering disciplines and to enforce all lifecycle activities from requirements to verification, validation and operation, provided new connectors are developed for them.

Page 44/55
Statement of Work
ESA Express Procurement Plus - EXPRO+

Model-Based Engineering Hub
ESA-TRP-TECSYE-SOW-014909
Date 01/02/2021
Issue 2 Rev 1

European Space Agency
Agence spatiale européenne

<- User Requirements from Statement of Work

- Multiple discussions on feasibility and scoping
- Requirements on such things as:
  - Configuration management
  - Data structures (i.e., datasets, ontology)
  - Governance

Specification and Architecture of System Factory (SASyF, executed by GMV and LSIs)

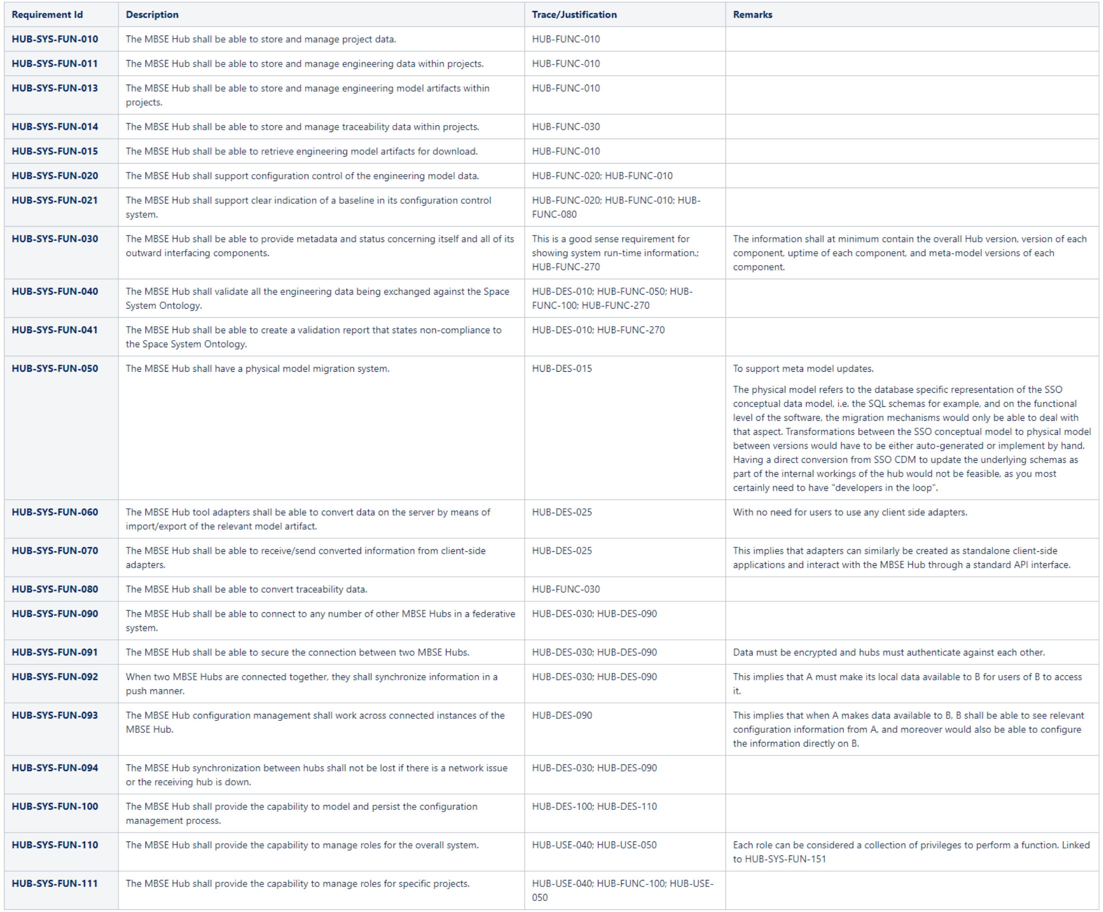
- Requirements for the System Factory
- Logical and Physical Architectures

