Generative Concurrent Design

Final Presentation

Objectives & Scope

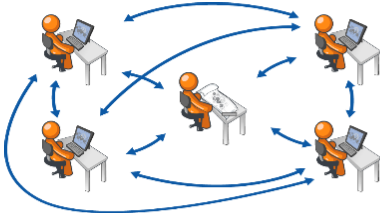
Objectives

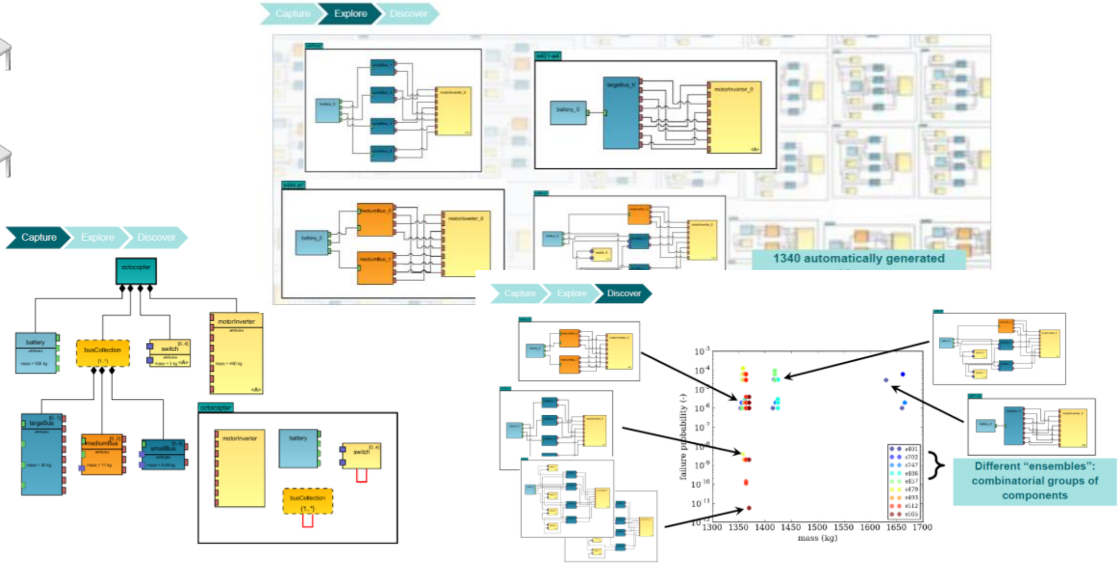
RHEA GROUP

OHB

SIEMENS
Ingenuity for life

Merging collaborative engineering and generative design.





1340 automatically generated

Different "ensembles": combinatorial groups of components

Objectives

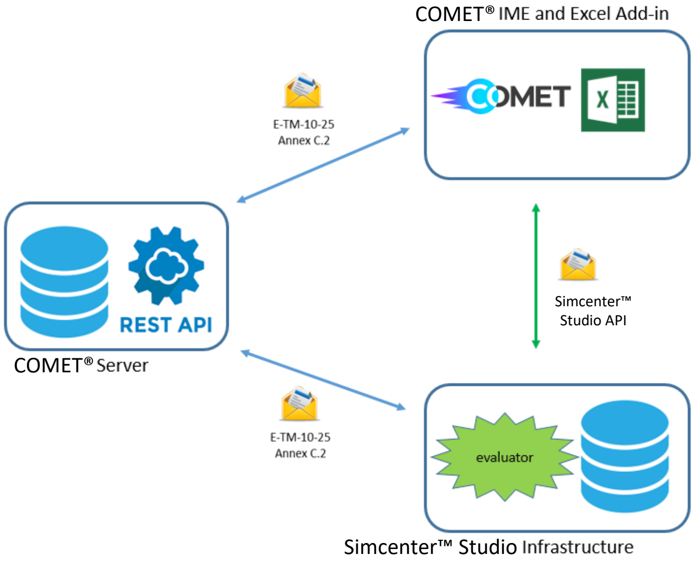
- Define user requirements and validation scenarios
- Develop a beta version of the COMET® based on the updated E-TM-10-25, and of the Simcenter Studio tool, that is suitable to be used in a Generative Concurrent Design environment
- Validation of the GCD software, the combination of COMET® and Simcenter Studio, by means of 6 distinct use cases

- Primarily intended to be used in Phases 0/A → Early concept exploration

Generative Concurrent Design

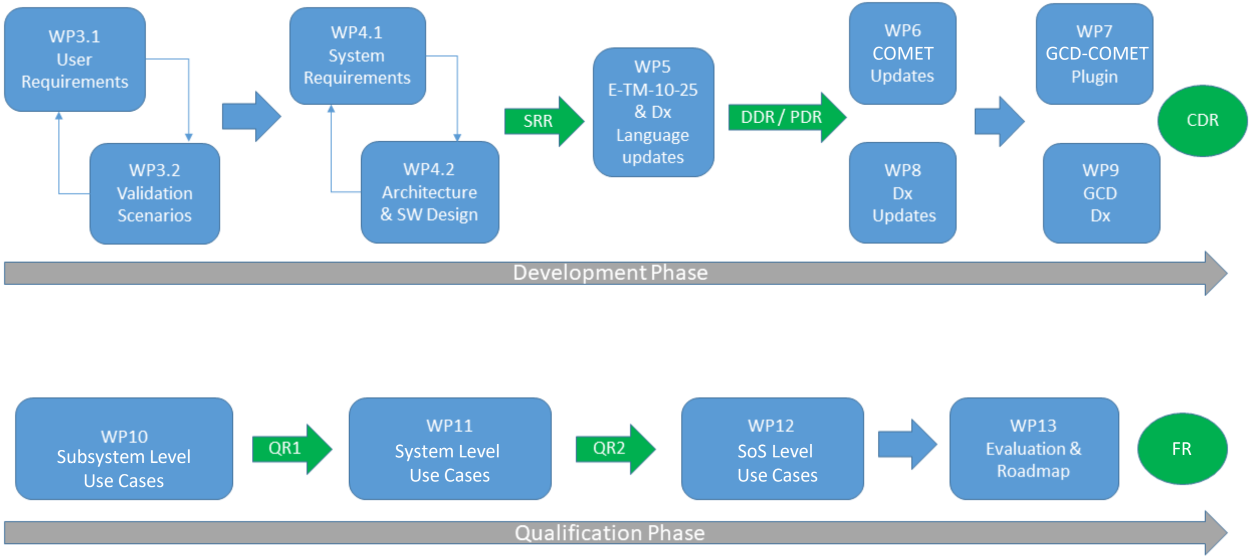
“The proposed technology development is the combination of the RHEA **COMET®** collaborative engineering capability and the generative design capability of Siemens’ **Simcenter™ Studio**.”

“The proposed tooling will be able to automatically generate and evaluate system variants that will allow a team to focus on those design alternatives that are most promising.”



Objectives

Merging collaborative engineering and generative design.



Partners

Generative Concurrent Design

OHB →

- User Requirements
- CONOPS & Use Cases
- Validation

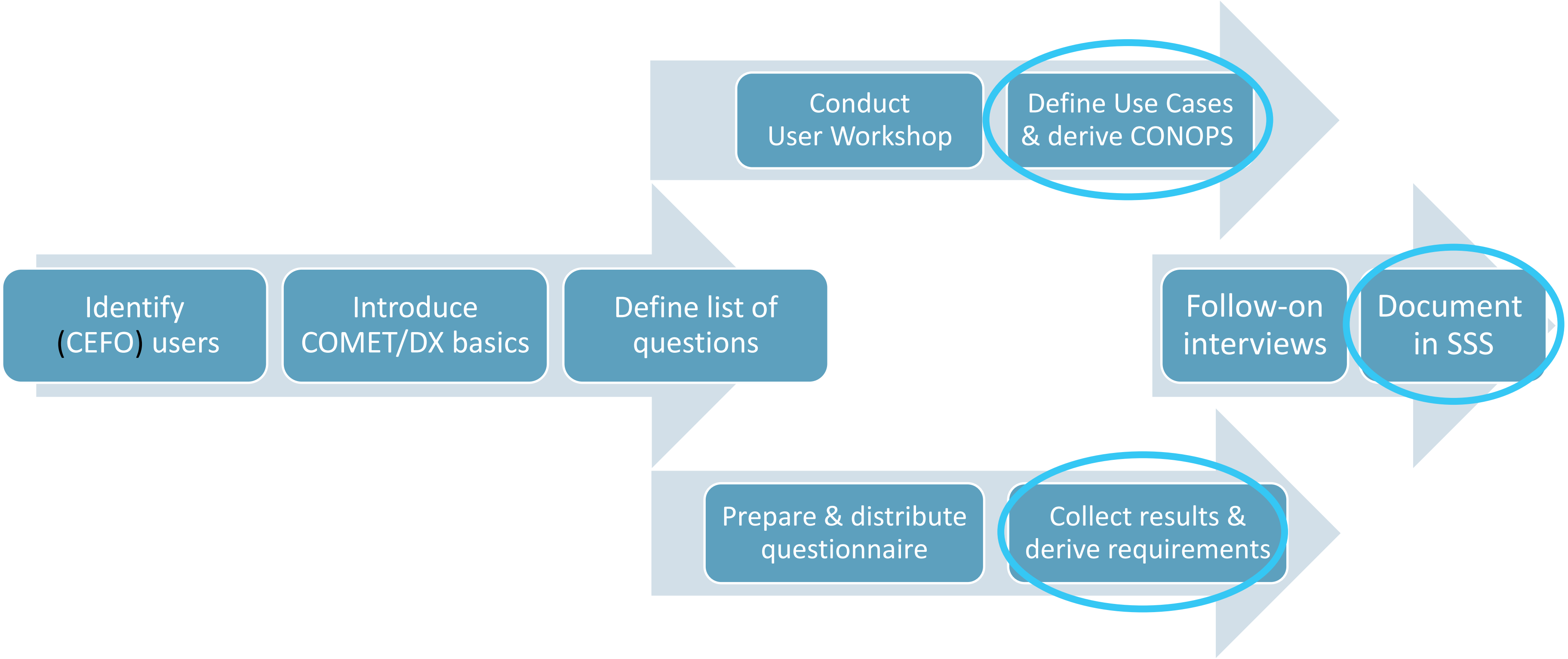
SIEMENS
Ingenuity for life →

- Language Update & Mapping
- Simcenter Studio Updates
- Interface between Studio & E-TM-10-25

Analysis

Requirements, Use Cases & CONOPS

Requirements Elicitation/User Requirements – Process



Identify (CEFO) users

Introduce COMET/DX basics

Define list of questions

Conduct User Workshop

Define Use Cases & derive CONOPS

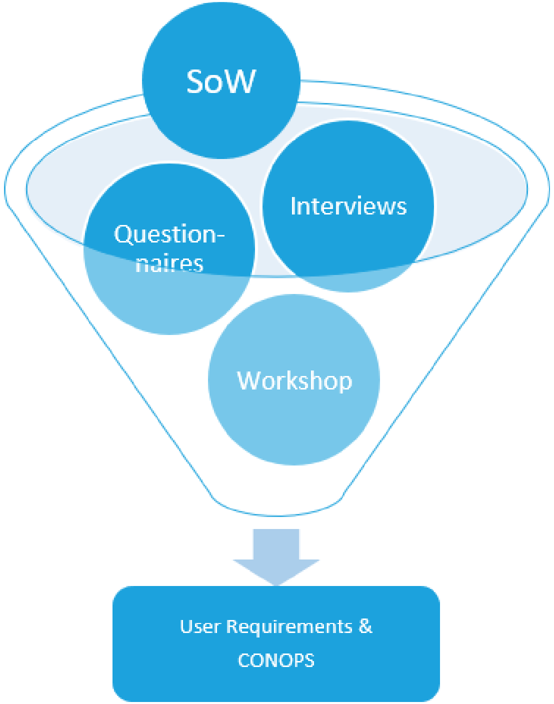
Prepare & distribute questionnaire

Collect results & derive requirements

Follow-on interviews

Document in SSS

Requirements Elicitation/User Requirements - URD & CONOPS



User Requirements:

- Level of integration between COMET & DX
- Responsibilities for the different steps & hierarchy levels
- Diagramming and result visualization, user experience & interactions

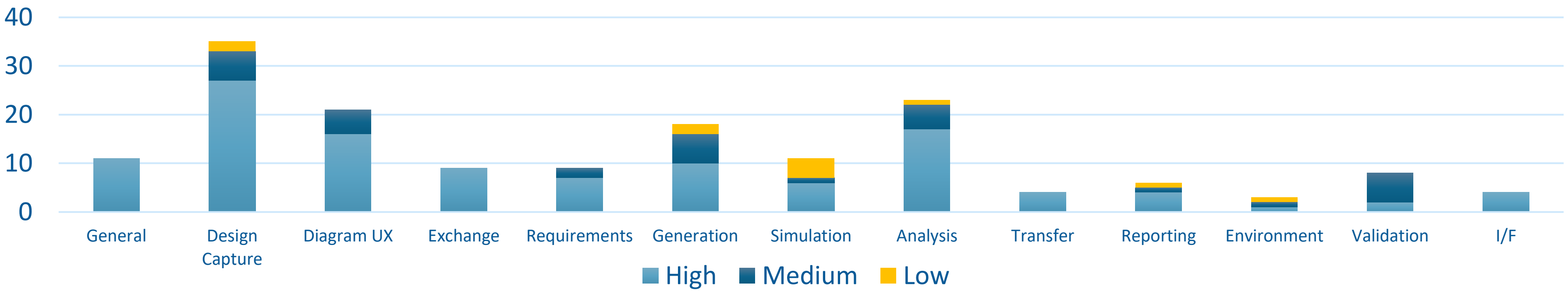
Concept of Operations

- Work flow and use cases of the tool-chain in typical environment (e.g. CDF / CEFO)

Requirements Elicitation/User Requirements – Questions/Statistics

- Total number of User Requirements: 162
- Specifics per GCD function:
  - Requirements modelling → required functions & expressions
  - Design Capture → information types, 10-25 concepts, representation & sync
  - Exchange → ownership, merging, publication
  - Generation → setup, execution, representation, selection
  - Simulation → tools, execution
  - Solution Space Analysis → visualization, filtering, comparison / scoring, selection
  - Transfer → 10-25 model update / creation
  - Reporting → setup, data selection, export

Number of Requirements per Category



CONOPS – Overview & Steps

Step 0: Problem Space Abstraction

Workspaces: White-board sketches, round table discussion, informal representations, etc.

Step 1: Design Space Capture

Workspaces: Elements Definition and Architecture Definition windows

Step 2: Design Space Generation

Workspaces: Solver setup window, Architecture Instances check window

Step 3: Simulation and Evaluation

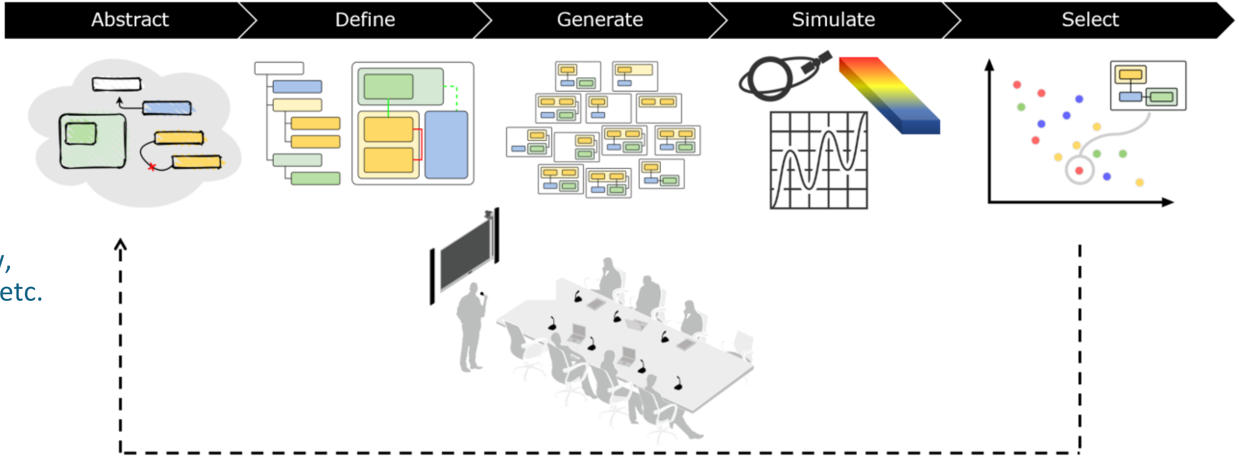
Workspaces: DST simulation setup window, Excel spreadsheet, scripting environment, etc.

