



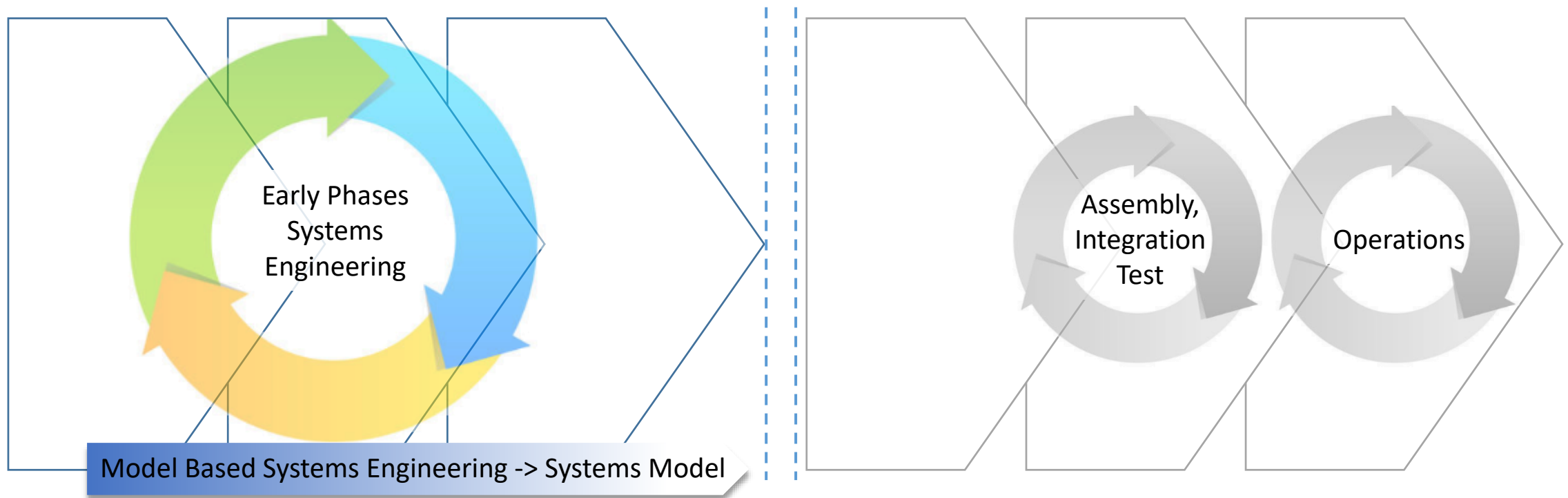
# IMAP

## The Integrated MBSE Analytics Platform

*Tracing AI based analysis of AIT/AIV telemetry data results to MBSE models*

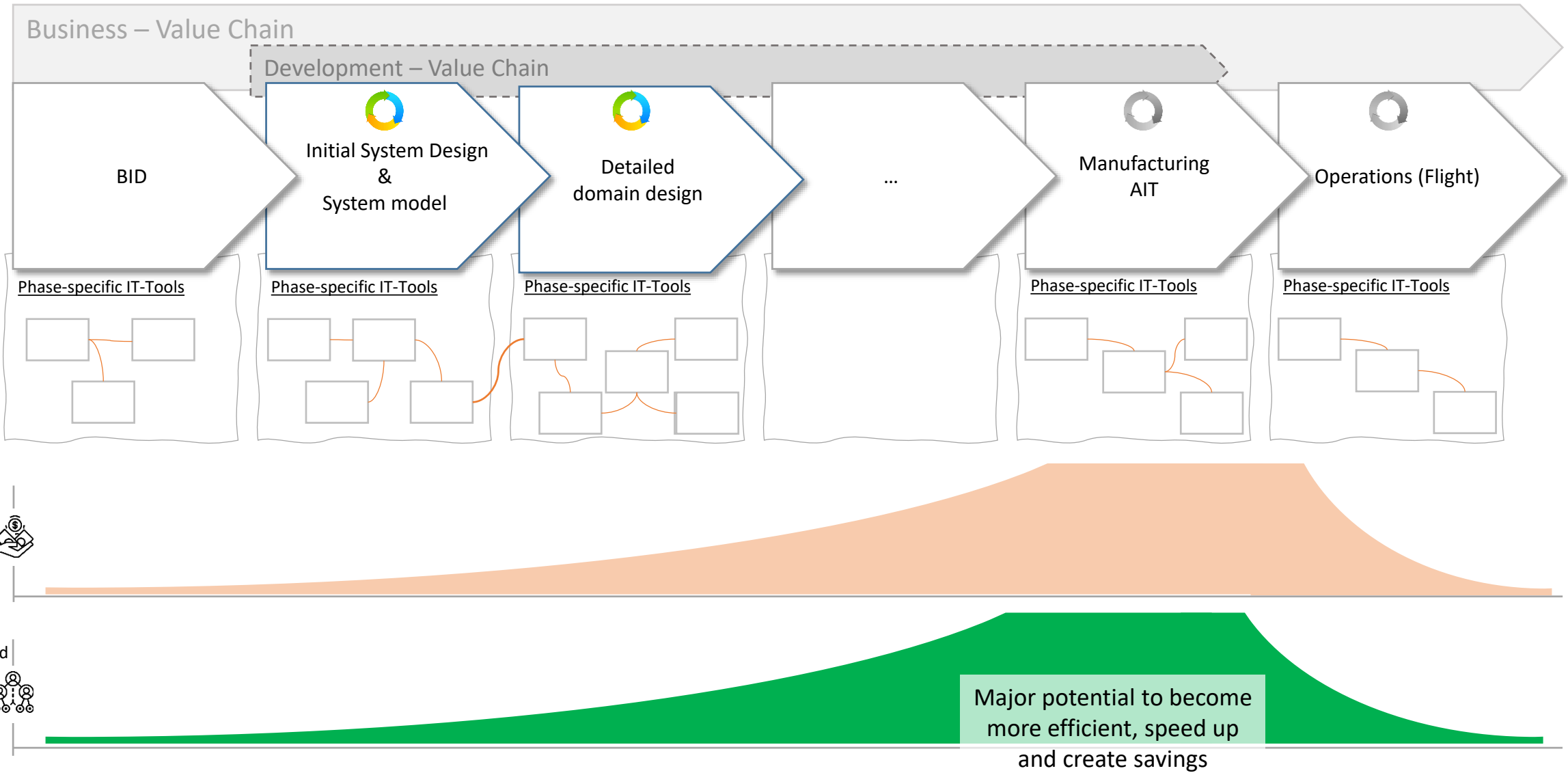
November 22, 2022, ScopeSET

# Where we are today – State of the Art in MBSE use

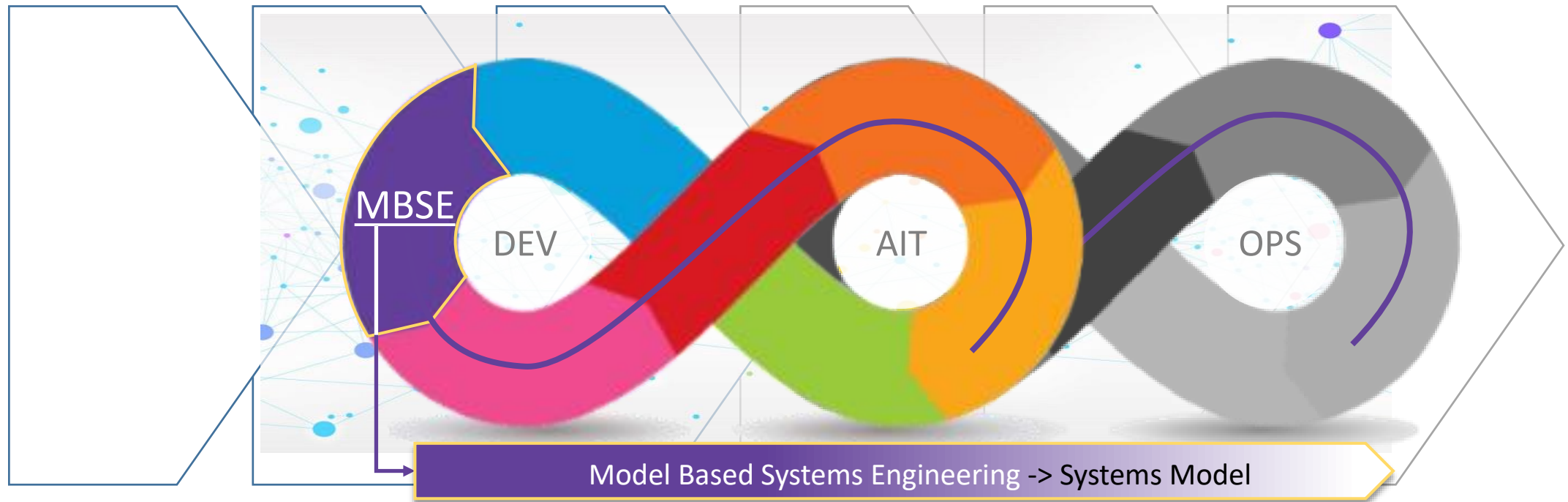