- Alignment of produced artifacts

Software Requirements

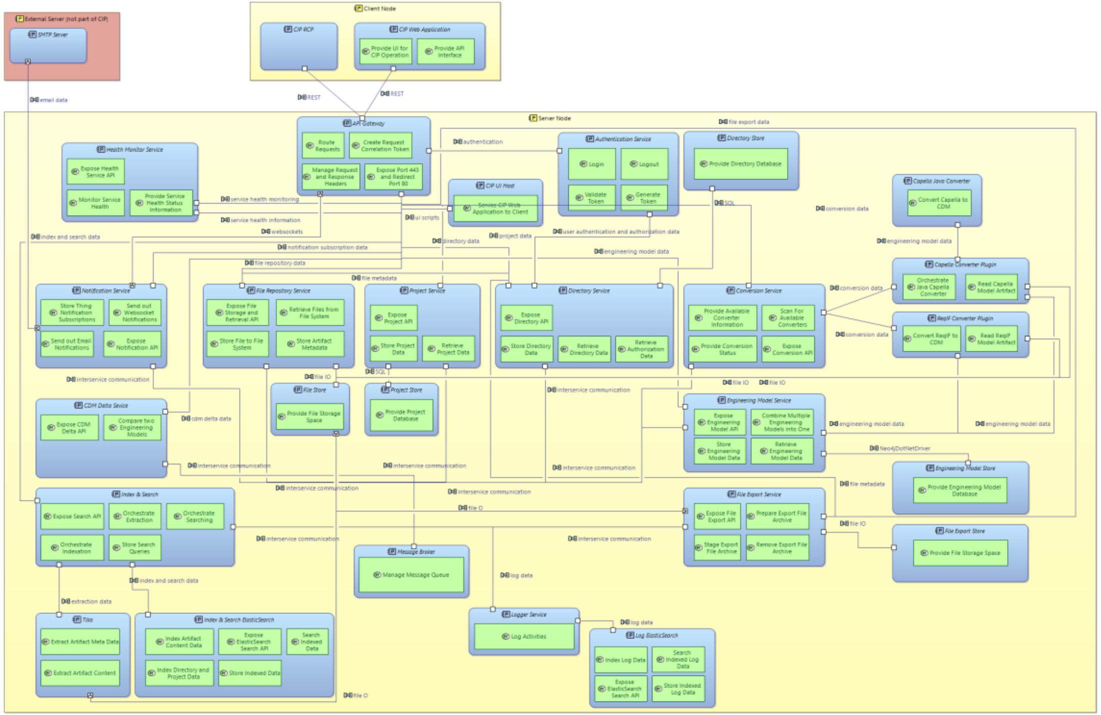
Functional: 49
Performance: 4
Interface: 5
Design: 12
Security: 21
Quality: 7
Maintainability: 3
Configuration: 19
Data: 15
Validation: 2

Total: 137 requirements



Requirement Id	Description	Trace/Justification	Remarks
HUB-SYS-FUN-010	The MBSE Hub shall be able to store and manage project data.	HUB-FUNC-010	
HUB-SYS-FUN-011	The MBSE Hub shall be able to store and manage engineering data within projects.	HUB-FUNC-010	
HUB-SYS-FUN-013	The MBSE Hub shall be able to store and manage engineering model artifacts within projects.	HUB-FUNC-010	
HUB-SYS-FUN-014	The MBSE Hub shall be able to store and manage traceability data within projects.	HUB-FUNC-030	
HUB-SYS-FUN-015	The MBSE Hub shall be able to retrieve engineering model artifacts for download.	HUB-FUNC-010	
HUB-SYS-FUN-020	The MBSE Hub shall support configuration control of the engineering model data.	HUB-FUNC-020; HUB-FUNC-010	
HUB-SYS-FUN-021	The MBSE Hub shall support clear indication of a baseline in its configuration control system.	HUB-FUNC-020; HUB-FUNC-010; HUB-FUNC-080	
HUB-SYS-FUN-030	The MBSE Hub shall be able to provide metadata and status concerning itself and all of its outward interfacing components.	This is a good sense requirement for showing system run-time information.; HUB-FUNC-270	The information shall at minimum contain the overall Hub version, version of each component, uptime of each component, and meta-model versions of each component.
HUB-SYS-FUN-040	The MBSE Hub shall validate all the engineering data being exchanged against the Space System Ontology.	HUB-DES-010; HUB-FUNC-050; HUB-FUNC-100; HUB-FUNC-270	
HUB-SYS-FUN-041	The MBSE Hub shall be able to create a validation report that states non-compliance to the Space System Ontology.	HUB-DES-010; HUB-FUNC-270	
HUB-SYS-FUN-050	The MBSE Hub shall have a physical model migration system.	HUB-DES-015	To support meta model updates. The physical model refers to the database specific representation of the SSO conceptual data model, i.e. the SQL schemas for example, and on the functional level of the software, the migration mechanisms would only be able to deal with that aspect. Transformations between the SSO conceptual model to physical model between versions would have to be either auto-generated or implement by hand. Having a direct conversion from SSO CDM to update the underlying schemas as part of the internal workings of the hub would not be feasible, as you most certainly need to have "developers in the loop".
HUB-SYS-FUN-060	The MBSE Hub tool adapters shall be able to convert data on the server by means of import/export of the relevant model artifact.	HUB-DES-025	With no need for users to use any client side adapters.
HUB-SYS-FUN-070	The MBSE Hub shall be able to receive/send converted information from client-side adapters.	HUB-DES-025	This implies that adapters can similarly be created as standalone client-side applications and interact with the MBSE Hub through a standard API interface.
HUB-SYS-FUN-080	The MBSE Hub shall be able to convert traceability data.	HUB-FUNC-030	
HUB-SYS-FUN-090	The MBSE Hub shall be able to connect to any number of other MBSE Hubs in a federative system.	HUB-DES-030; HUB-DES-090	
HUB-SYS-FUN-091	The MBSE Hub shall be able to secure the connection between two MBSE Hubs.	HUB-DES-030; HUB-DES-090	Data must be encrypted and hubs must authenticate against each other.
HUB-SYS-FUN-092	When two MBSE Hubs are connected together, they shall synchronize information in a push manner.	HUB-DES-030; HUB-DES-090	This implies that A must make its local data available to B for users of B to access it.
HUB-SYS-FUN-093	The MBSE Hub configuration management shall work across connected instances of the MBSE Hub.	HUB-DES-090	This implies that when A makes data available to B, B shall be able to see relevant configuration information from A, and moreover would also be able to configure the information directly on B.
HUB-SYS-FUN-094	The MBSE Hub synchronization between hubs shall not be lost if there is a network issue or the receiving hub is down.	HUB-DES-030; HUB-DES-090	
HUB-SYS-FUN-100	The MBSE Hub shall provide the capability to model and persist the configuration management process.	HUB-DES-100; HUB-DES-110	
HUB-SYS-FUN-110	The MBSE Hub shall provide the capability to manage roles for the overall system.	HUB-USE-040; HUB-USE-050	Each role can be considered a collection of privileges to perform a function. Linked to HUB-SYS-FUN-151
HUB-SYS-FUN-111	The MBSE Hub shall provide the capability to manage roles for specific projects.	HUB-USE-040; HUB-FUNC-100; HUB-USE-050	