Step 4: Solution Space Analysis

Workspaces: Simcenter Studio Discover

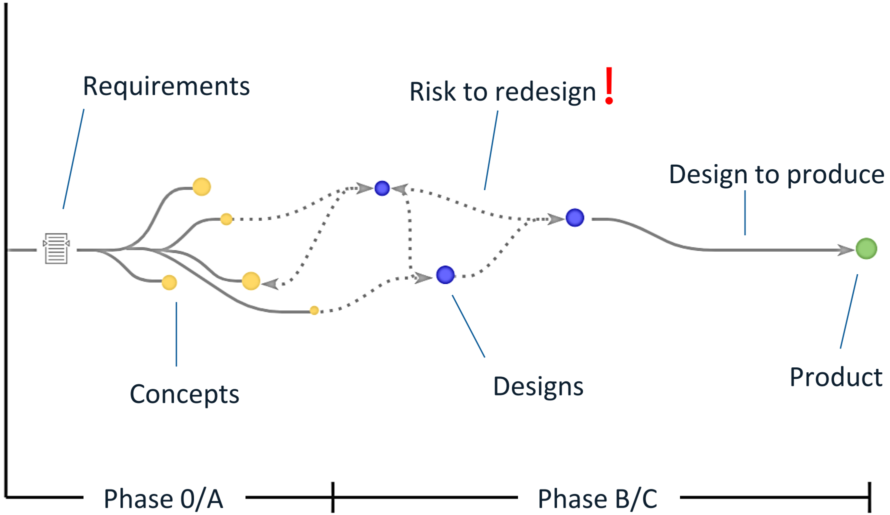
Step 5: Transfer

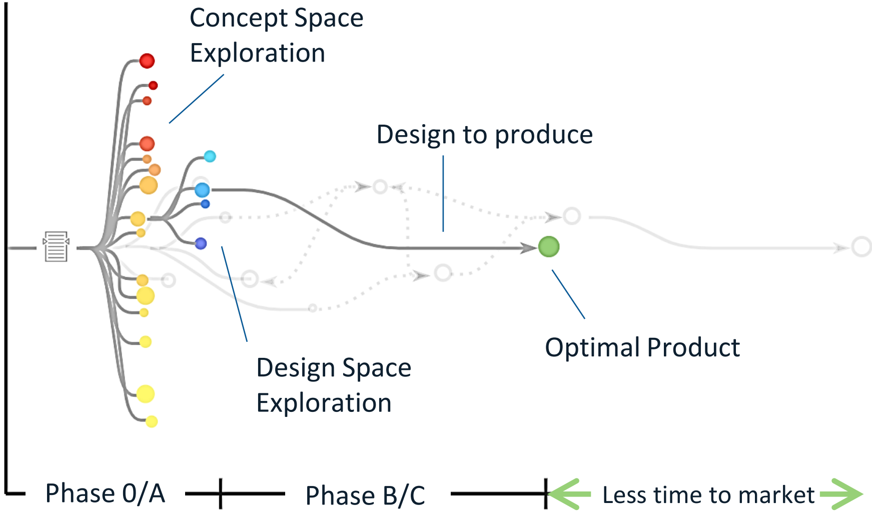
Workspaces: Transfer check window



Differences: Traditional CD / Generative CD

Design Space vs. Timeline





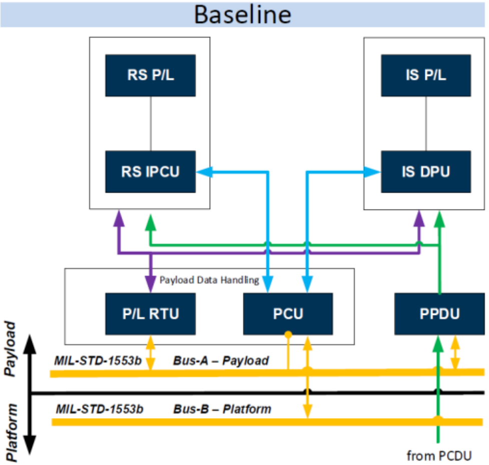
Use Cases

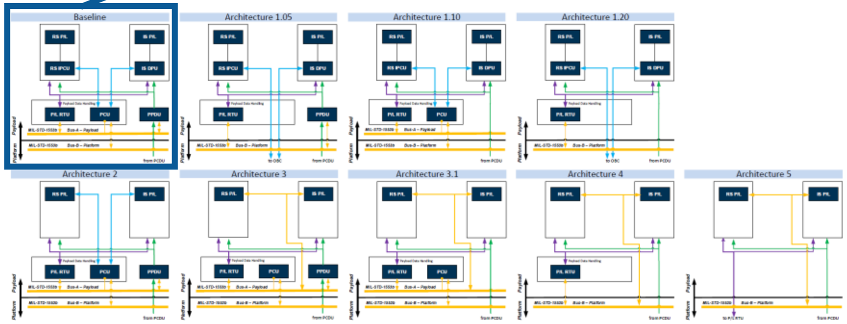
Validation based on use cases defined for different hierarchy levels

ID	Test Case	Use Case
#1	Subsystem Test Case #1	UC#1: Avionics & Electrical Architecture (reduced to Avionics S/S)
#2	Subsystem Test Case #2	UC#2: Propulsion Architecture (EP vs. CP) (reduced to Propulsion S/S)
#3	System Test Case #1	UC#3: Data Return / COM Architecture
#4	System Test Case #2	UC#4: Configuration & Assembly
#5	System of System Test Case #1	UC#5: Earth Observation Optical System - Mission Architecture & Orbit selection
#6	System of System Test Case #2	UC#6: Copernicus Next Generation Architecture

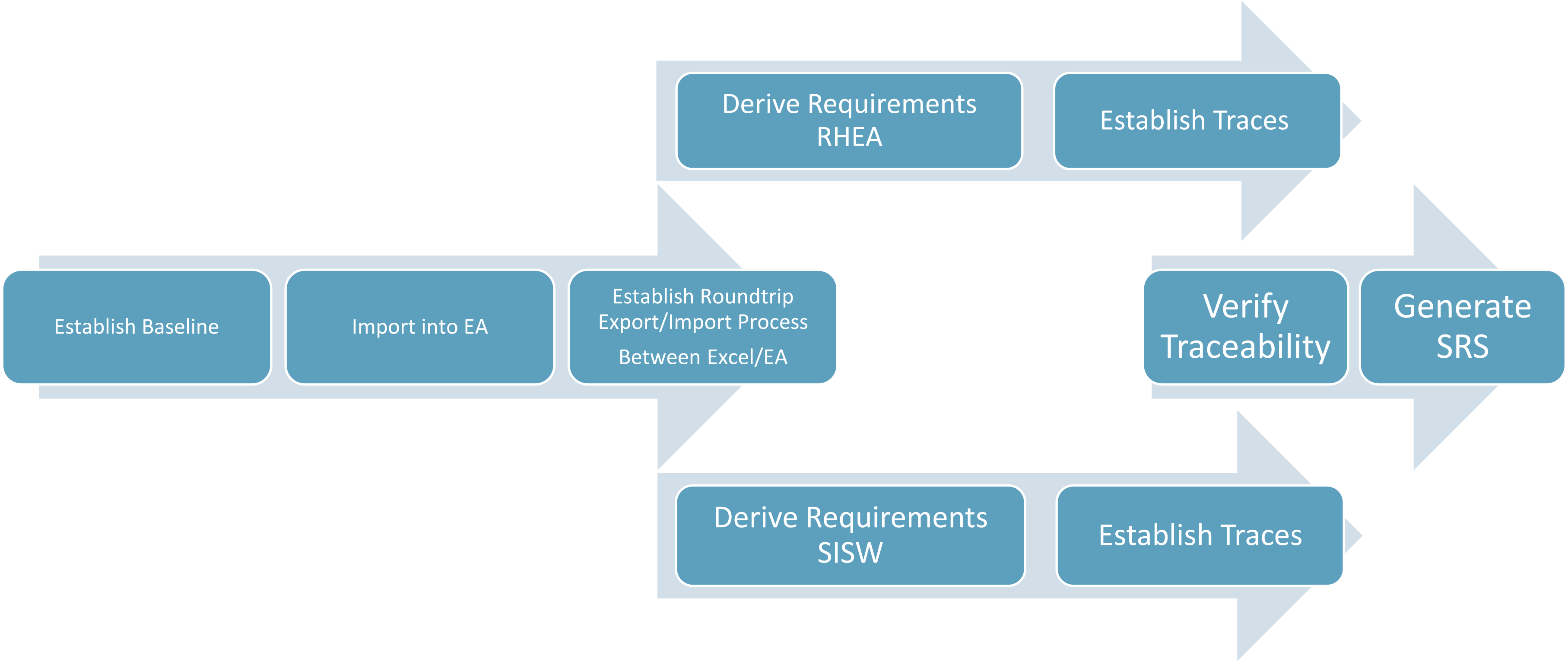
Use Cases – Example: UC#1 Avionics Subsystem

- Multiple architectures possible on payload interface level or data handling level (e.g., SMU vs OBC + MMU)
- Unit selection based on hardware matrix to represent architectures with components from different supplier
- Typical DH subsystem requirements allow to cross-check with architecture
  - e.g., The total mass, including all margins of the DHS shall not exceed 28 kg.
- Parameters are given based on evaluation criteria and key interest to conclude trade-off





Requirements Derivation/Software Requirements - Process



Establish Baseline

Import into EA

Establish Roundtrip Export/Import Process

Between Excel/EA

Derive Requirements RHEA

Establish Traces

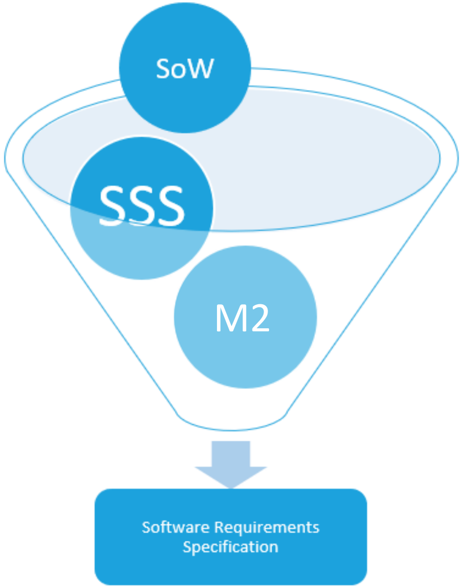
Derive Requirements SISW

Establish Traces

Verify Traceability

Generate SRS

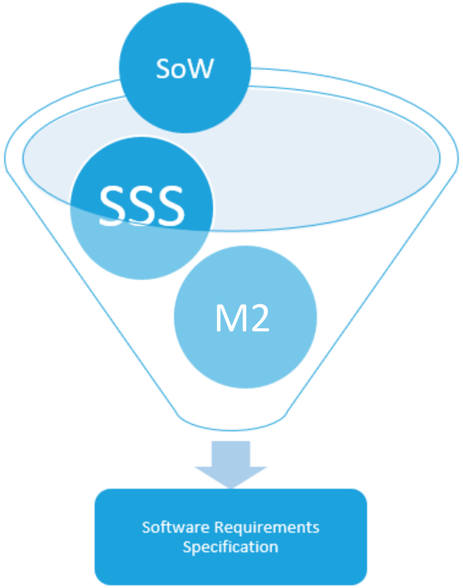
Software Requirements

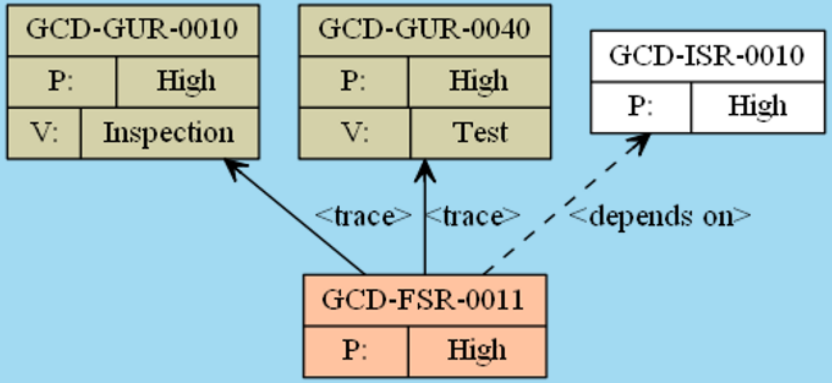


Software Requirements:

- Functional and non-functional requirements of the individual tools and the tool-chain as a whole
- Traces back to each User Requirement
- Priority:
  - High – absolutely needs to be implemented
  - Medium – implementation after high priority requirements
  - Low – nice to have, will only be implemented if effort allows

Software Requirements



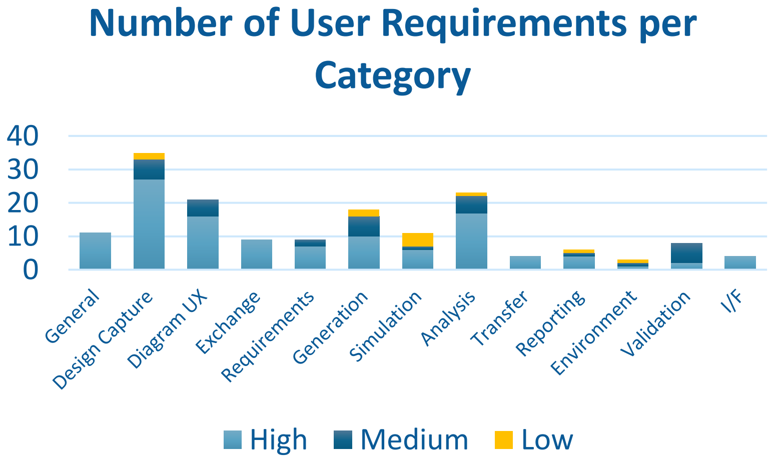
ID	GCD-FSR-0011
Statement	The software must persist Architecture Definition diagrams together with the EngineeringModel.
SSS Trace	GCD-GUR-0040; GCD-GUR-0010;
Depends On	GCD-ISR-0010;
Priority	High
Graph	

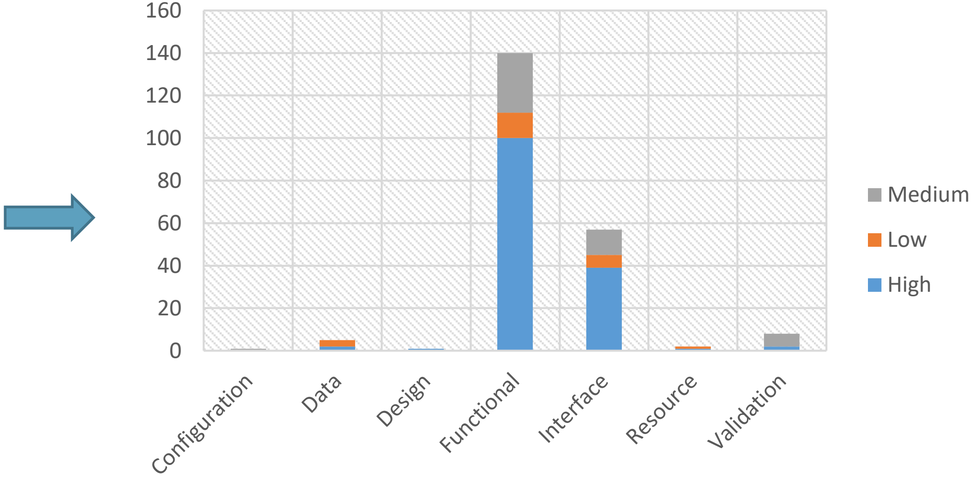
Software Requirements - Statistics

- Total number of Software Requirements: 214 derived from 165 (3 more than at SSS based on feedback) user requirements.
  - High: 145
  - Medium: 47
  - Low: 22

Functional: 140

Non-functional: 74

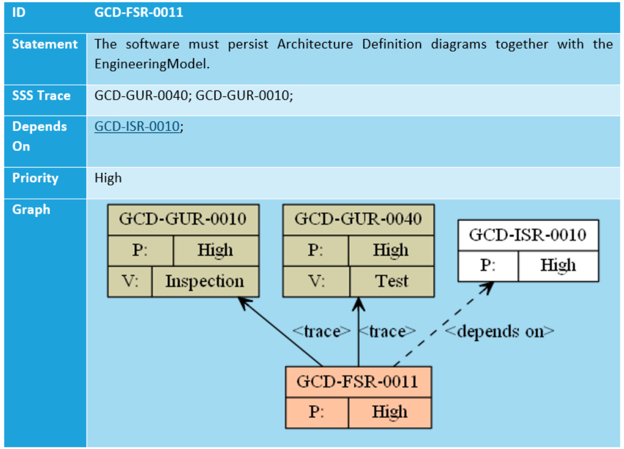




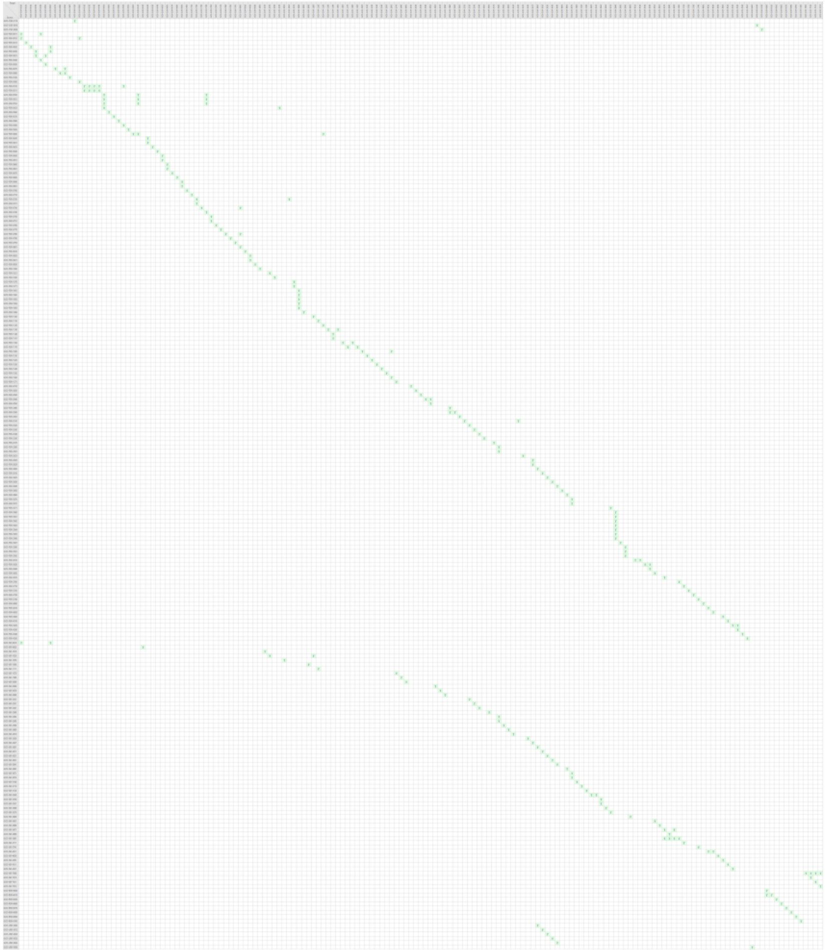
Requirements Tracing

EA Model

D4A-GCD-SRS-TMATRIX (Excel and PNG)

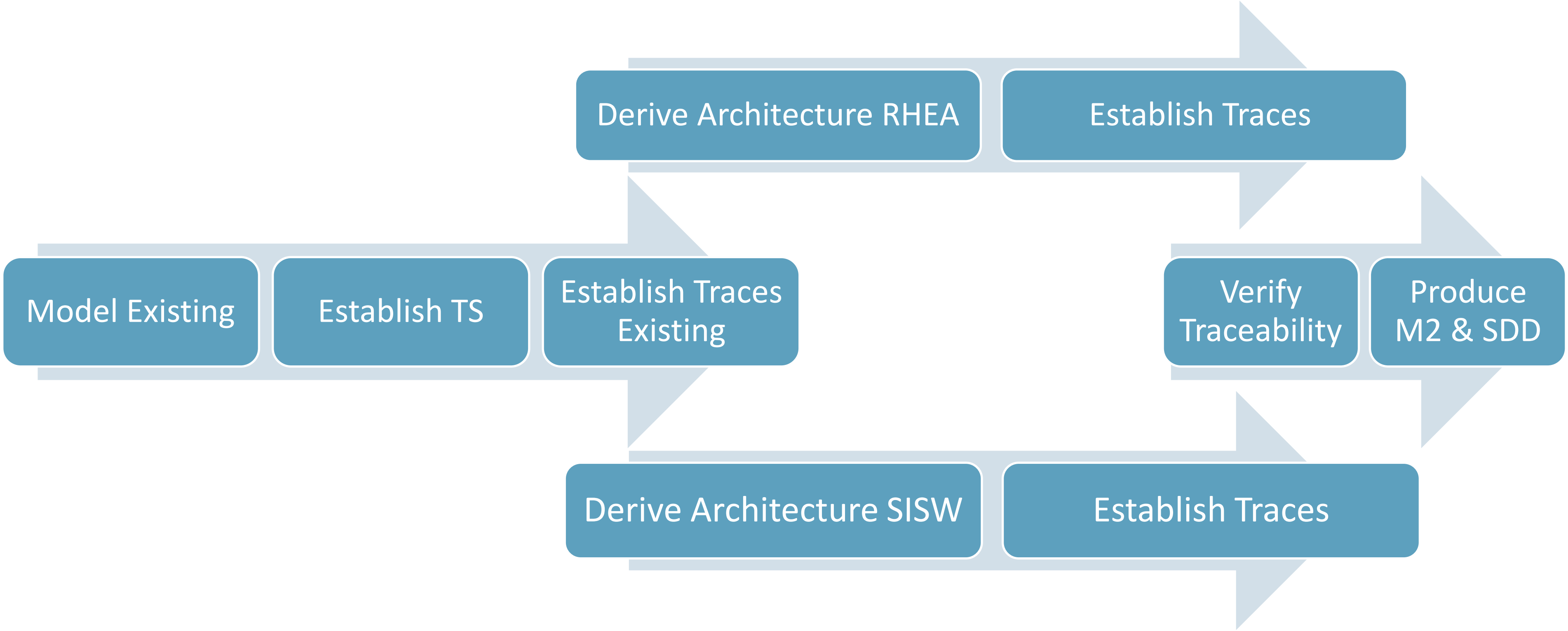


ID	GCD-FSR-0011
Statement	The software must persist Architecture Definition diagrams together with the EngineeringModel.
SSS Trace	GCD-GUR-0040; GCD-GUR-0010;
Depends On	GCD-ISR-0010;
Priority	High
Graph	GCD-GUR-0010 P: High V: Inspection GCD-GUR-0040 P: High V: Test GCD-ISR-0010 P: High <trace> <trace> <depends on> GCD-FSR-0011 P: High