Common Information Platform

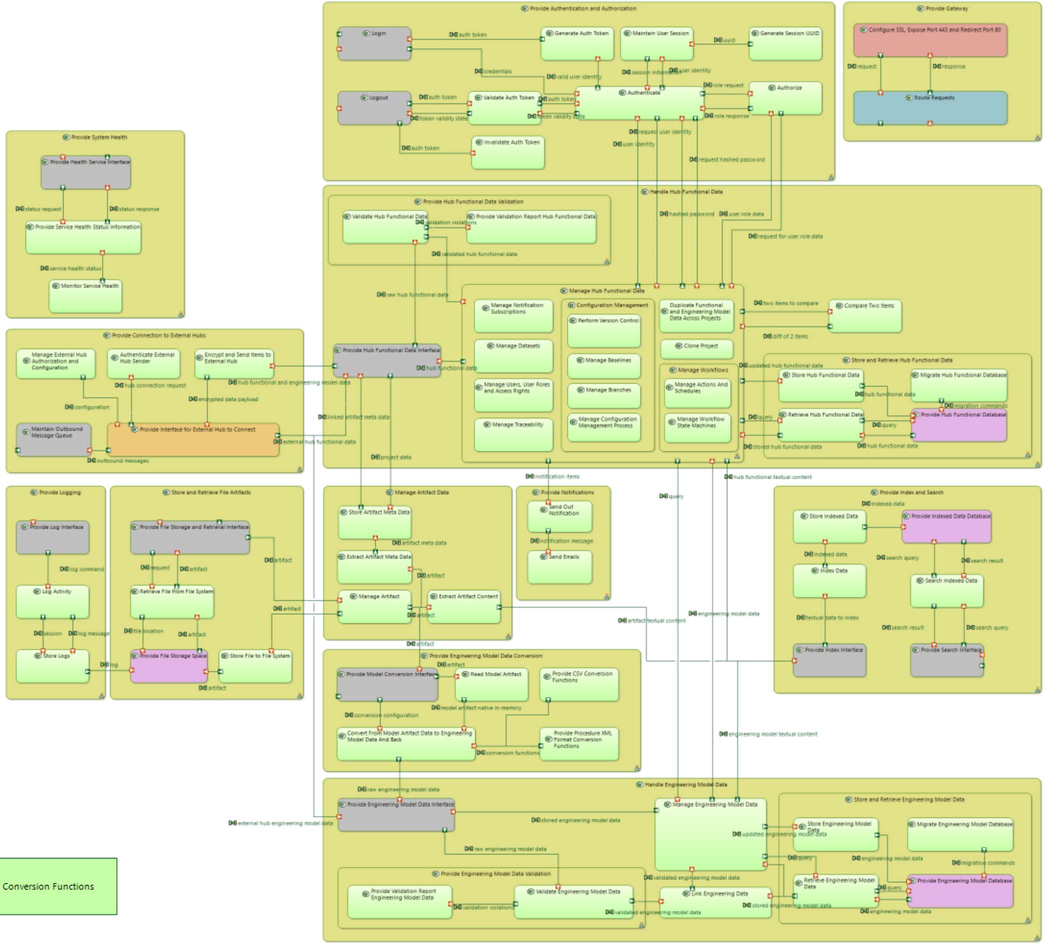

External Server (not part of CIP)
SMTP Server
Client Node
CIP RCP
CIP Web Application
Provide UI for CIP Operation
Provide API Interface
email data
REST
REST
Server Node
API Gateway
Route Requests
Create Request Correlation Token
Manage Request and Response Headers
Expose Port 443 and Redirect Port 80
authentication
Authentication Service
Login
Logout
Validate Token
Generate Token
Directory Store
Provide Directory Database
file export data
CIP UI Host
Service CIP Web Application to Client
Health Monitor Service
Expose Health Service API
Monitor Service Health
Provide Service Health Status Information
service health monitoring
service health information
websockets
ui scripts
SQL
conversion data
Capella Java Converter
Convert Capella to CDM
index and search data
notification subscription data
directory data
project data
user authentication and authorization data
engineering model data
engineering model data
file repository data
file metadata
Notification Service
Store Thing Notification Subscriptions
Send out Websocket Notifications
Send out Email Notifications
Expose Notification API
File Repository Service
Expose File Storage and Retrieval API
Retrieve Files from File System
Store File to File System
Store Artifact Metadata
Project Service
Expose Project API
Store Project Data
Retrieve Project Data
Directory Service
Expose Directory API
Store Directory Data
Retrieve Directory Data
Retrieve Authorization Data
Conversion Service
Provide Available Converter Information
Scan For Available Converters
Provide Conversion Status
Expose Conversion API
conversion data
Capella Converter Plugin
Orchestrate Java Capella Converter
Read Capella Model Artifact
conversion data
ReqIF Converter Plugin
Convert ReqIF to CDM
Read ReqIF Model Artifact
interservice communication
file IO
SQL
file IO
file IO
interservice communication
File Store
Provide File Storage Space
Project Store
Provide Project Database
CDM Delta Sevice
Expose CDM Delta API
Compare two Engineering Models
cdm delta data
Engineering Model Service
Expose Engineering Model API
Combine Multiple Engineering Models into One
Store Engineering Model Data
Retrieve Engineering Model Data
engineering model data
engineering model data
Neo4jDotNetDriver
Engineering Model Store
Provide Engineering Model Database
interservice communication
interservice communication
interservice communication
interservice communication
interservice communication
file metadata
file O
Index &amp; Search
Expose Search API
Orchestrate Extraction
Orchestrate Searching
Orchestrate Indexation
Store Search Queries
File Export Service
Expose File Export API
Prepare Export File Archive
Stage Export File Archive
Remove Export File Archive
file IO
File Export Store
Provide File Storage Space
interservice communication
interservice communication
Message Broker
Manage Message Queue
log data
index and search data
extraction data
Tika
Extract Artifact Meta Data
Extract Artifact Content
Index &amp; Search ElasticSearch
Index Artifact Content Data
Expose ElasticSearch Search API
Search Indexed Data
Index Directory and Project Data
Store Indexed Data
file O
Logger Service
Log Activities
log data
Log ElasticSearch
Index Log Data
Search Indexed Log Data
Expose ElasticSearch Search API
Store Indexed Log Data