Architecture

Architecture - Process



Model Existing

Establish TS

Establish Traces Existing

Derive Architecture RHEA

Establish Traces

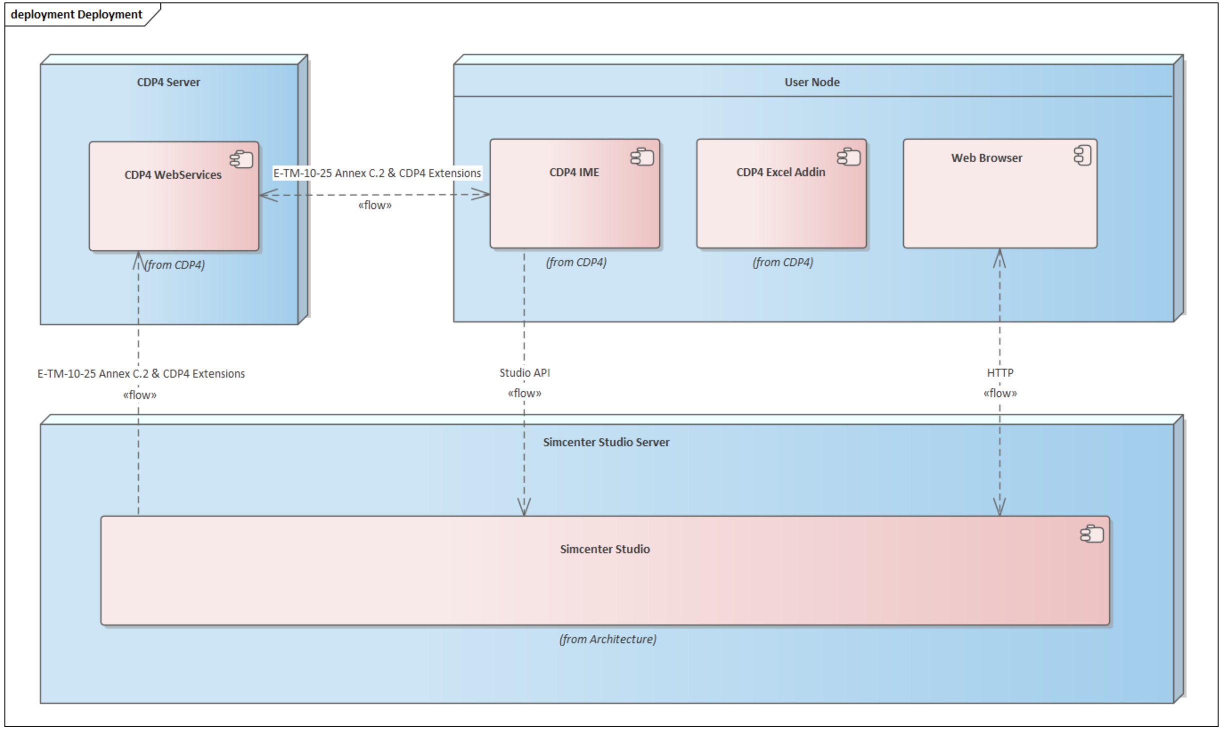
Derive Architecture SISW

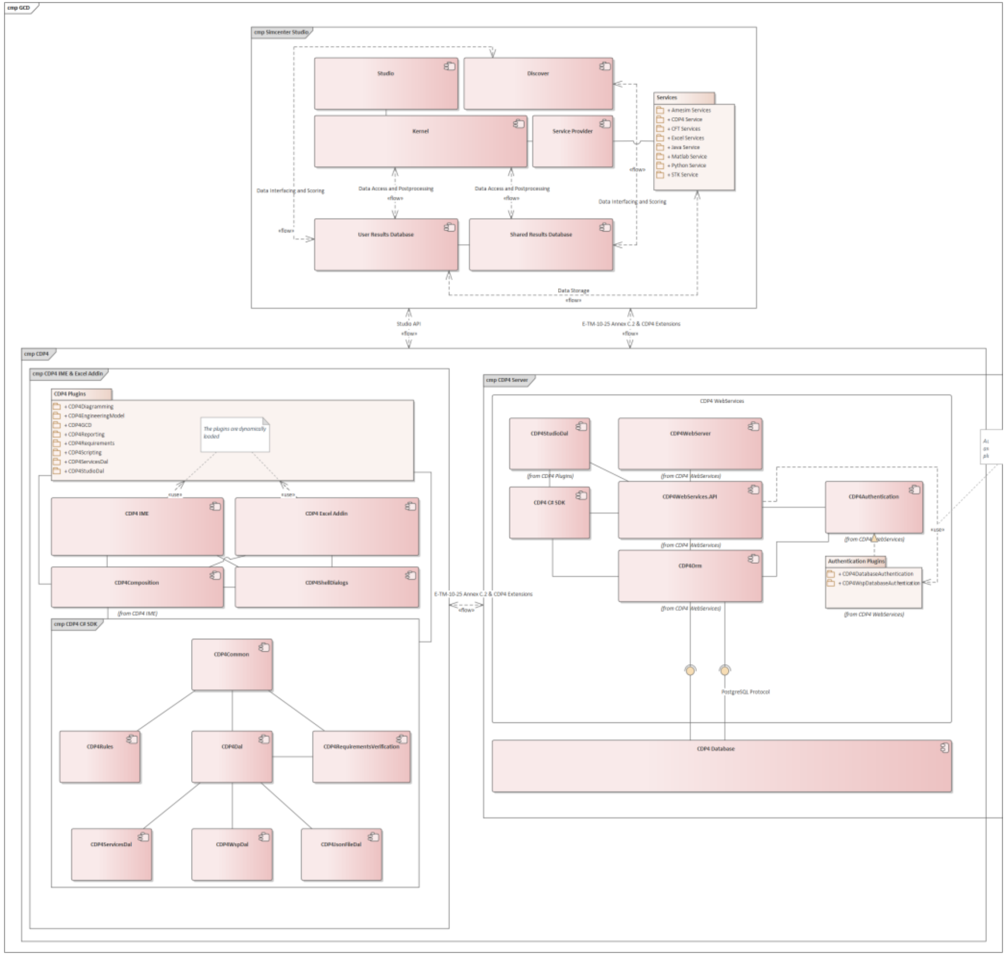
Establish Traces

Verify Traceability

Produce M2 & SDD

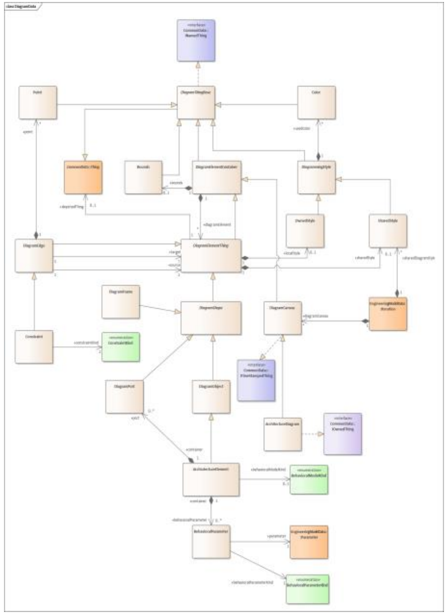
Architecture – Deployment

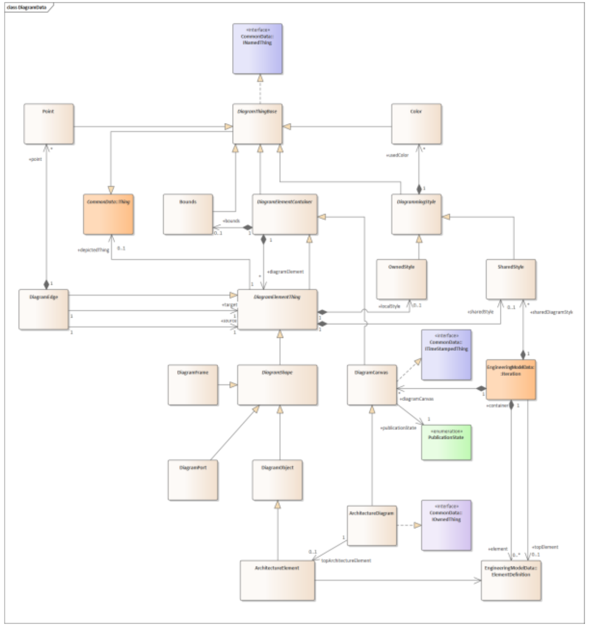


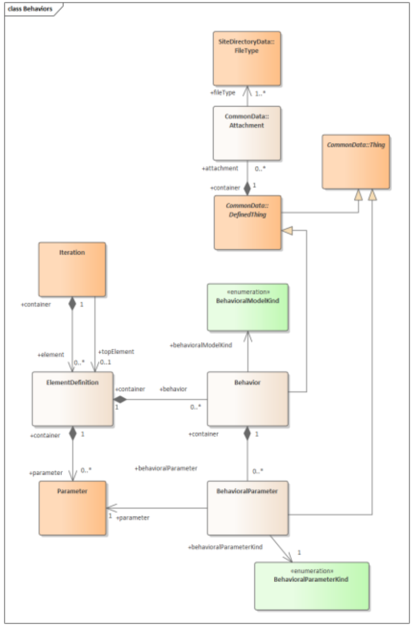

cmp GCD
cmp Simcenter Studio
Studio
Discover
Kernel
Service Provider
Services
+ Amesim Services
+ CDP4 Service
+ CFT Services
+ Excel Services
+ Java Service
+ Matlab Service
+ Python Service
+ STK Service
Data Interfacing and Scoring
«flow»
Data Access and Postprocessing
«flow»
Data Access and Postprocessing
«flow»
«flow»
Data Interfacing and Scoring
User Results Database
Shared Results Database
Data Storage
«flow»
Studio API
«flow»
E-TM-10-25 Annex C.2 & CDP4 Extensions
«flow»
cmp CDP4
cmp CDP4 IME & Excel Addin
CDP4 Plugins
+ CDP4Diagramming
+ CDP4EngineeringModel
+ CDP4GCD
+ CDP4Reporting
+ CDP4Requirements
+ CDP4Scripting
+ CDP4ServicesDal
+ CDP4StudioDal
The plugins are dynamically loaded
«use»
«use»
CDP4 IME
CDP4 Excel Addin
CDP4Composition
CDP4ShellDialogs
(from CDP4 IME)
cmp CDP4 C# SDK
CDP4Common
CDP4Rules
CDP4Dal
CDP4RequirementsVerification
CDP4ServicesDal
CDP4WspDal
CDP4JsonFileDal
E-TM-10-25 Annex C.2 & CDP4 Extensions
«flow»
cmp CDP4 Server
CDP4 WebServices
CDP4StudioDal
(from CDP4 Plugins)
CDP4WebServer
(from CDP4 WebServices)
CDP4 C# SDK
CDP4WebServices.API
(from CDP4 WebServices)
CDP4Authentication
(from CDP4 WebServices)
«use»
Authentication Plugins
+ CDP4DatabaseAuthentication
+ CDP4WspDatabaseAuthentication
(from CDP4 WebServices)
CDP4Orm
(from CDP4 WebServices)
PostgreSQL Protocol
CDP4 Database

Software Architecture Updates (SDD) – Data Model

- No direct updates to E-TM-10-25 due to lengthy process
- Updates carried out as COMET Master model extensions for fast prototyping
- Resulting in COMET Master Model version 1.3 → Can eventually be transferred to E-TM-10-25







Simcenter™ Studio Language Updates

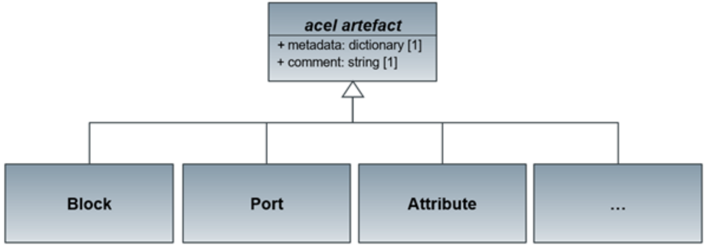
D15 Generative Concurrent Design – Simcenter Studio Language Updates

Maintain traceability to E-TM-10-25: Metadata (UUID)

Other:

Comments [GCD-FSR-0650], [0651], [3650]

Metadata for files, scoring [0690], [2628], [2629]

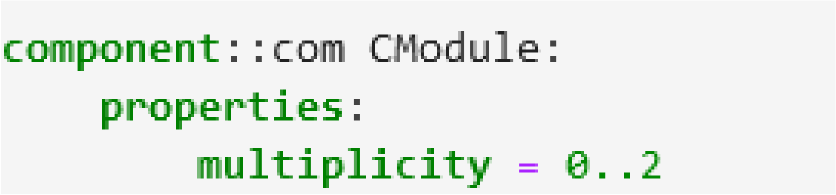


Specific syntax additions for managing new COMET services

COMET® - Simcenter™ Studio Language Mapping

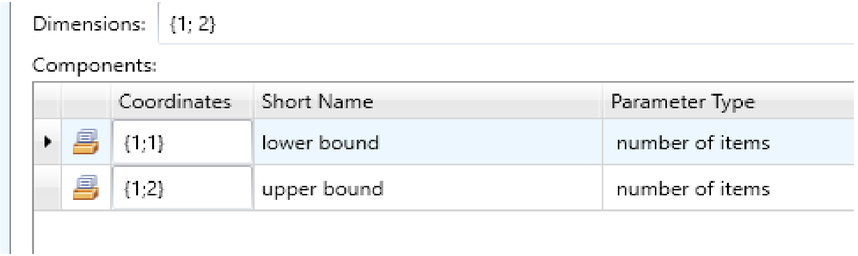
D16 Generative Concurrent Design E-TM-10-25 Dx Language Mapping

Multiplicity



```
component::com CModule:
    properties:
        multiplicity = 0..2
```

Array Parameter Type

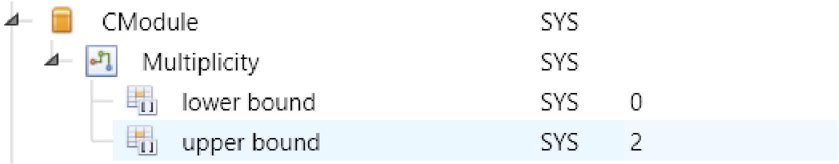


Dimensions: {1; 2}

Components:

Coordinates	Short Name	Parameter Type
{1;1}	lower bound	number of items
{1;2}	upper bound	number of items

Usage

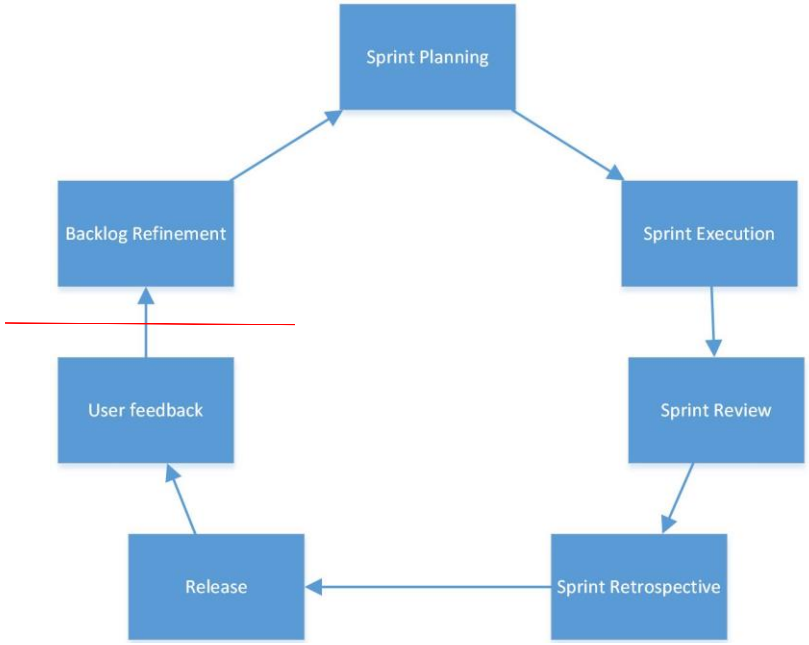


CModule	SYS	
Multiplicity	SYS	
lower bound	SYS	0
upper bound	SYS	2

Development

Software Project Development Lifecycle

User Story Lifecycle



State	Meaning
Open	The work on the story/engineering task has not started yet
In Progress	The user story/engineering task is under implementation.
Under Review	The user story/ engineering task has been developed and has passed automatic unit testing, integration testing and end2end testing and is ready for peer review.
Closed	The user story has been successfully implemented, reviewed so it is closed