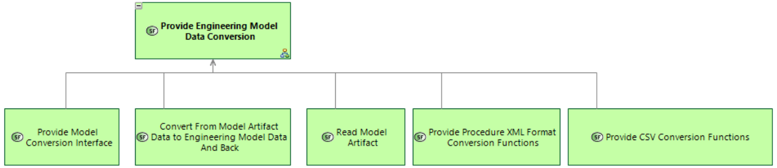
CIP

Common Information Platform

MBSE Hub Architecture

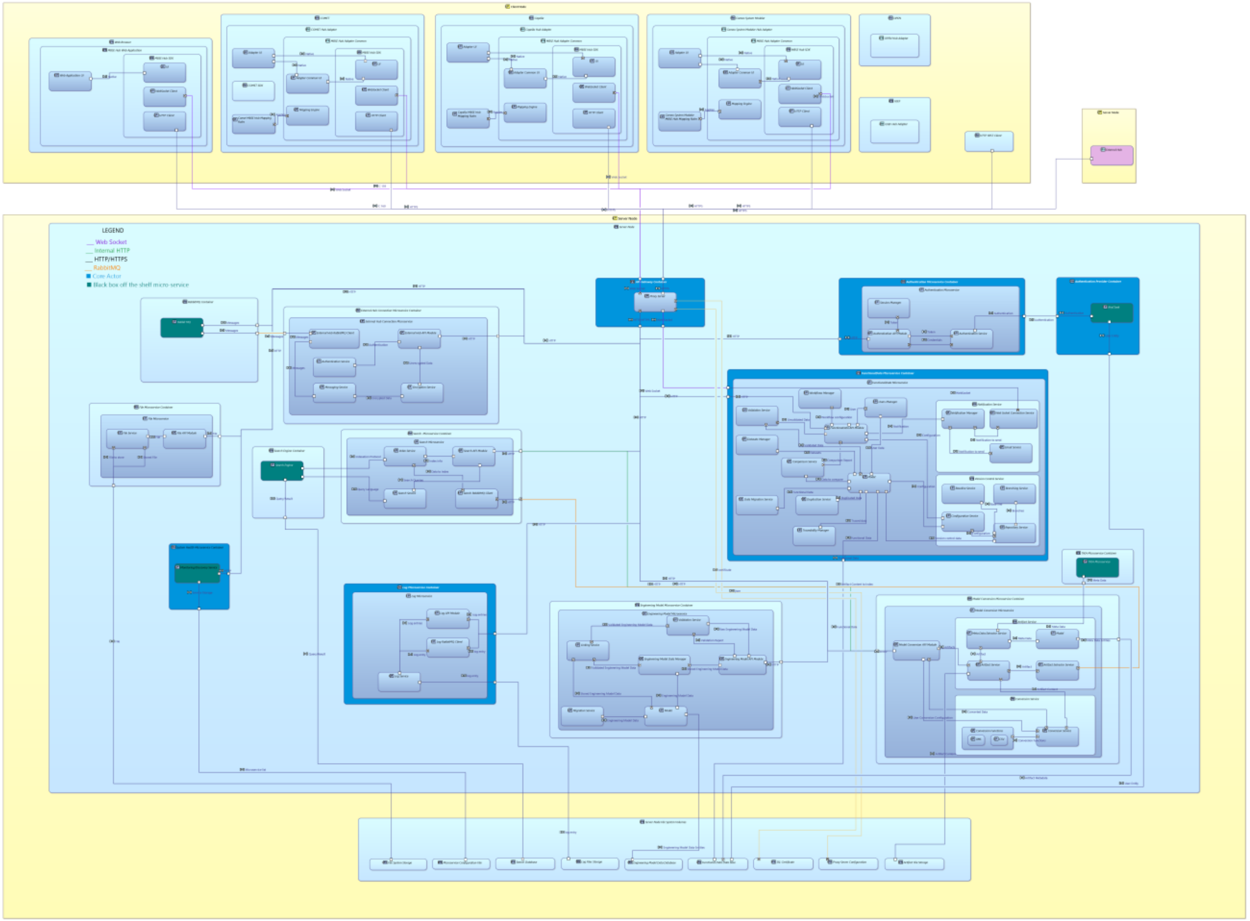
- Split into modules that eventually become microservices just as in the CIP.