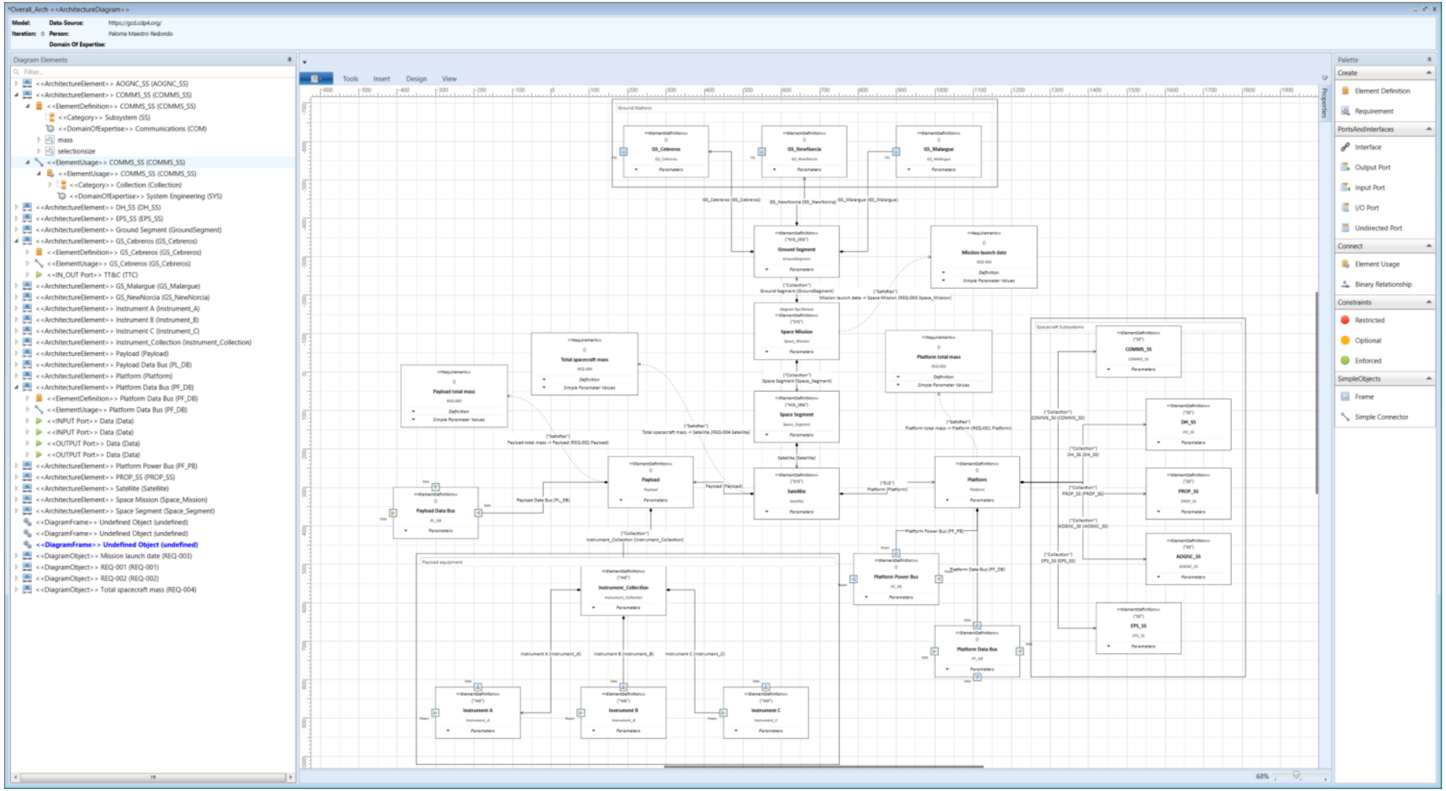
Software Development Environment

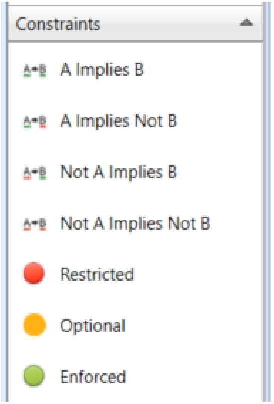
- COMET SDK/IME/WebServices related features included directly on RHEA Group Github and maintained as part of normal development cycle
- Continuous integration and code quality control through Appveyor and Sonarcloud
- Log on to https://github.com/RHEAGROUP/ to find the source code

- GCD Specific COMET Plugin proprietary development and available in the future as part of the Enterprise Edition
- GCD Specific Studio development proprietary

COMET

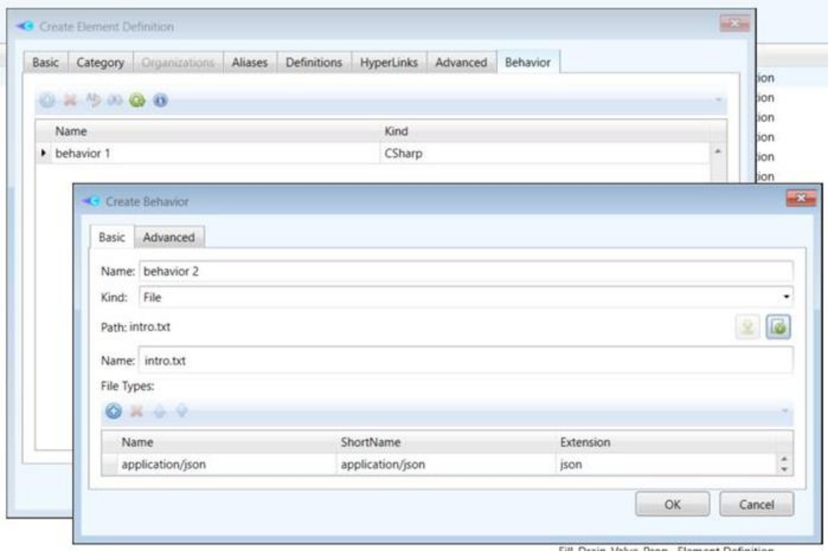
COMET Features → Diagramming





COMET Features → Behaviours

- Describe the dynamic nature of an Element Definition
- Specific scripting languages can be directly used
- Behavioural models defined in binary files (i.e. external scripts, simulation files, Excel sheets) are also supported



Name	Kind
behavior 1	CSharp

Name: behavior 2

Kind: File

Path: intro.txt

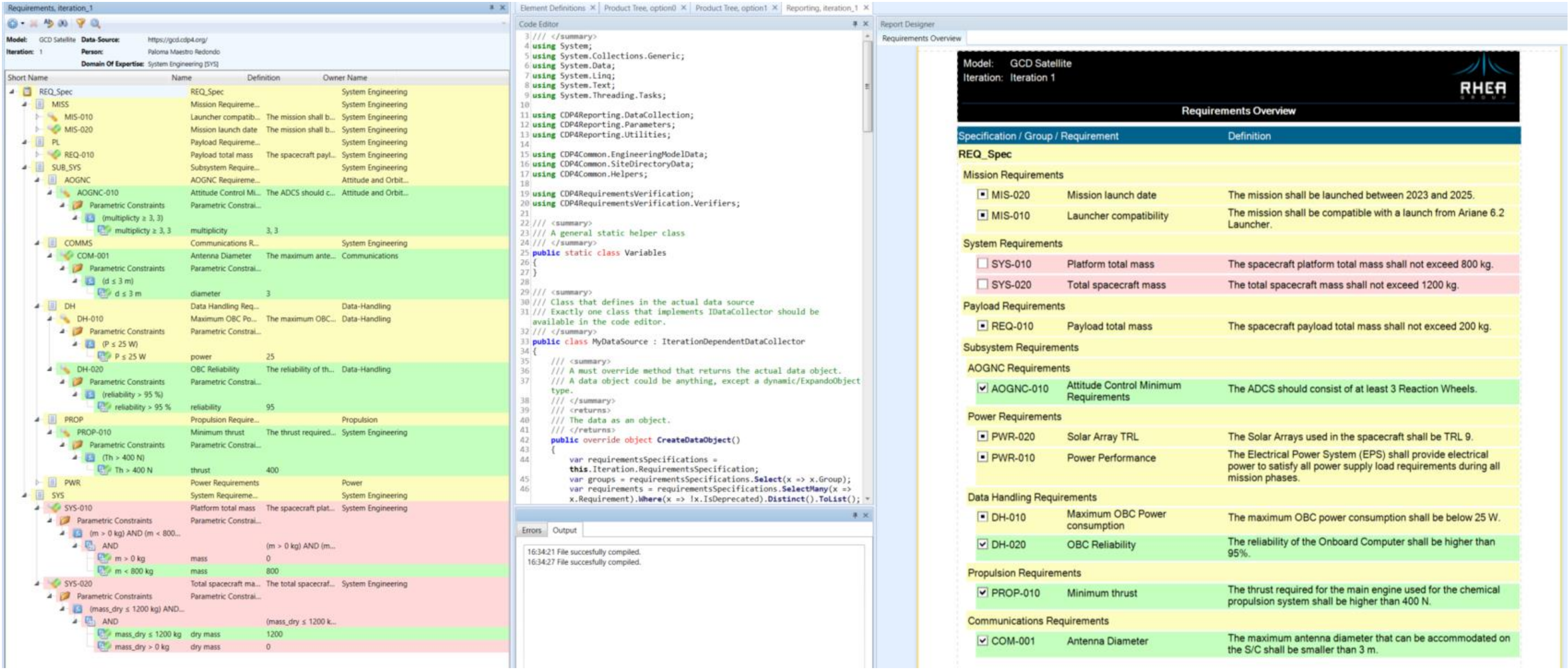
Name: intro.txt

File Types:

Name	ShortName	Extension
application/json	application/json	json

COMET Features → Requirement Verification & Reporting

- Added support for option-dependent verification
- Custom Requirement Verification Report



Model: GCD Satellite **Data-Source:** https://gcd.cdp4.org/
Iteration: 1 **Person:** Paloma Maestro Redondo
Domain Of Expertise: System Engineering [SYS]

Short Name	Name	Definition	Owner Name
REQ_Spec	REQ_Spec		System Engineering
MISS	Mission Requireme...		System Engineering
MIS-010	Launcher compatib...	The mission shall b...	System Engineering
MIS-020	Mission launch date	The mission shall b...	System Engineering
PL	Payload Requireme...		System Engineering
REQ-010	Payload total mass	The spacecraft payl...	System Engineering
SUB_SYS	Subsystem Require...		System Engineering
AOGNC	AOGNC Requireme...		Attitude and Orbit...
AOGNC-010	Attitude Control Mi...	The ADCS should c...	Attitude and Orbit...
Parametric Constraints	Parametric Constrai...		
(multiplicty ≥ 3, 3)			
multiplicty ≥ 3, 3	multiplicity	3, 3	
COMMS	Communications R...		System Engineering
COM-001	Antenna Diameter	The maximum ante...	Communications
Parametric Constraints	Parametric Constrai...		
(d ≤ 3 m)			
d ≤ 3 m	diameter	3	
DH	Data Handling Req...		Data-Handling
DH-010	Maximum OBC Po...	The maximum OBC...	Data-Handling
Parametric Constraints	Parametric Constrai...		
(P ≤ 25 W)			
P ≤ 25 W	power	25	
DH-020	OBC Reliability	The reliability of th...	Data-Handling
Parametric Constraints	Parametric Constrai...		
(reliability > 95 %)			
reliability > 95 %	reliability	95	
PROP	Propulsion Require...		Propulsion
PROP-010	Minimum thrust	The thrust required...	System Engineering
Parametric Constraints	Parametric Constrai...		
(Th > 400 N)			
Th > 400 N	thrust	400	
PWR	Power Requirements		Power
SYS	System Requireme...		System Engineering
SYS-010	Platform total mass	The spacecraft plat...	System Engineering
Parametric Constraints	Parametric Constrai...		
(m > 0 kg) AND (m < 800...			
AND		(m > 0 kg) AND (m...	
m > 0 kg	mass	0	
m < 800 kg	mass	800	
SYS-020	Total spacecraft ma...	The total spacecraf...	System Engineering
Parametric Constraints	Parametric Constrai...		
(mass_dry ≤ 1200 kg) AND...			
AND		(mass_dry ≤ 1200 k...	
mass_dry ≤ 1200 kg	dry mass	1200	
mass_dry > 0 kg	dry mass	0	

```
/// </summary>
using System;
using System.Collections.Generic;
using System.Data;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

using CDP4Reporting.DataCollection;
using CDP4Reporting.Parameters;
using CDP4Reporting.Utilities;

using CDP4Common.EngineeringModelData;
using CDP4Common.SiteDirectoryData;
using CDP4Common.Helpers;

using CDP4RequirementsVerification;
using CDP4RequirementsVerification.Verifiers;

/// <summary>
/// A general static helper class
/// </summary>
public static class Variables
{
}

/// <summary>
/// Class that defines in the actual data source
/// Exactly one class that implements IDataCollector should be
available in the code editor.
/// </summary>
public class MyDataSource : IterationDependentDataCollector
{
    /// <summary>
    /// A must override method that returns the actual data object.
    /// A data object could be anything, except a dynamic/ExpandoObject
    type.
    /// </summary>
    /// <returns>
    /// The data as an object.
    /// </returns>
    public override object CreateDataObject()
    {
        var requirementsSpecifications =
        this.Iteration.RequirementsSpecification;
        var groups = requirementsSpecifications.Select(x => x.Group);
        var requirements = requirementsSpecifications.SelectMany(x =>
        x.Requirement).Where(x => !x.IsDeprecated).Distinct().ToList();
```

```
16:34:21 File succesfully compiled.
16:34:27 File succesfully compiled.
```

Model: GCD Satellite
Iteration: Iteration 1

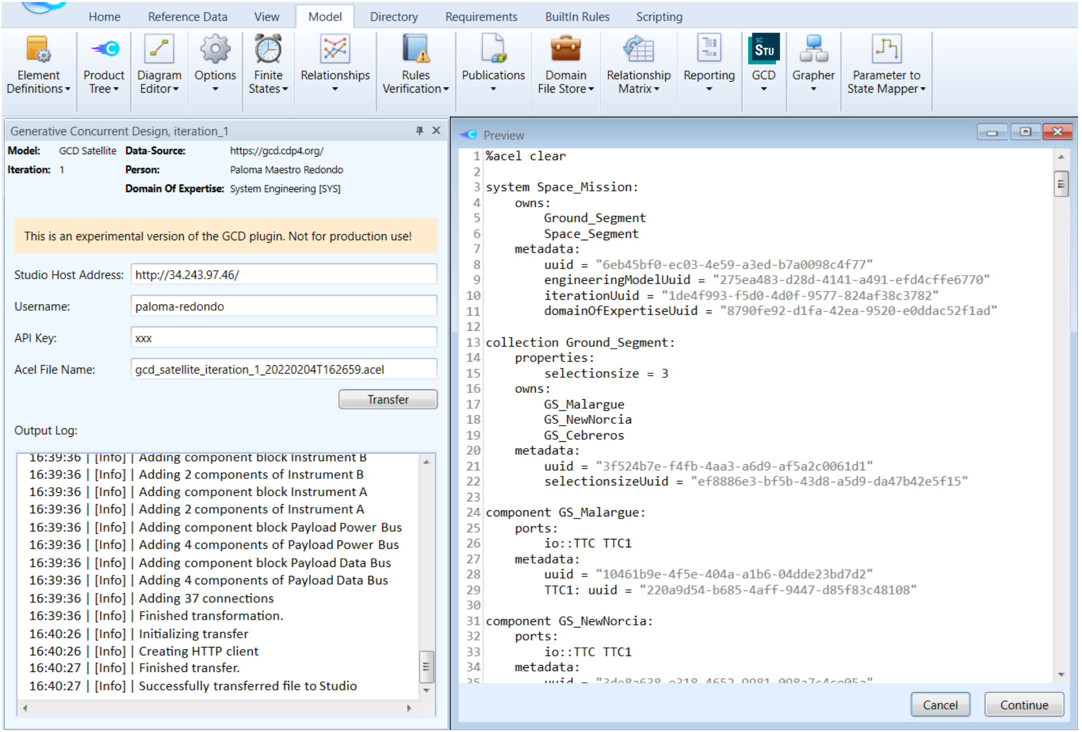
RHEA GROUP

Requirements Overview

Specification / Group / Requirement		Definition
REQ_Spec		
Mission Requirements		
▣ MIS-020	Mission launch date	The mission shall be launched between 2023 and 2025.
▣ MIS-010	Launcher compatibility	The mission shall be compatible with a launch from Ariane 6.2 Launcher.
System Requirements		
☐ SYS-010	Platform total mass	The spacecraft platform total mass shall not exceed 800 kg.
☐ SYS-020	Total spacecraft mass	The total spacecraft mass shall not exceed 1200 kg.
Payload Requirements		
▣ REQ-010	Payload total mass	The spacecraft payload total mass shall not exceed 200 kg.
Subsystem Requirements		
AOGNC Requirements		
☑ AOGNC-010	Attitude Control Minimum Requirements	The ADCS should consist of at least 3 Reaction Wheels.
Power Requirements		
▣ PWR-020	Solar Array TRL	The Solar Arrays used in the spacecraft shall be TRL 9.
▣ PWR-010	Power Performance	The Electrical Power System (EPS) shall provide electrical power to satisfy all power supply load requirements during all mission phases.
Data Handling Requirements		
▣ DH-010	Maximum OBC Power consumption	The maximum OBC power consumption shall be below 25 W.
☑ DH-020	OBC Reliability	The reliability of the Onboard Computer shall be higher than 95%.
Propulsion Requirements		
☑ PROP-010	Minimum thrust	The thrust required for the main engine used for the chemical propulsion system shall be higher than 400 N.
Communications Requirements		
☑ COM-001	Antenna Diameter	The maximum antenna diameter that can be accommodated on the S/C shall be smaller than 3 m.

COMET Features → GCD Plugin

- New plugin to establish acel code generation/ transformation from E-TM-10-25 and direct transfer to Studio.
- Rudimentary acel code error checking
- Code preview before transfer



Generative Concurrent Design, iteration_1

Model: GCD Satellite **Data-Source:** https://gcd.cdp4.org/

Iteration: 1 **Person:** Paloma Maestro Redondo

Domain Of Expertise: System Engineering [SYS]

This is an experimental version of the GCD plugin. Not for production use!

Studio Host Address: http://34.243.97.46/

Username: paloma-redondo

API Key: xxx

Acel File Name: gcd_satellite_iteration_1_20220204T162659.acel

Transfer

Output Log:

```
16:39:36 | [Info] | Adding component block Instrument B
16:39:36 | [Info] | Adding 2 components of Instrument B
16:39:36 | [Info] | Adding component block Instrument A
16:39:36 | [Info] | Adding 2 components of Instrument A
16:39:36 | [Info] | Adding component block Payload Power Bus
16:39:36 | [Info] | Adding 4 components of Payload Power Bus
16:39:36 | [Info] | Adding component block Payload Data Bus
16:39:36 | [Info] | Adding 4 components of Payload Data Bus
16:39:36 | [Info] | Adding 37 connections
16:39:36 | [Info] | Finished transformation.
16:40:26 | [Info] | Initializing transfer
16:40:26 | [Info] | Creating HTTP client
16:40:27 | [Info] | Finished transfer.
16:40:27 | [Info] | Successfully transferred file to Studio
```

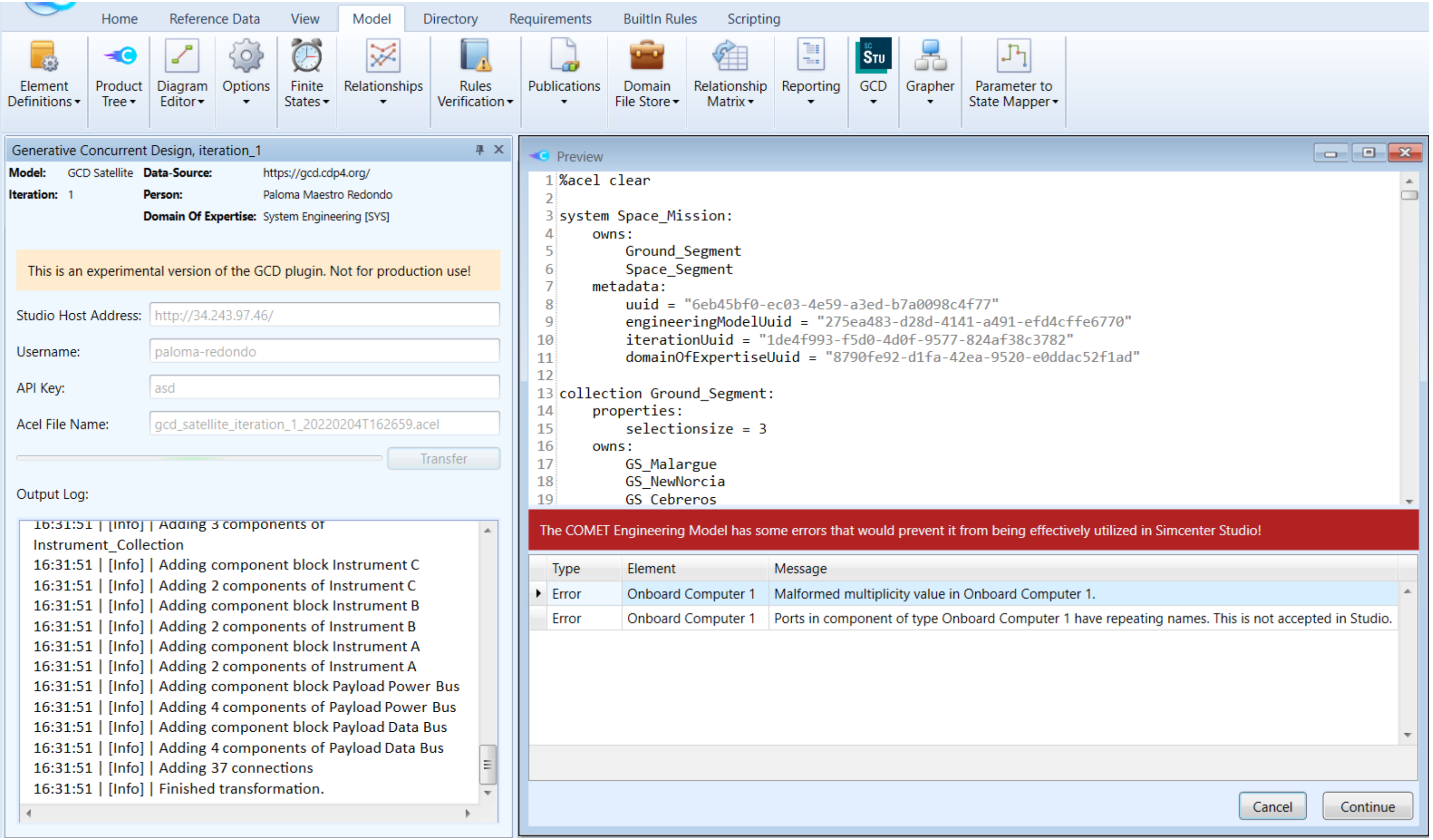
```
%acel clear

system Space_Mission:
    owns:
        Ground_Segment
        Space_Segment
    metadata:
        uuid = "6eb45bf0-ec03-4e59-a3ed-b7a0098c4f77"
        engineeringModelUuid = "275ea483-d28d-4141-a491-efd4cffe6770"
        iterationUuid = "1de4f993-f5d0-4d0f-9577-824af38c3782"
        domainOfExpertiseUuid = "8790fe92-d1fa-42ea-9520-e0ddac52f1ad"

collection Ground_Segment:
    properties:
        selectionsize = 3
    owns:
        GS_Malargue
        GS_NewNorcia
        GS_Cebreros
    metadata:
        uuid = "3f524b7e-f4fb-4aa3-a6d9-af5a2c0061d1"
        selectionsizeUuid = "ef8886e3-bf5b-43d8-a5d9-da47b42e5f15"

component GS_Malargue:
    ports:
        io::TTC TTC1
    metadata:
        uuid = "10461b9e-4f5e-404a-a1b6-04dde23bd7d2"
        TTC1: uuid = "220a9d54-b685-4aff-9447-d85f83c48108"

component GS_NewNorcia:
    ports:
        io::TTC TTC1
    metadata:
```



Generative Concurrent Design, iteration_1

Model: GCD Satellite **Data-Source:** https://gcd.cdp4.org/

Iteration: 1 **Person:** Paloma Maestro Redondo

Domain Of Expertise: System Engineering [SYS]

This is an experimental version of the GCD plugin. Not for production use!

Studio Host Address: http://34.243.97.46/

Username: paloma-redondo

API Key: asd

Acel File Name: gcd_satellite_iteration_1_20220204T162659.acel

Output Log:

```
16:31:51 | [Info] | Adding 3 components of
Instrument_Collection
16:31:51 | [Info] | Adding component block Instrument C
16:31:51 | [Info] | Adding 2 components of Instrument C
16:31:51 | [Info] | Adding component block Instrument B
16:31:51 | [Info] | Adding 2 components of Instrument B
16:31:51 | [Info] | Adding component block Instrument A
16:31:51 | [Info] | Adding 2 components of Instrument A
16:31:51 | [Info] | Adding component block Payload Power Bus
16:31:51 | [Info] | Adding 4 components of Payload Power Bus
16:31:51 | [Info] | Adding component block Payload Data Bus
16:31:51 | [Info] | Adding 4 components of Payload Data Bus
16:31:51 | [Info] | Adding 37 connections
16:31:51 | [Info] | Finished transformation.
```

```
%acel clear

system Space_Mission:
    owns:
        Ground_Segment
        Space_Segment
    metadata:
        uuid = "6eb45bf0-ec03-4e59-a3ed-b7a0098c4f77"
        engineeringModelUuid = "275ea483-d28d-4141-a491-efd4cffe6770"
        iterationUuid = "1de4f993-f5d0-4d0f-9577-824af38c3782"
        domainOfExpertiseUuid = "8790fe92-d1fa-42ea-9520-e0ddac52f1ad"

collection Ground_Segment:
    properties:
        selectionsize = 3
    owns:
        GS_Malargue
        GS_NewNorcia
        GS_Cebreros
```

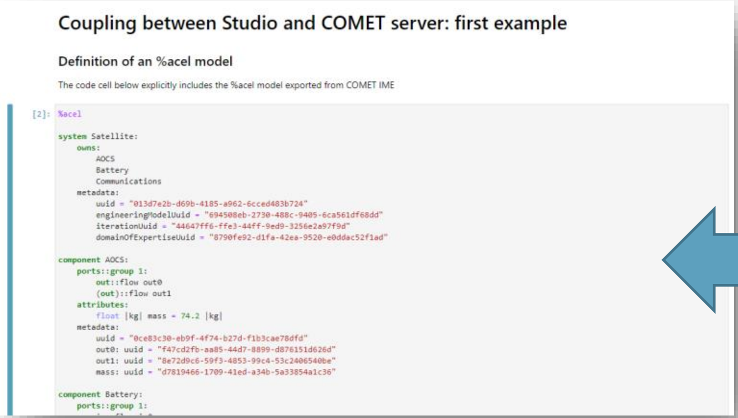
The COMET Engineering Model has some errors that would prevent it from being effectively utilized in Simcenter Studio!

Type	Element	Message
Error	Onboard Computer 1	Malformed multiplicity value in Onboard Computer 1.
Error	Onboard Computer 1	Ports in component of type Onboard Computer 1 have repeating names. This is not accepted in Studio.

Simcenter Studio

Simcenter Studio Features → Transfer of the Model

Directly adding acel model in code cell



Coupling between Studio and COMET server: first example

Definition of an %acel model

The code cell below explicitly includes the %acel model exported from COMET IME

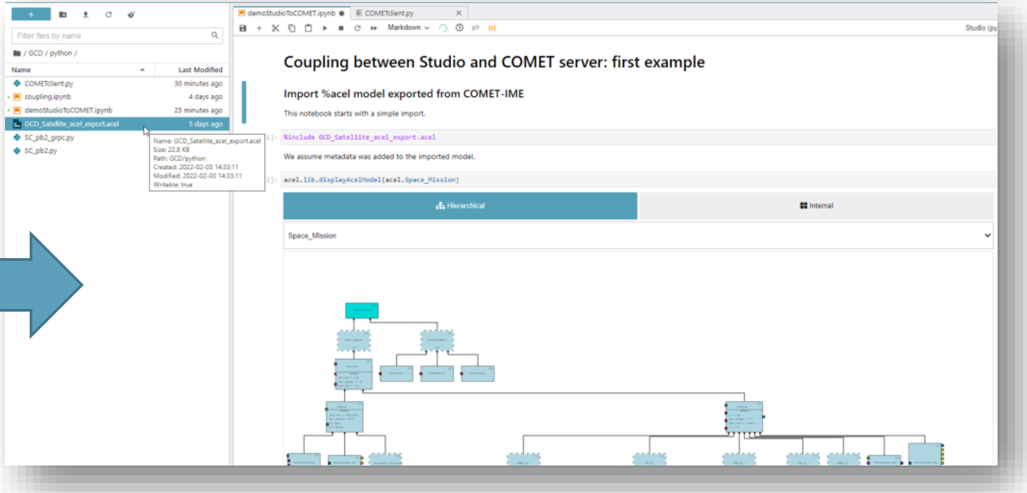
```
[2]: %acel

system Satellite:
    owns:
        AOCS
        Battery
        Communications
    metadata:
        uuid = "013d7e2b-d69b-4185-a962-6cced483b724"
        engineeringModelUuid = "694508eb-2730-488c-9405-6ca561df68dd"
        iterationUuid = "44647ff6-ffe3-44ff-9ed9-3256e2a97f9d"
        domainOfExpertiseUuid = "8790fe92-d1fa-42ea-9520-e0ddac52f1ad"

component AOCS:
    ports::group 1:
        out::flow out0
        (out)::flow out1
    attributes:
        float |kg| mass = 74.2 |kg|
    metadata:
        uuid = "0ce83c30-eb9f-4f74-b27d-f1b3cae78dfd"
        out0: uuid = "f47cd2fb-aa85-44d7-8899-d876151d626d"
        out1: uuid = "8e72d9c6-59f3-4853-99c4-53c2406540be"
        mass: uuid = "d7819466-1709-41ed-a34b-5a33854a1c36"

component Battery:
    ports::group 1:
```

%include: load model since read-only



Name	Last Modified
COMETclient.py	30 minutes ago
coupling.ipynb	4 days ago
demoStudioToCOMET.ipynb	25 minutes ago
GCD_Satellite_acel_export.acel	5 days ago
SC_pb2_grpc.py	
SC_pb2.py	

Name: GCD_Satellite_acel_export.acel
Size: 22.8 KB
Path: GCD/python
Created: 2022-02-03 14:33:11
Modified: 2022-02-03 14:33:11
Writable: true

Coupling between Studio and COMET server: first example

Import %acel model exported from COMET-IME

This notebook starts with a simple import.

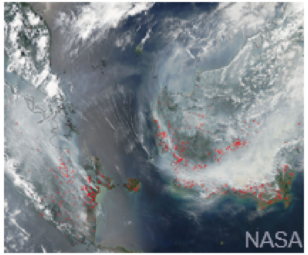
```
[1]: %include GCD_Satellite_acel_export.acel
```

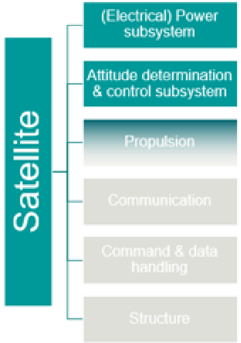
We assume metadata was added to the imported model.

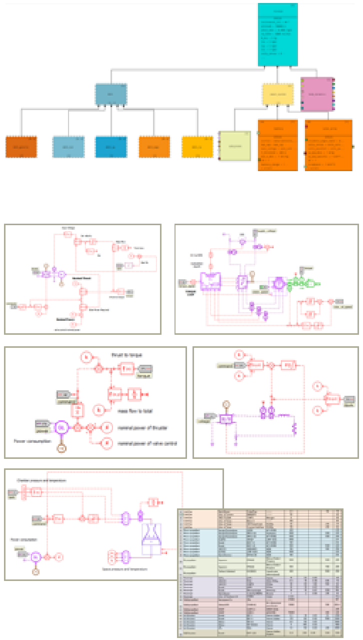
```
[2]: acel.lib.displayAcelModel(acel.Space_Mission)
```

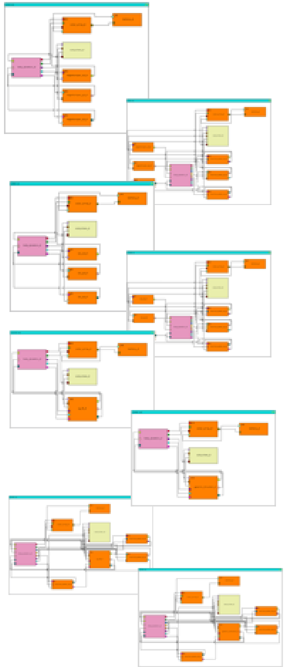
Simcenter Studio Features → Architecture Generation

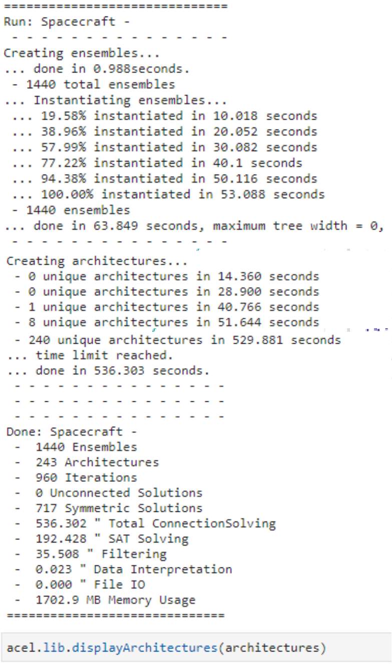
Mission design & objective → Subsystem definition, asset library → Architecture & configuration generation → Simulation & postprocessing → Attribute balancing & solution selection




Satellite
(Electrical) Power subsystem
Attitude determination & control subsystem
Propulsion
Communication
Command & data handling
Structure

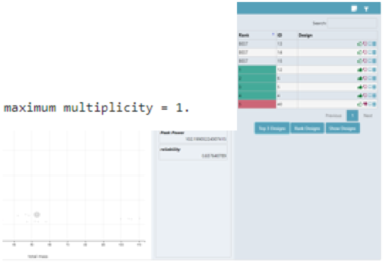




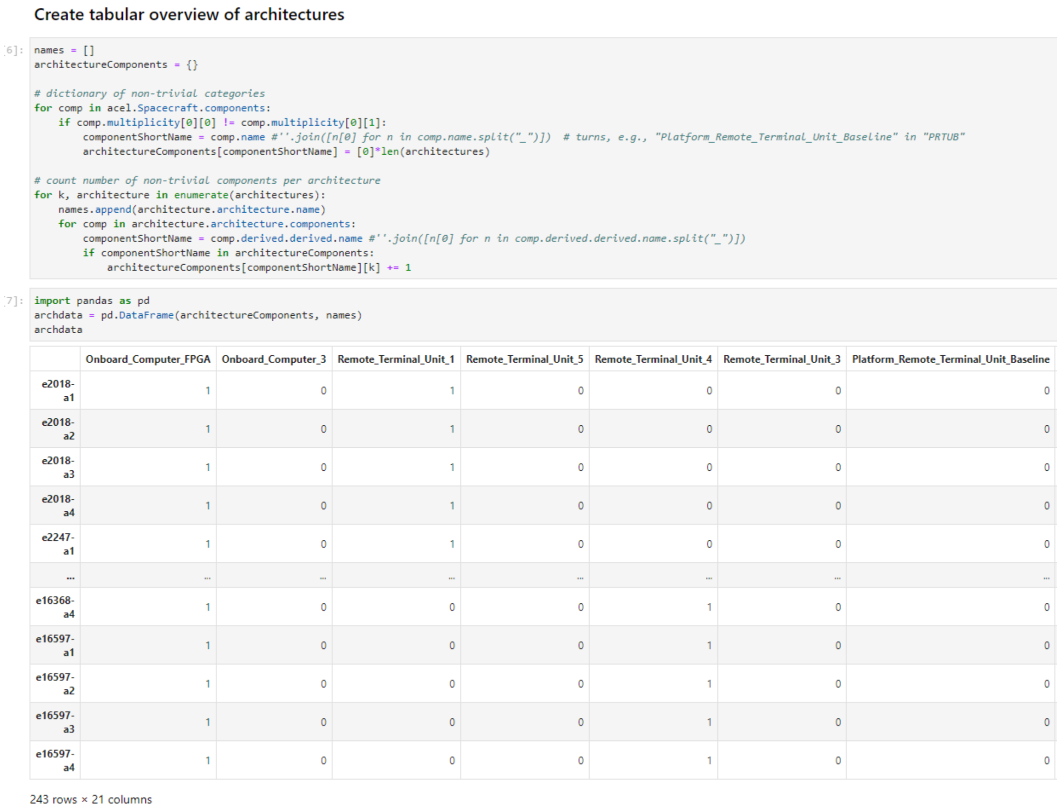


```
==============================
Run: Spacecraft -
 - - - - - - - - - - - - - - -
Creating ensembles...
... done in 0.988seconds.
 - 1440 total ensembles
... Instantiating ensembles...
 ... 19.58% instantiated in 10.018 seconds
 ... 38.96% instantiated in 20.052 seconds
 ... 57.99% instantiated in 30.082 seconds
 ... 77.22% instantiated in 40.1 seconds
 ... 94.38% instantiated in 50.116 seconds
 ... 100.00% instantiated in 53.088 seconds
 - 1440 ensembles
... done in 63.849 seconds, maximum tree width = 0, maximum multiplicity = 1.
 - - - - - - - - - - - - - - -
Creating architectures...
 - 0 unique architectures in 14.360 seconds
 - 0 unique architectures in 28.900 seconds
 - 1 unique architectures in 40.766 seconds
 - 8 unique architectures in 51.644 seconds
 - 240 unique architectures in 529.881 seconds
... time limit reached.
... done in 536.303 seconds.
 - - - - - - - - - - - - - - -
 - - - - - - - - - - - - - - -
 - - - - - - - - - - - - - - -
Done: Spacecraft -
 -  1440 Ensembles
 -  243 Architectures
 -  960 Iterations
 -  0 Unconnected Solutions
 -  717 Symmetric Solutions
 -  536.302 " Total ConnectionSolving
 -  192.428 " SAT Solving
 -  35.508 " Filtering
 -  0.023 " Data Interpretation
 -  0.000 " File IO
 -  1702.9 MB Memory Usage
==============================
```

```
acel.lib.displayArchitectures(architectures)
```



Simcenter Studio Features → Notebooks



Create tabular overview of architectures

[6]:

```python
names = []
architectureComponents = {}

# dictionary of non-trivial categories
for comp in acel.Spacecraft.components:
    if comp.multiplicity[0][0] != comp.multiplicity[0][1]:
        componentShortName = comp.name #''.join([n[0] for n in comp.name.split("_")])  # turns, e.g., "Platform_Remote_Terminal_Unit_Baseline" in "PRTUB"
        architectureComponents[componentShortName] = [0]*len(architectures)

# count number of non-trivial components per architecture
for k, architecture in enumerate(architectures):
    names.append(architecture.architecture.name)
    for comp in architecture.architecture.components:
        componentShortName = comp.derived.derived.name #''.join([n[0] for n in comp.derived.derived.name.split("_")])
        if componentShortName in architectureComponents:
            architectureComponents[componentShortName][k] += 1
```

[7]:

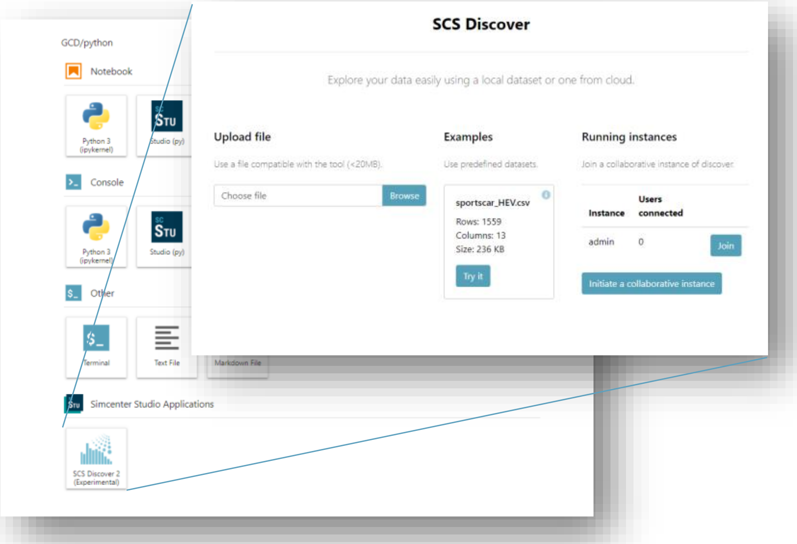
```python
import pandas as pd
archdata = pd.DataFrame(architectureComponents, names)
archdata
```

	Onboard_Computer_FPGA	Onboard_Computer_3	Remote_Terminal_Unit_1	Remote_Terminal_Unit_5	Remote_Terminal_Unit_4	Remote_Terminal_Unit_3	Platform_Remote_Terminal_Unit_Baseline
e2018-a1	1	0	1	0	0	0	0
e2018-a2	1	0	1	0	0	0	0
e2018-a3	1	0	1	0	0	0	0
e2018-a4	1	0	1	0	0	0	0
e2247-a1	1	0	1	0	0	0	0
...	...	...	...	...	...	...	...
e16368-a4	1	0	0	0	1	0	0
e16597-a1	1	0	0	0	1	0	0
e16597-a2	1	0	0	0	1	0	0
e16597-a3	1	0	0	0	1	0	0
e16597-a4	1	0	0	0	1	0	0

243 rows × 21 columns

Simcenter Studio Features → Discover/Multi-user Scoring

- Collaborative tradeoff and ranking
- Own likes/dislikes vs. other's input



SCS Discover

Explore your data easily using a local dataset or one from cloud.