Provide Authentication and Authorization
Login
auth token
Generate Auth Token
Maintain User Session
uuid
Generate Session UUID
credentials
valid user identity
session information
user identity
role request
Authorize
Logout
auth token
Validate Auth Token
auth token
Authenticate
token validity state
token validity state
role response
request user identity
user identity
auth token
Invalidate Auth Token
request hashed password
Provide Gateway
Configure SSL, Expose Port 443 and Redirect Port 80
request
response
Route Requests
Provide System Health
Provide Health Service Interface
status request
status response
Provide Service Health Status Information
service health status
Monitor Service Health
Provide Connection to External Hubs
Manage External Hub Authorization and Configuration
Authenticate External Hub Sender
Encrypt and Send Items to External Hub
hub connection request
hub functional and engineering model data
encrypted data payload
configuration
Maintain Outbound Message Queue
Provide Interface for External Hub to Connect
external hub functional data
outbound messages
linked artifact meta data
Handle Hub Functional Data
Provide Hub Functional Data Validation
Validate Hub Functional Data
validation violations
Provide Validation Report Hub Functional Data
validated hub functional data
hashed password
user role data
request for user role data
raw hub functional data
Provide Hub Functional Data Interface
hub functional data
project data
Manage Hub Functional Data
Manage Notification Subscriptions
Manage Datasets
Manage Users, User Roles and Access Rights
Manage Traceability
Configuration Management
Perform Version Control
Manage Baselines
Manage Branches
Manage Configuration Management Process
Duplicate Functional and Engineering Model Data Across Projects
Clone Project
Manage Workflows
Manage Actions And Schedules
Manage Workflow State Machines
two items to compare
Compare Two Items
diff of 2 items
Store and Retrieve Hub Functional Data
updated hub functional data
Store Hub Functional Data
Migrate Hub Functional Database
hub functional data
migration commands
query
Retrieve Hub Functional Data
Provide Hub Functional Database
query
stored hub functional data
hub functional data
notification items
hub functional textual content
query
Provide Logging
Provide Log Interface
log command
Log Activity
session
log message
Store Logs
log
Store and Retrieve File Artifacts
Provide File Storage and Retrieval Interface
request
artifact
artifact
Retrieve File from File System
artifact
file location
artifact
Provide File Storage Space
Store File to File System
artifact
Manage Artifact Data
Store Artifact Meta Data
artifact meta data
Extract Artifact Meta Data
artifact
Manage Artifact
Extract Artifact Content
artifact
artifact
Provide Notifications
Send Out Notification
notification message
Send Emails
artifact textual content
engineering model data
Provide Index and Search
indexed data
Store Indexed Data
Provide Indexed Data Database
indexed data
search query
search result
Index Data
Search Indexed Data
textual data to index
search result
search query
Provide Index Interface
Provide Search Interface
Provide Engineering Model Data Conversion
artifact
Provide Model Conversion Interface
Read Model Artifact
Provide CSV Conversion Functions
model artifact native in-memory
conversion configuration
Convert From Model Artifact Data to Engineering Model Data And Back
Provide Procedure XML Format Conversion Functions
conversion functions
engineering model textual content
Handle Engineering Model Data
raw engineering model data
Provide Engineering Model Data Interface
external hub engineering model data
stored engineering model data
Manage Engineering Model Data
raw engineering model data
Store and Retrieve Engineering Model Data
Store Engineering Model Data
Migrate Engineering Model Database
updated engineering model data
query
engineering model data
migration commands
Provide Engineering Model Data Validation
Provide Validation Report Engineering Model Data
validation violations
Validate Engineering Model Data
validated engineering model data
validated engineering model data
Link Engineering Data
stored engineering model data
Retrieve Engineering Model Data
query
Provide Engineering Model Database
engineering model data

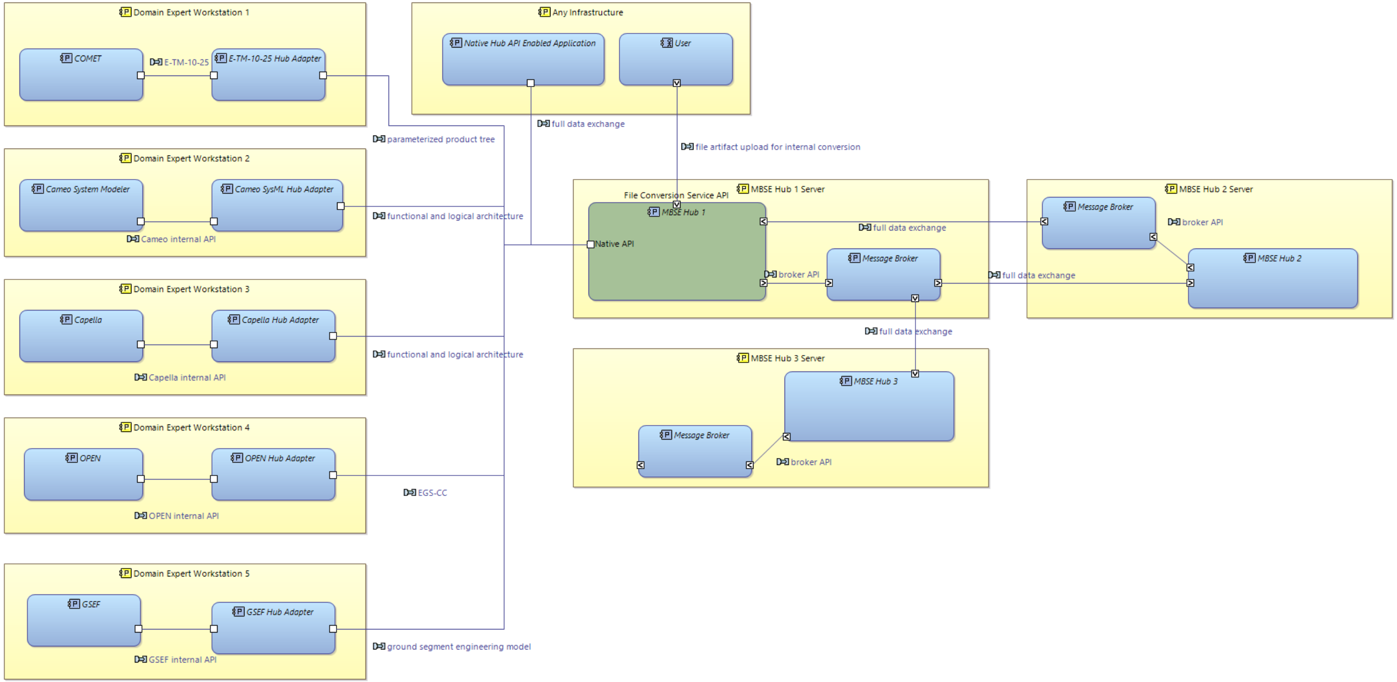

Provide Engineering Model Data Conversion
Provide Model Conversion Interface
Convert From Model Artifact Data to Engineering Model Data And Back
Read Model Artifact
Provide Procedure XML Format Conversion Functions
Provide CSV Conversion Functions