Upload file

Use a file compatible with the tool (<20MB).

Choose file | Browse

Examples

Use predefined datasets.

sportscar_HEV.csv
Rows: 1559
Columns: 13
Size: 236 KB

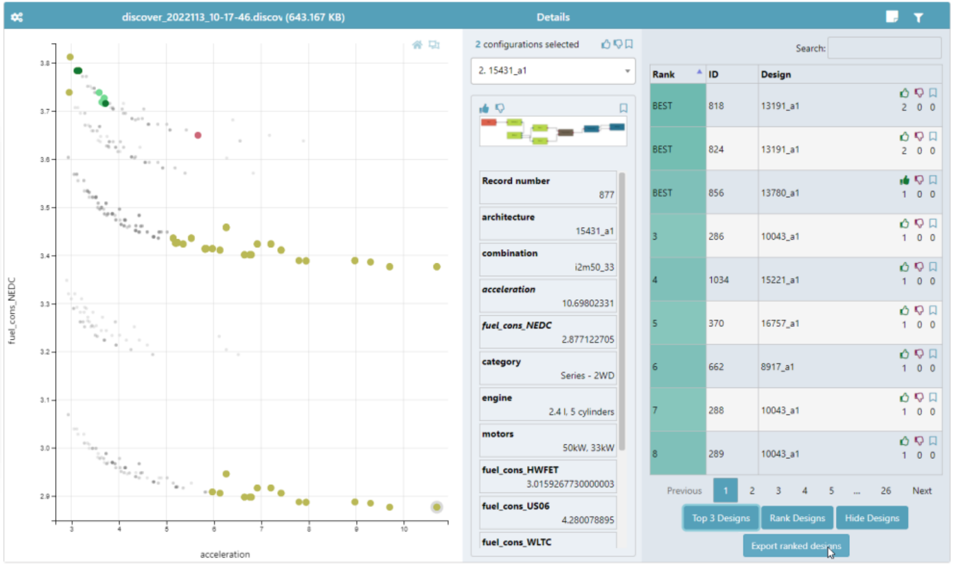
Try it

Running instances

Join a collaborative instance of discover.

Instance	Users connected	
admin	0	Join

Initiate a collaborative instance



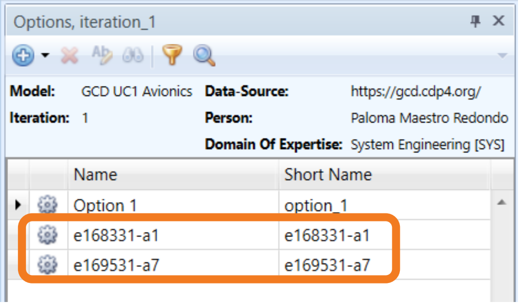
2 configurations selected

2. 15431_a1

Record number	877
architecture	15431_a1
combination	i2m50_33
acceleration	10.69802331
fuel_cons_NEDC	2.877122705
category	Series - 2WD
engine	2.4 l, 5 cylinders
motors	50kW, 33kW
fuel_cons_HWFET	3.0159267730000003
fuel_cons_US06	4.280078895
fuel_cons_WLTC	

Rank	ID	Design			
BEST	818	13191_a1	2	0	0
BEST	824	13191_a1	2	0	0
BEST	856	13780_a1	1	0	0
3	286	10043_a1	1	0	0
4	1034	15221_a1	1	0	0
5	370	16757_a1	1	0	0
6	662	8917_a1	1	0	0
7	288	10043_a1	1	0	0
8	289	10043_a1	1	0	0

Simcenter Studio Features → Transfer back



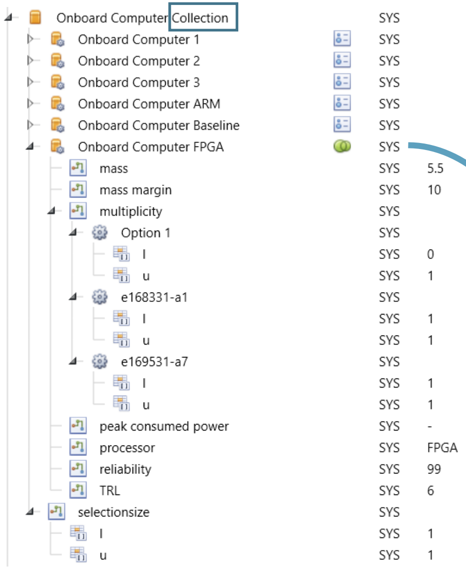
Options, iteration_1

Model: GCD UC1 Avionics **Data-Source:** https://gcd.cdp4.org/

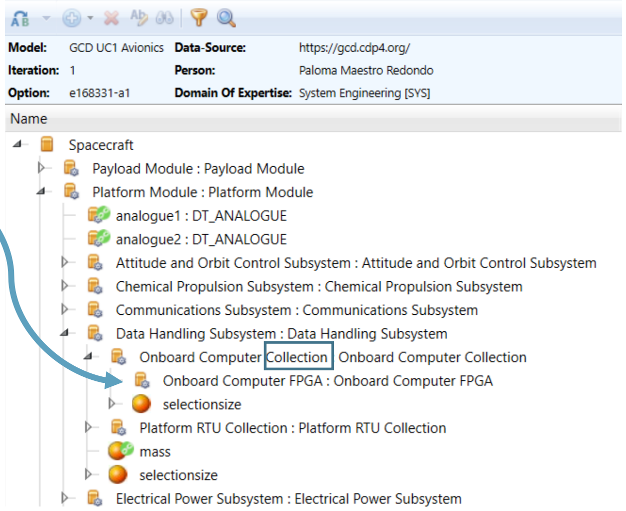
Iteration: 1 **Person:** Paloma Maestro Redondo

Domain Of Expertise: System Engineering [SYS]

Name	Short Name
Option 1	option_1
e168331-a1	e168331-a1
e169531-a7	e169531-a7



Onboard Computer Collection	SYS	
Onboard Computer 1	SYS	
Onboard Computer 2	SYS	
Onboard Computer 3	SYS	
Onboard Computer ARM	SYS	
Onboard Computer Baseline	SYS	
Onboard Computer FPGA	SYS	
mass	SYS	5.5
mass margin	SYS	10
multiplicity	SYS	
Option 1	SYS	
l	SYS	0
u	SYS	1
e168331-a1	SYS	
l	SYS	1
u	SYS	1
e169531-a7	SYS	
l	SYS	1
u	SYS	1
peak consumed power	SYS	-
processor	SYS	FPGA
reliability	SYS	99
TRL	SYS	6
selectionsize	SYS	
l	SYS	1
u	SYS	1



Model: GCD UC1 Avionics **Data-Source:** https://gcd.cdp4.org/

Iteration: 1 **Person:** Paloma Maestro Redondo

Option: e168331-a1 **Domain Of Expertise:** System Engineering [SYS]

Name

- Spacecraft
  - Payload Module : Payload Module
  - Platform Module : Platform Module
    - analogue1 : DT_ANALOGUE
    - analogue2 : DT_ANALOGUE
    - Attitude and Orbit Control Subsystem : Attitude and Orbit Control Subsystem
    - Chemical Propulsion Subsystem : Chemical Propulsion Subsystem
    - Communications Subsystem : Communications Subsystem
    - Data Handling Subsystem : Data Handling Subsystem
      - Onboard Computer Collection : Onboard Computer Collection
        - Onboard Computer FPGA : Onboard Computer FPGA
        - selectionsize
      - Platform RTU Collection : Platform RTU Collection
      - mass
      - selectionsize
    - Electrical Power Subsystem : Electrical Power Subsystem

Validation

Test Case Definition

ID	Test Case	Use Case
#1	Subsystem Test Case #1	UC#1: Avionics & Electrical Architecture (reduced to Avionics S/S)
#2	Subsystem Test Case #2	UC#2: Propulsion Architecture (EP vs. CP) (reduced to Propulsion S/S)
#3	System Test Case #1	UC#3: Data Return / COM Architecture
#4	System Test Case #2	UC#4: Configuration & Assembly
#5	System of System Test Case #1	UC#5: Earth Observation Optical System - Mission Architecture & Orbit selection
#6	System of System Test Case #2	UC#6: Copernicus Next Generation Architecture

Test Setup

- E-TM-10-25 Server
  - GCD specific RDL, test case Engineering Models and Catalogue
- Multiple instances of COMET IME for concurrent work

- Simcenter Studio Server
  - One/Multiple users for generative aspects

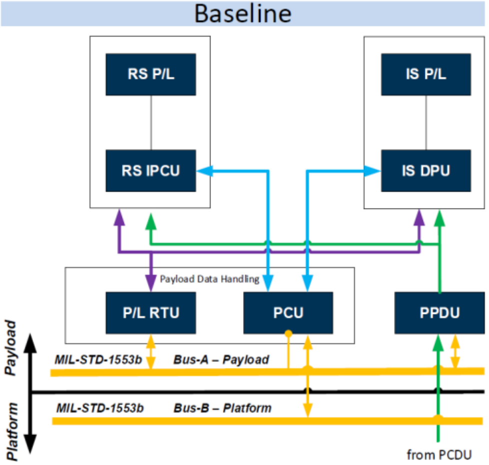
- Modelling level of detail reduced to the necessary to support validation
- For un-supported DSTs: replacement with equations/scripts with reduced fidelity
- All functions tested at least in one test for each level (S/S, Sys, SoS)

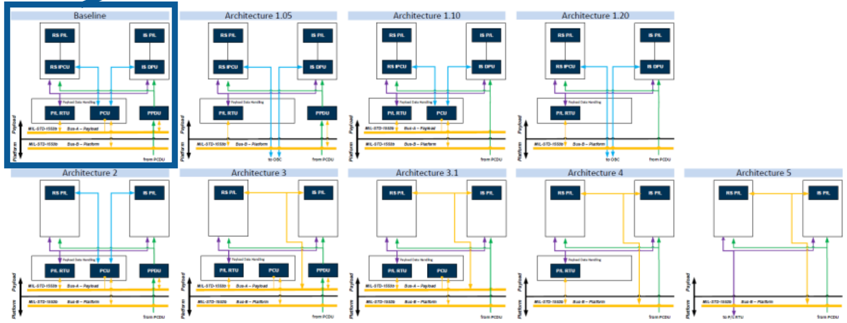
Use Case Example: UC#1 Avionics Subsystem Use Case

Multiple architectures possible on payload interface level or data handling level (e.g., SMU vs OBC + MMU)

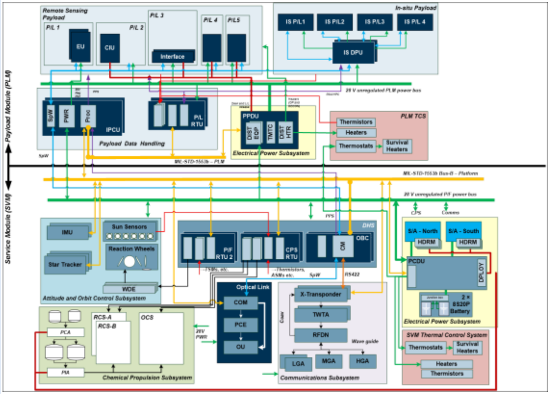
Unit selection based on hardware matrix to represent architectures with components from different supplier

- Typical DH subsystem requirements allow to cross-check with architecture
  - e.g., The total mass, including all margins of the DHS shall not exceed 28 kg.
- Parameters are given based on evaluation criteria and key interest to conclude trade-off

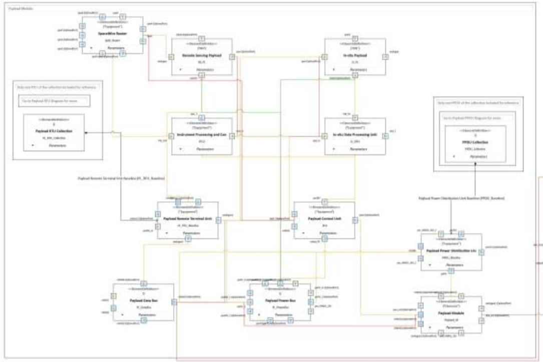


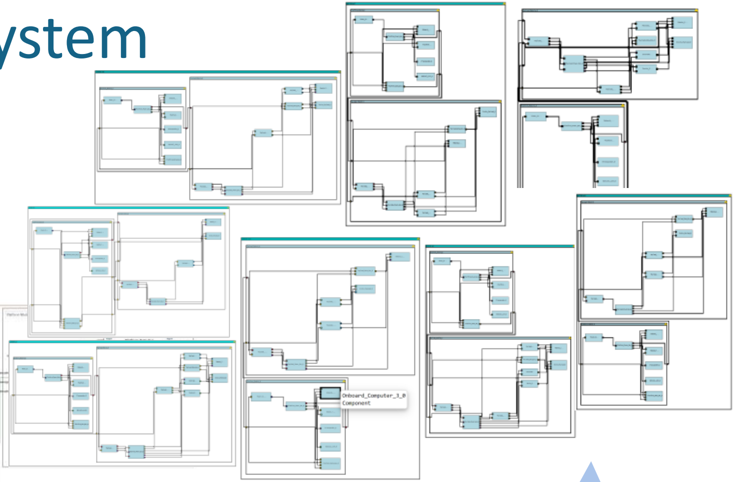


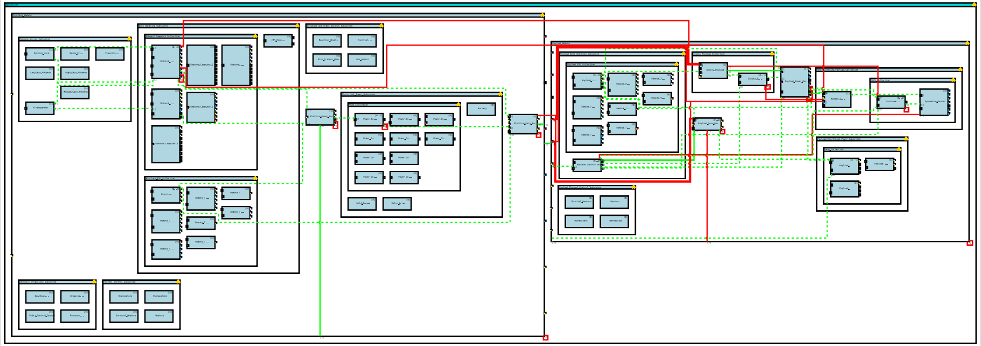
Use Case Example: UC#1 Avionics Subsystem



Alternatives	Remote Sensing IPCU and In-situ DPU	Connector for PL to specific control unit	PPDU	PCU	PL Mil bus	PL RTU
Architecture 1 Baseline	X	SpW	X	X	X	X
Architecture 1.05 Removed PCU only	X	SpW	X			X
Architecture 1.10 Removed PPDU	X	SpW		X	X	X
Architecture 1.20 Removed PPDU and PCU	X	SpW				X
Architecture 2 Removed IPCU/DPU, with SpW		SpW (RS) and CAN (IS)	X	X	X	X
Architecture 3 Removed IPCU/DPU, with PL MilBus		PL MilBus and SpW	X	X	X	X
Architecture 3.10 Removed IPCU/DPU and PPDU, with PL MilBus		PL MilBus and SpW		X	X	X
Architecture 4 Removed IPCU/DPU, PCU, PPDU and PL MilBus		PF MilBus and SpW				X
Architecture 5 Removed IPCU/DPU, PCU, PPDU, PL RTU and PL MilBus		PF MilBus and SpW				

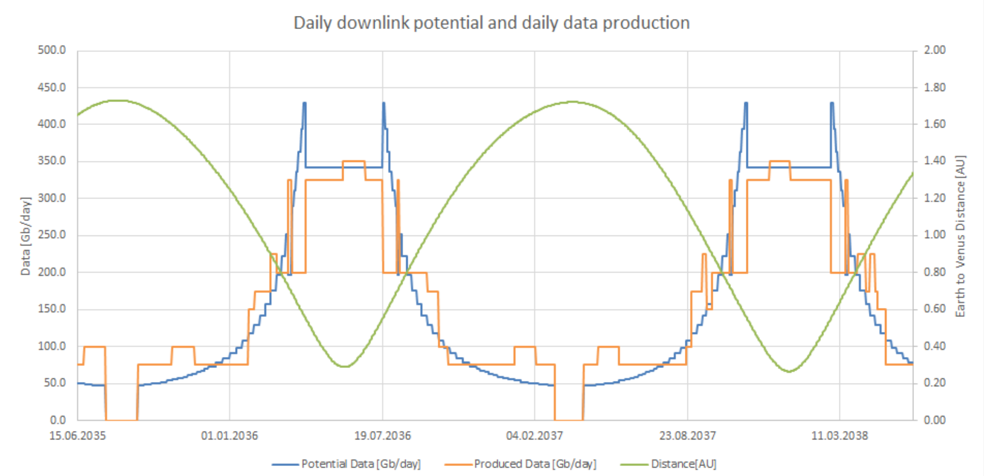

Payload Module
Remote Sensing Payload
In-situ Payload
Instrument Processing and Control Unit
In-situ Data Processing Unit
Payload Remote Terminal Unit
Payload Control Unit
Payload Power Distribution Unit
Payload Data Bus
Payload Power Bus
Payload Module

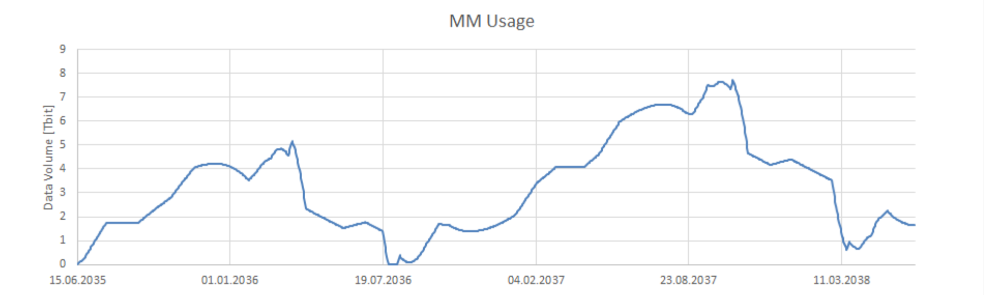




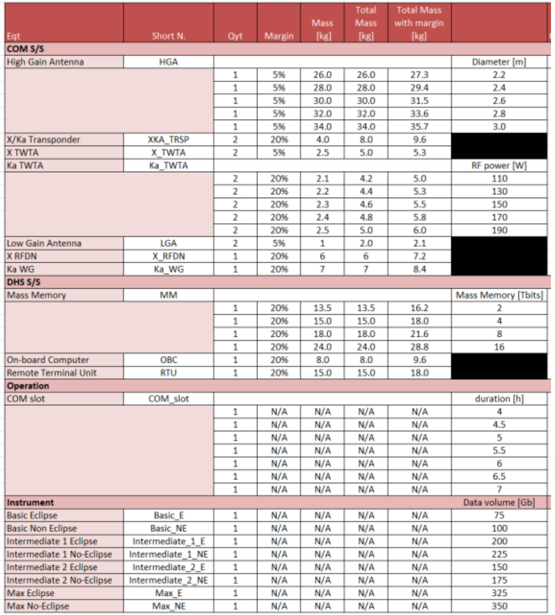
Use Case Example: UC#3 Data Return/Com Architecture System Use Case

- Data return trade-off based on science observation profile and communication step profile
- Analysis performed for different variables
  - HGA diameter 2.2m to 3.0m
  - TWTA RF power 110W to 190W
  - Communication slot duration 4h to 7h
- Investigate impact on Mass Memory

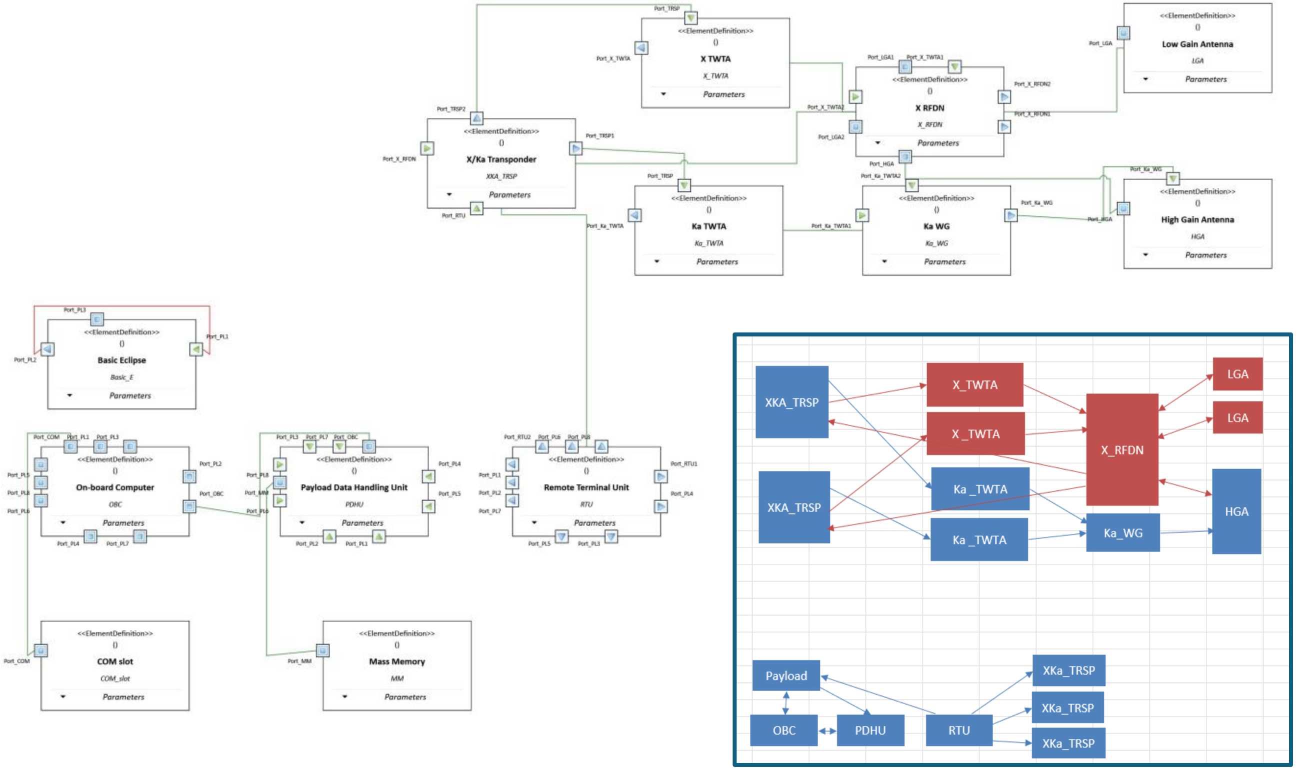




Use Case Example: UC#3 Data Return/Com Architecture System



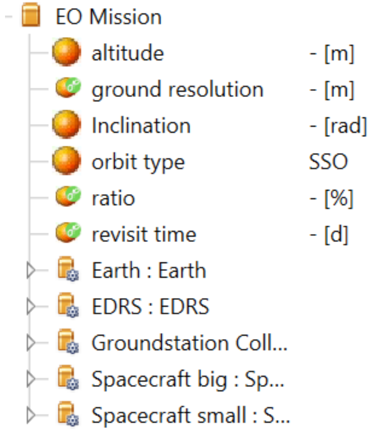
Eqt	Short N.	Qyt	Margin	Mass [kg]	Total Mass [kg]	Total Mass with margin [kg]	
COM S/S							
High Gain Antenna	HGA						Diameter [m]
		1	5%	26.0	26.0	27.3	2.2
		1	5%	28.0	28.0	29.4	2.4
		1	5%	30.0	30.0	31.5	2.6
		1	5%	32.0	32.0	33.6	2.8
		1	5%	34.0	34.0	35.7	3.0
X/Ka Transponder	XKA_TRSP	2	20%	4.0	8.0	9.6	[redacted]
X TWTA	X_TWTA	2	5%	2.5	5.0	5.3	[redacted]
Ka TWTA	Ka_TWTA						RF power [W]
		2	20%	2.1	4.2	5.0	110
		2	20%	2.2	4.4	5.3	130
		2	20%	2.3	4.6	5.5	150
		2	20%	2.4	4.8	5.8	170
		2	20%	2.5	5.0	6.0	190
Low Gain Antenna	LGA	2	5%	1	2.0	2.1	[redacted]
X RFDN	X_RFDN	1	20%	6	6	7.2	[redacted]
Ka WG	Ka_WG	1	20%	7	7	8.4	[redacted]
DHS S/S							
Mass Memory	MM						Mass Memory [Tbits]
		1	20%	13.5	13.5	16.2	2
		1	20%	15.0	15.0	18.0	4
		1	20%	18.0	18.0	21.6	8
		1	20%	24.0	24.0	28.8	16
On-board Computer	OBC	1	20%	8.0	8.0	9.6	[redacted]
Remote Terminal Unit	RTU	1	20%	15.0	15.0	18.0	[redacted]
Operation							
COM slot	COM_slot						duration [h]
		1	N/A	N/A	N/A	N/A	4
		1	N/A	N/A	N/A	N/A	4.5
		1	N/A	N/A	N/A	N/A	5
		1	N/A	N/A	N/A	N/A	5.5
		1	N/A	N/A	N/A	N/A	6
		1	N/A	N/A	N/A	N/A	6.5
		1	N/A	N/A	N/A	N/A	7
Instrument							Data volume [Gb]
Basic Eclipse	Basic_E	1	N/A	N/A	N/A	N/A	75
Basic Non Eclipse	Basic_NE	1	N/A	N/A	N/A	N/A	100
Intermediate 1 Eclipse	Intermediate_1_E	1	N/A	N/A	N/A	N/A	200
Intermediate 1 No-Eclipse	Intermediate_1_NE	1	N/A	N/A	N/A	N/A	225
Intermediate 2 Eclipse	Intermediate_2_E	1	N/A	N/A	N/A	N/A	150
Intermediate 2 No-Eclipse	Intermediate_2_NE	1	N/A	N/A	N/A	N/A	175
Max Eclipse	Max_E	1	N/A	N/A	N/A	N/A	325
Max No-Eclipse	Max_NE	1	N/A	N/A	N/A	N/A	350



Use Case Example: UC #5 EO Mission Architecture & Orbit Selection Use Case

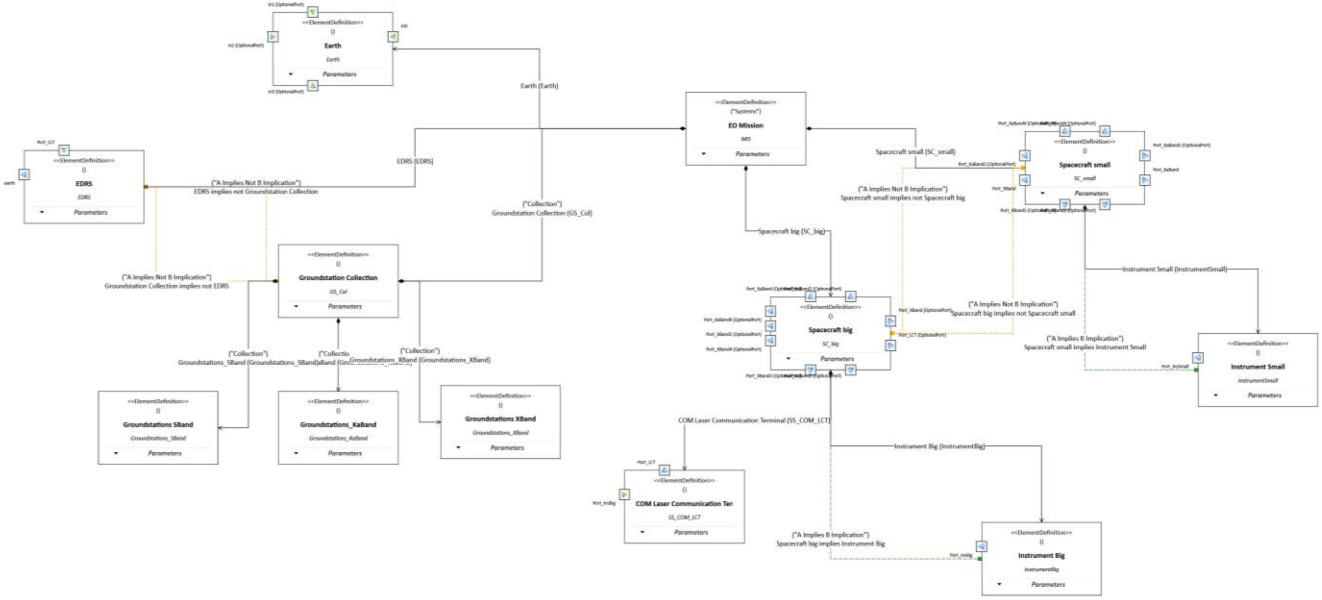
- Trade-off for Mission Architecture (Space & Ground Segment) incl. Orbit selection
- Objective: find the optimal solution for full-earth coverage with a certain resolution & repetition time and minimal GS usage
- Orbit: SSO with different altitudes
- Space Segment:
  - one big vs. Several identical small satellites (constellation)
  - variants on Instrument design
  - Variants on Communication Subsystem (incl. EDRS for big S/C)
- Ground Segment: different ground stations incl. Available Communication Bands

Use Case Example: UC #5 EO Mission Architecture & Orbit Selection Use Case



EO Mission

altitude	- [m]
ground resolution	- [m]
Inclination	- [rad]
orbit type	SSO
ratio	- [%]
revisit time	- [d]
Earth : Earth	
EDRS : EDRS	
Groundstation Coll...	
Spacecraft big : Sp...	
Spacecraft small : S...	



Use Case Outcomes

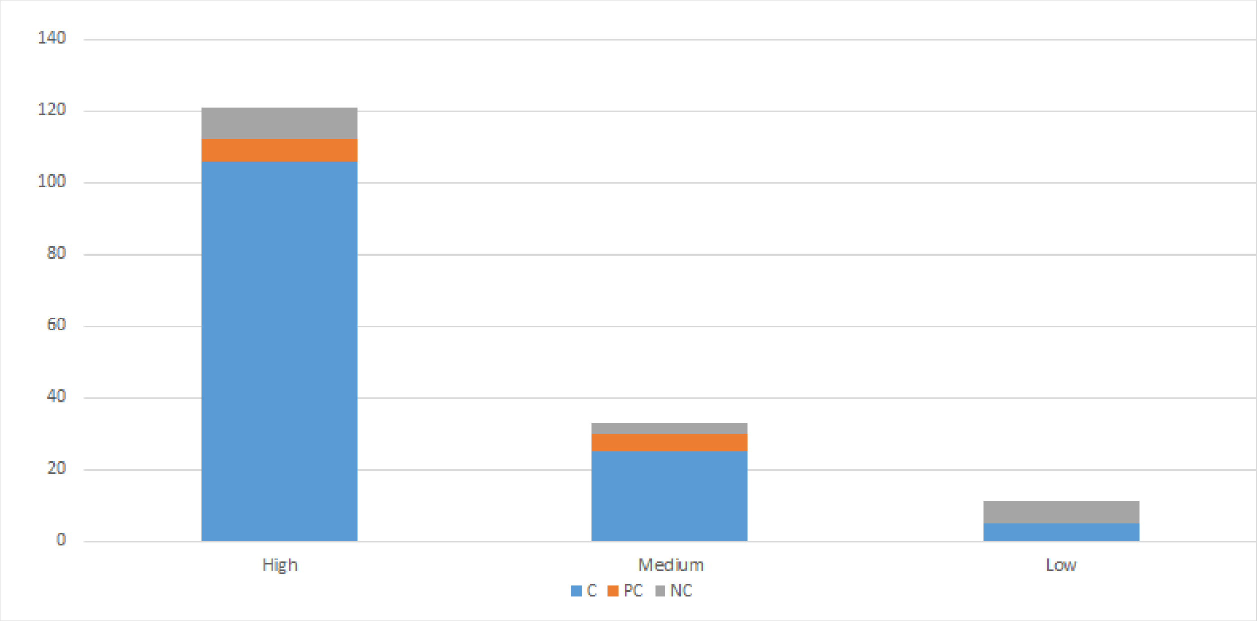
- System Requirements validated based on 3 system level use cases: subsystem, system and system of systems.
- One of each used to validate the requirements on all levels + 1 extra usecase per level modeled and provided as part of the deliverables.
- Results also used to improve the software during the study.
- Process used to define the methodology necessary to perform GCD.

Software Validation

Validation Metrics

SSS contains 165 Req.
136 C
11 PC
18 NC
0 N/A

- All NC and PC have associated RFW or RFD documented in the SValR deliverable.



CEFO Workshop

CEFO Workshop

Dates: 7th-8th June

In person in Bremen

5 People from OHB side, RHEA & Siemens present

1 Person from ESA

Initial training for new COMET features & Studio shall be part of workshop

Access for externals to OHB Servers (COMET & Simcenter Studio) through dedicated laptops --> direct collaboration of every participant

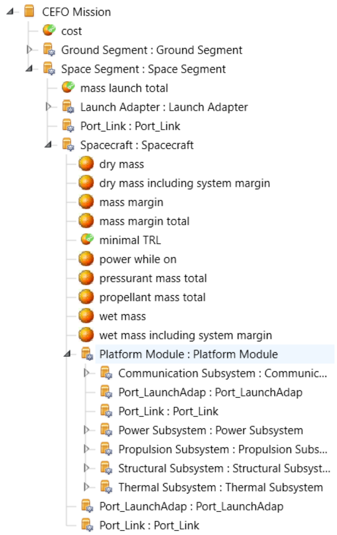
Extension of Use Case 3 with additional subsystem aspects

Propulsion, Thermal, Power, Structure, Mission, Programmatics

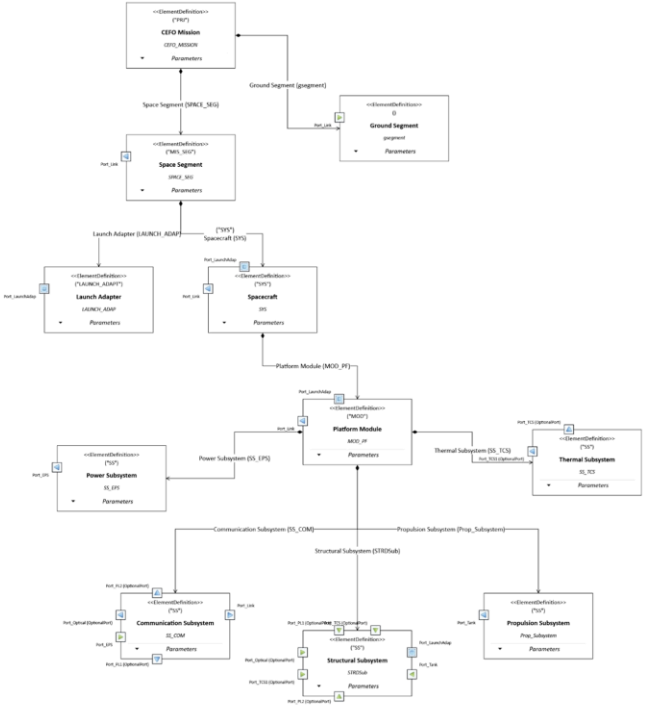
CEFO Workshop Modelling

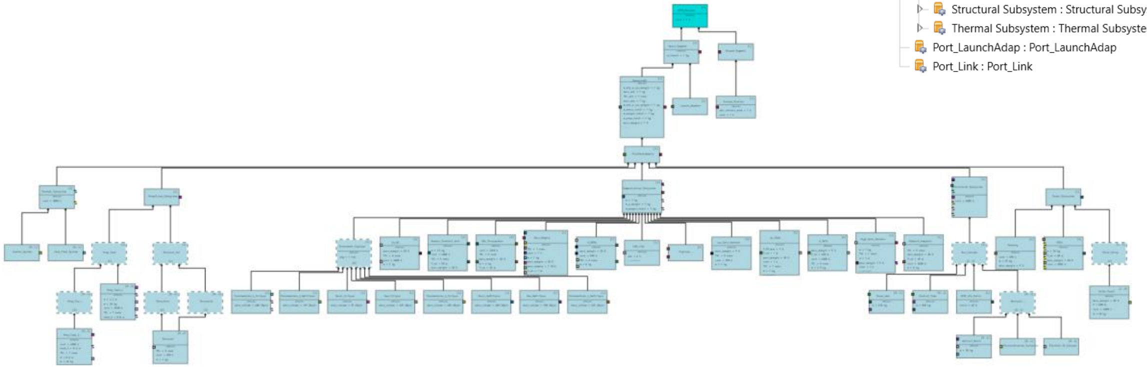
Extension to UC3:
5 subsystems + ground system