Physical Architecture

- Microservice Based
- Kubernetes deployable

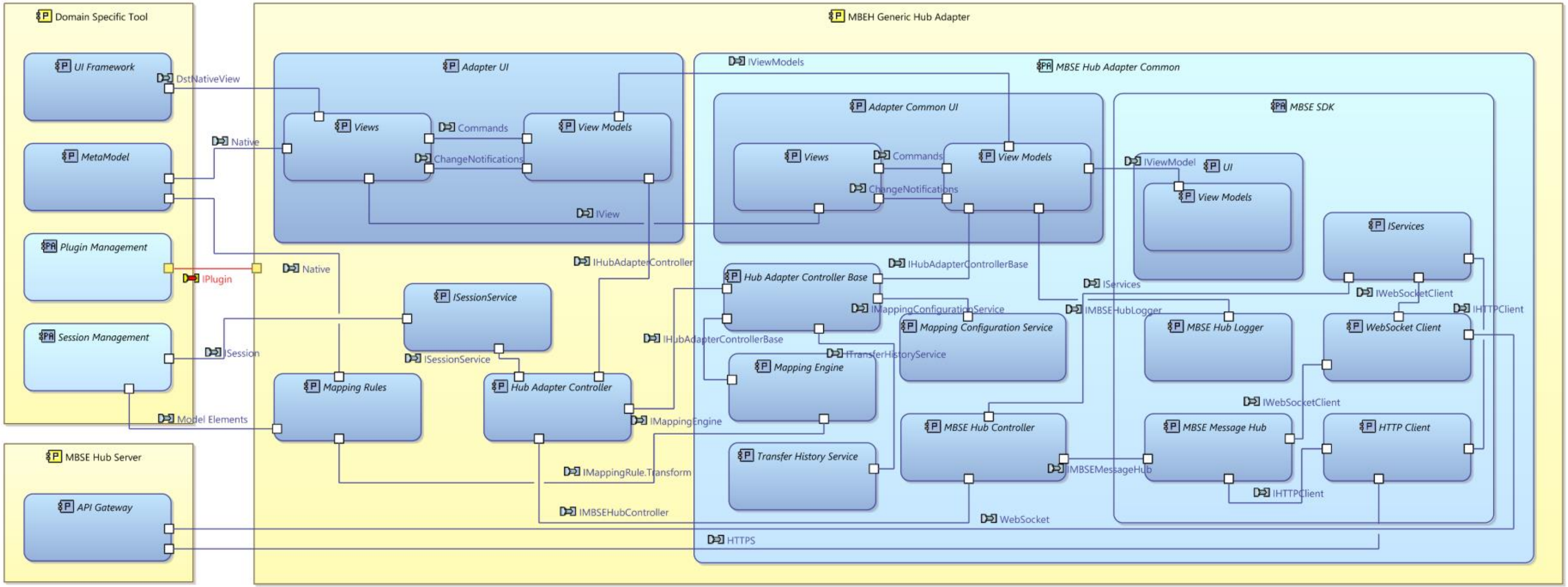


Domain Specific Tools and Hub to Hub Interface



Adapters to Domain Specific Tools

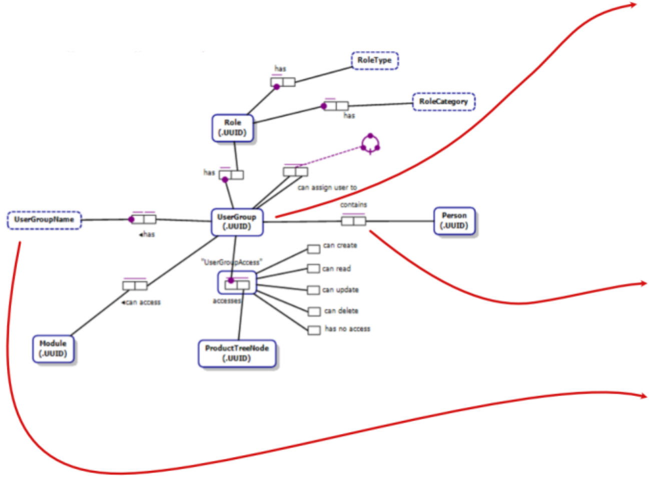
Legacy of Digital Engineering Hub Pathfinder

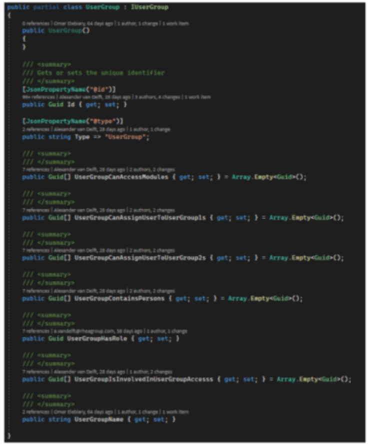


Ontology to Code

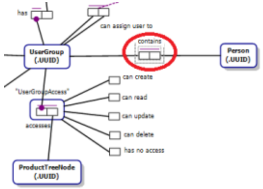
Kalliope

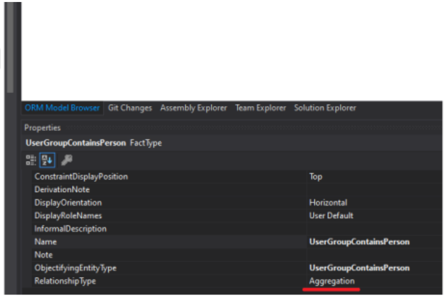
Object Role Modelling





```
public partial class UserGroup : IUserGroup
{
    0 references | Omar Elebiary, 64 days ago | 1 author, 1 change | 1 work item
    public UserGroup()
    {
    }

    /// <summary>
    /// Gets or sets the unique identifier
    /// </summary>
    [JsonPropertyName("@id")]
    99+ references | Alexander van Delft, 28 days ago | 3 authors, 4 changes | 1 work item
    public Guid Id { get; set; }

    [JsonPropertyName("@type")]
    2 references | Alexander van Delft, 28 days ago | 1 author, 1 change
    public string Type => "UserGroup";

    /// <summary>
    /// </summary>
    7 references | Alexander van Delft, 28 days ago | 2 authors, 2 changes
    public Guid[] UserGroupCanAccessModules { get; set; } = Array.Empty<Guid>();

    /// <summary>
    /// </summary>
    7 references | Alexander van Delft, 28 days ago | 2 authors, 2 changes
    public Guid[] UserGroupCanAssignUserToUserGroup1s { get; set; } = Array.Empty<Guid>();