12 fundamental architectures
16,800 configurations



- CEFO Mission
  - cost
  - Ground Segment : Ground Segment
  - Space Segment : Space Segment
    - mass launch total
    - Launch Adapter : Launch Adapter
    - Port_Link : Port_Link
    - Spacecraft : Spacecraft
      - dry mass
      - dry mass including system margin
      - mass margin
      - mass margin total
      - minimal TRL
      - power while on
      - pressurant mass total
      - propellant mass total
      - wet mass
      - wet mass including system margin
      - Platform Module : Platform Module
        - Communication Subsystem : Communic...
        - Port_LaunchAdap : Port_LaunchAdap
        - Port_Link : Port_Link
        - Power Subsystem : Power Subsystem
        - Propulsion Subsystem : Propulsion Subs...
        - Structural Subsystem : Structural Subsyst...
        - Thermal Subsystem : Thermal Subsystem
      - Port_LaunchAdap : Port_LaunchAdap
      - Port_Link : Port_Link



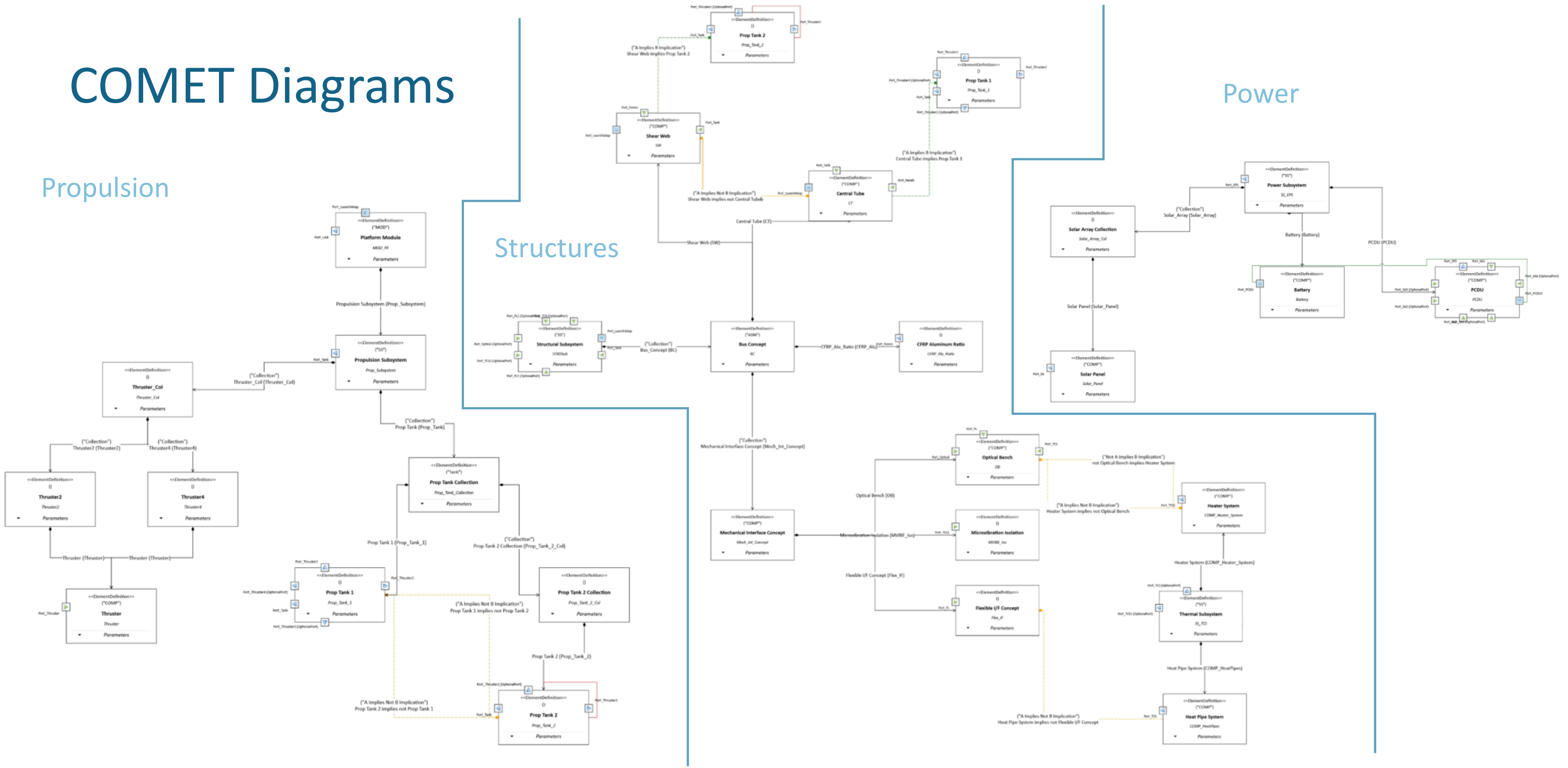


COMET Diagrams

Propulsion

Structures

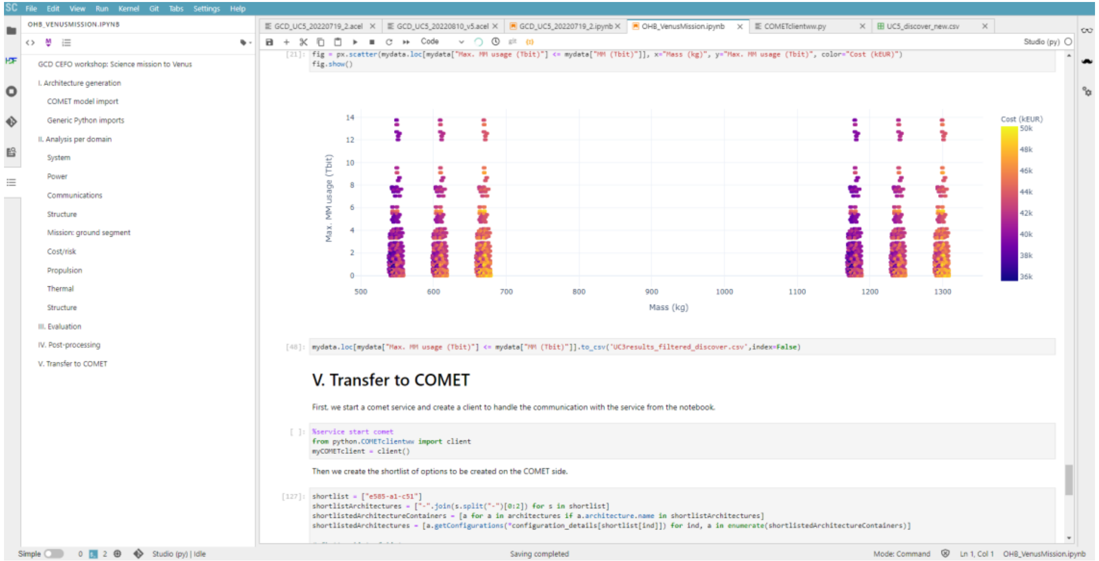
Power



Prop Tank 2
("A Implies B Implication")
Shear Web implies Prop Tank 2
Prop Tank 1
Shear Web
("A Implies B Implication")
Central Tube implies Prop Tank 1
("A Implies Not B Implication")
Shear Web implies not Central Tube
Central Tube
Central Tube (CT)
Shear Web (SW)
Platform Module
Propulsion Subsystem (Prop_Subsystem)
Propulsion Subsystem
("Collection")
Thruster_Col (Thruster_Col)
Thruster_Col
("Collection")
Thruster2 (Thruster2)
("Collection")
Thruster4 (Thruster4)
Thruster2
Thruster4
Thruster (Thruster)
Thruster (Thruster)
Thruster
("Collection")
Prop Tank (Prop_Tank)
Prop Tank Collection
Prop Tank 1 (Prop_Tank_1)
("Collection")
Prop Tank 2 Collection (Prop_Tank_2_Col)
Prop Tank 1
("A Implies Not B Implication")
Prop Tank 1 implies not Prop Tank 2
Prop Tank 2 Collection
Prop Tank 2 (Prop_Tank_2)
("A Implies Not B Implication")
Prop Tank 2 implies not Prop Tank 1
Prop Tank 2
Structural Subsystem
("Collection")
Bus_Concept (BC)
Bus Concept
CFRP_Alu_Ratio (CFRP_Alu)
CFRP Aluminum Ratio
("Collection")
Mechanical Interface Concept (Mech_Int_Concept)
Mechanical Interface Concept
Optical Bench (OB)
Microvibration Isolation (MVIBE_Iso)
Flexible I/F Concept (Flex_IF)
Optical Bench
("Not A Implies B Implication")
not Optical Bench implies Heater System
("A Implies Not B Implication")
Heater System implies not Optical Bench
Heater System
Microvibration Isolation
Heater System (COMP_Heater_System)
Flexible I/F Concept
Thermal Subsystem
Heat Pipe System (COMP_HeatPipes)
("A Implies Not B Implication")
Heat Pipe System implies not Flexible I/F Concept
Heat Pipe System
Power Subsystem
("Collection")
Solar_Array (Solar_Array)
Solar Array Collection
Battery (Battery)
PCDU (PCDU)
Battery
PCDU
Solar Panel (Solar_Panel)
Solar Panel

CEFO Workshop Modelling

- Worked out notebook structure that combines analysis per domain: Might be base of reusable template for GCD-specific design procedure
- Discussion on embedding within CDF session workflow



GCD CEFO workshop: Science mission to Venus

I. Architecture generation

COMET model import

Generic Python imports

II. Analysis per domain

System

Power

Communications

Structure

Mission: ground segment

Cost/risk

Propulsion

Thermal

Structure

III. Evaluation

IV. Post-processing

V. Transfer to COMET

```
[21]: fig = px.scatter(mydata.loc[mydata["Max. MM usage (Tbit)"] <= mydata["MM (Tbit)"]], x="Mass (kg)", y="Max. MM usage (Tbit)", color="Cost (kEUR)")
      fig.show()
```

```
[48]: mydata.loc[mydata["Max. MM usage (Tbit)"] <= mydata["MM (Tbit)"]].to_csv('UC3results_filtered_discover.csv',index=False)
```

V. Transfer to COMET

First, we start a comet service and create a client to handle the communication with the service from the notebook.

```
[ ]: %service start comet
     from python.COMETclientww import client
     myCOMETclient = client()
```

Then we create the shortlist of options to be created on the COMET side.

```
[127]: shortlist = ["e585-a1-c51"]
       shortlistArchitectures = ["-".join(s.split("-")[0:2]) for s in shortlist]
       shortlistedArchitectureContainers = [a for a in architectures if a.architecture.name in shortlistArchitectures]
       shortlistedArchitectures = [a.getConfigurations(*configuration_details[shortlist[ind]]) for ind, a in enumerate(shortlistedArchitectureContainers)]
```

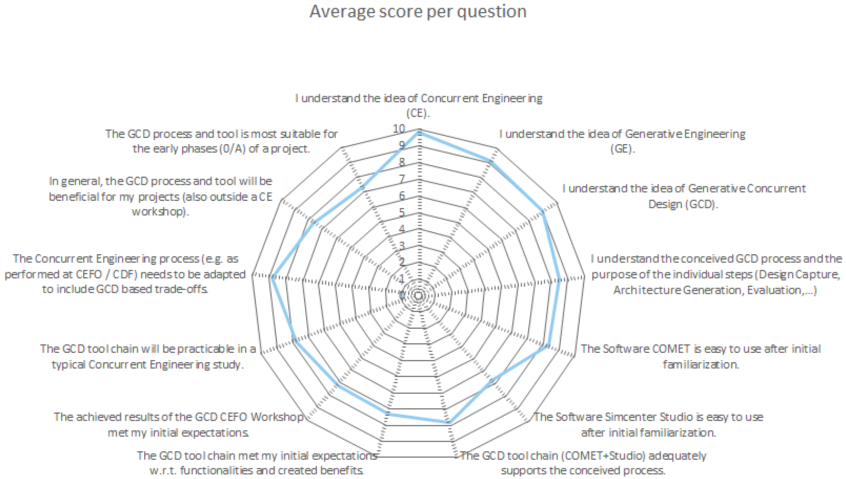
Acel model + notebook with table-of-contents view

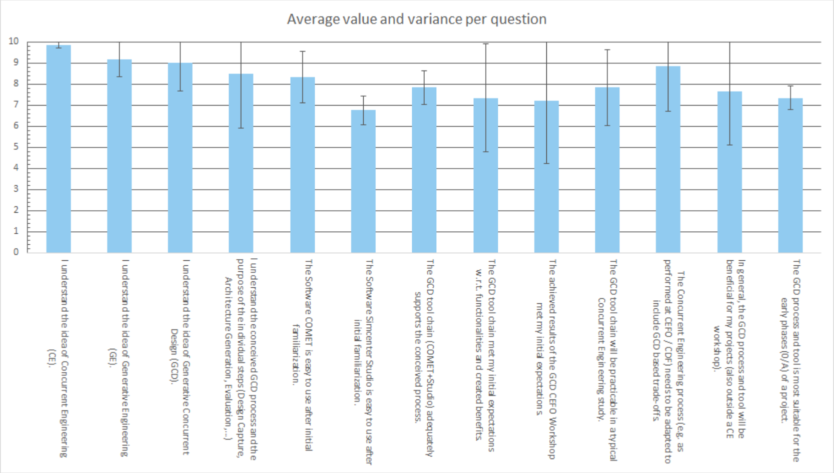
CEFO Workshop – Results / Findings

- Further awareness and understanding for GCD purpose and capabilities created for new users
- Basic steps for Design Capture in COMET was possible with very limited training --> more support required for the details and particular constraints
- Schedule was too ambitious (two days only for entire process incl. Training)
  - Additional session for evaluation / scoring part was conducted remotely on 23.06.
- Simcenter Studio was not actively used by all participants but controlled centrally by Siemens
- Ad-hoc model fixes where not possible during the CEFO Workshop
- Some IT issues uncovered still during the Workshop (COMET-->Simcenter)

CEFO Workshop – Results / Findings

Focus for this workshop was not to create a perfect technical result but collect additional user feedback as input for ER document --> questionnaire distributed to all participants





Conclusions

Conclusions

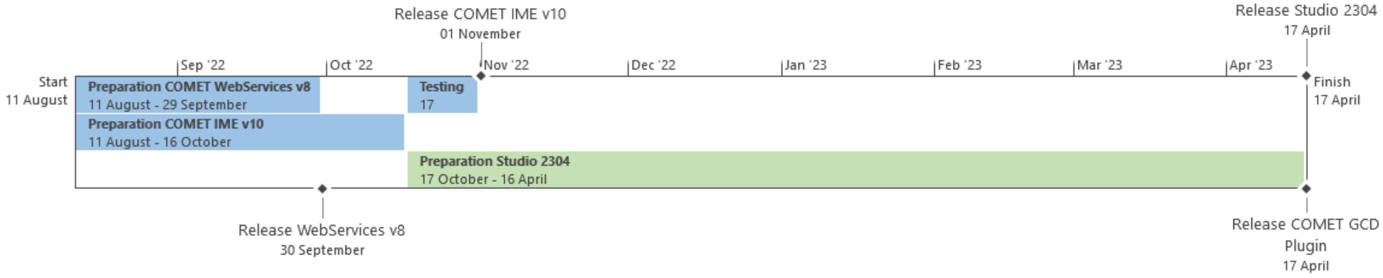
- Methodology and CONOPS defined and tested with applicable use cases at different architecture levels
- Improved functionalities, interfaces and new features developed for both tools which allow for a smooth workflow
- New diagramming capabilities implement in COMET and support for scripting behaviors
- Multi-user evaluation implemented in Simcenter Studio → Discover
- Direct link between COMET and Simcenter Studio
- Power of Python Notebooks for evaluation has been demonstrated, but integration/link with other DST is out of scope
- Potential Follow-up activities are being explored
  - Direct application of GCD to ongoing and future missions
  - Extension of features and further automation
  - More integrated link to Simulation tools during architecture option evaluation

Evaluation & Roadmap

- General understanding and expected benefits confirmed widely
- Big advantage for collaborative design capture by Architecture Diagrams in COMET compared to individually / script-based in Studio --> consistency / accessibility
- GCD will have impact on the way we are doing CDF/CEFO activities
- Correct modelling to be compatible with the Studio solver is not trivial and might even impact the usual model structure
- Transfer from COMET to Studio has big advantage, but more detailed debugging / validation / integrity checks would be appreciated
- Templates / supporting functions in Studio for non-expert users are needed

Roadmap - Immediate

- Preparation of integration of GCD developments into v10 of COMET and performance updates – up to November 2022
- Preparation of integration of GCD developments into v2304 – up to April 2023
- Release GCD Plugin for IME – April 2023



Release COMET IME v10
01 November

Release Studio 2304
17 April

Sep '22
Oct '22
Nov '22
Dec '22
Jan '23
Feb '23
Mar '23
Apr '23

Start
11 August

Preparation COMET WebServices v8
11 August - 29 September

Testing
17

Preparation COMET IME v10
11 August - 16 October

Preparation Studio 2304
17 October - 16 April

Finish
17 April

Release WebServices v8
30 September

Release COMET GCD Plugin
17 April

Roadmap - Future

- COMET
  - Diagramming – real-time collaboration and more diagram types ~2023
  - Behaviors – behavior evaluation in COMET and snippets ~2023-2024
  - GCD – continue improving acel validation and development of training material ~2023

- Studio
  - Usability – model debugging and traceability, collaborative workbooks and inspector improvements ~2023-2025
  - Simulation and optimization – parameter optimization directly in acel, link to other 3rd party simulators ~2022-2025
  - Security – docker architecture refactoring ~2023

Reach The Team

We love questions and feedback – and we're always happy to help!
Here are some ways to contact us.

ESA Contact

GCD ESA Technical Officer

Claire.Parfitt@esa.int

GCD Team

Alex Vorobiev (RHEA)	a.vorobiev@rheagroup.com
Stephan Jahnke (OHB)	stephan.jahnke@ohb.de
Jonathan Menu (Siemens Industry Software)	jonathan.menu@siemens.com

RHEA

GROUP