    /// <summary>
    /// </summary>
    7 references | Alexander van Delft, 28 days ago | 2 authors, 2 changes
    public Guid[] UserGroupCanAssignUserToUserGroup2s { get; set; } = Array.Empty<Guid>();

    /// <summary>
    /// </summary>
    7 references | Alexander van Delft, 28 days ago | 2 authors, 2 changes
    public Guid[] UserGroupContainsPersons { get; set; } = Array.Empty<Guid>();

    /// <summary>
    /// </summary>
    7 references | a.vandelft@rheagroup.com, 58 days ago | 1 author, 1 change
    public Guid UserGroupHasRole { get; set; }

    /// <summary>
    /// </summary>
    7 references | Alexander van Delft, 28 days ago | 1 author, 2 changes
    public Guid[] UserGroupIsInvolvedInUserGroupAccesss { get; set; } = Array.Empty<Guid>();

    /// <summary>
    /// </summary>
    2 references | Omar Elebiary, 64 days ago | 1 author, 1 change | 1 work item
    public string UserGroupName { get; set; }
}
```





Properties

UserGroupContainsPerson FactType

ConstraintDisplayPosition	Top
DerivationNote	
DisplayOrientation	Horizontal
DisplayRoleNames	User Default
InformalDescription	
Name	**UserGroupContainsPerson**
Note	
ObjectifyingEntityType	**UserGroupContainsPerson**
RelationshipType	Aggregation

C# Framework for writing OO transformation of ORM models for code generation

https://github.com/RHEAGROUP/Kalliope

THANK YOU