- Disconnected Data Silos and Teams along the lifecycle, mainly Point-to-Point interface
- Interconnected knowledge still in experts brains mostly
- Benefits of MBSE (System Models) do not pay out in later phases
  - Models become outdated and are no longer used in AIT and Operations

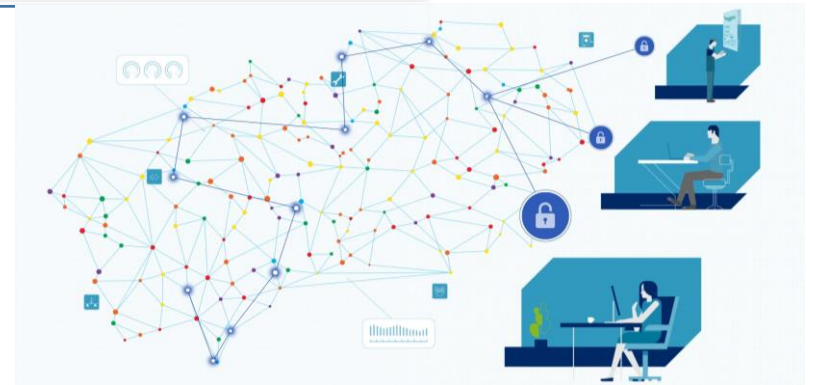
# State of the Art – Detailed View



# The Answers – Hypothesis of this Project

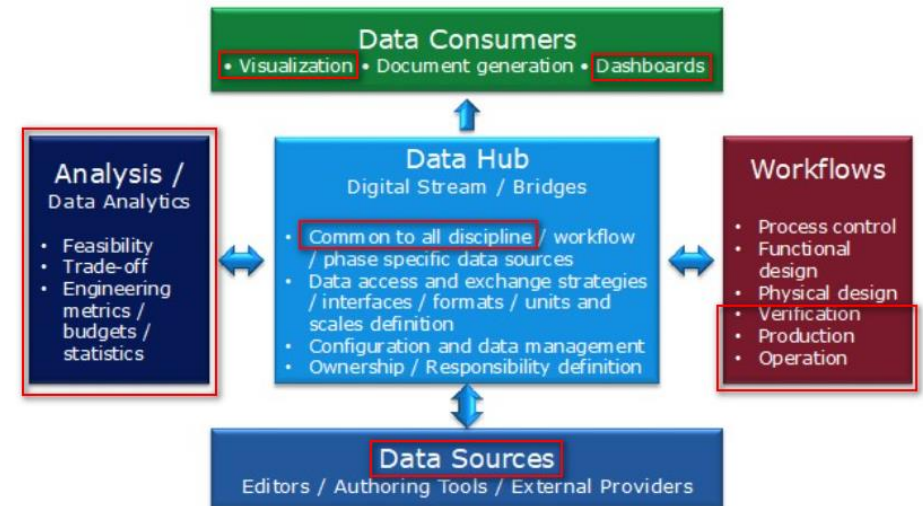


- Integration of MBSE and AIT / Operations into a holistic Workflow
  - Elimination of (functional) failures in the context of the AIT process
- Supported by suitable visualisations
  - less system failure
  - faster troubleshooting
  - improved communication between the project participants.

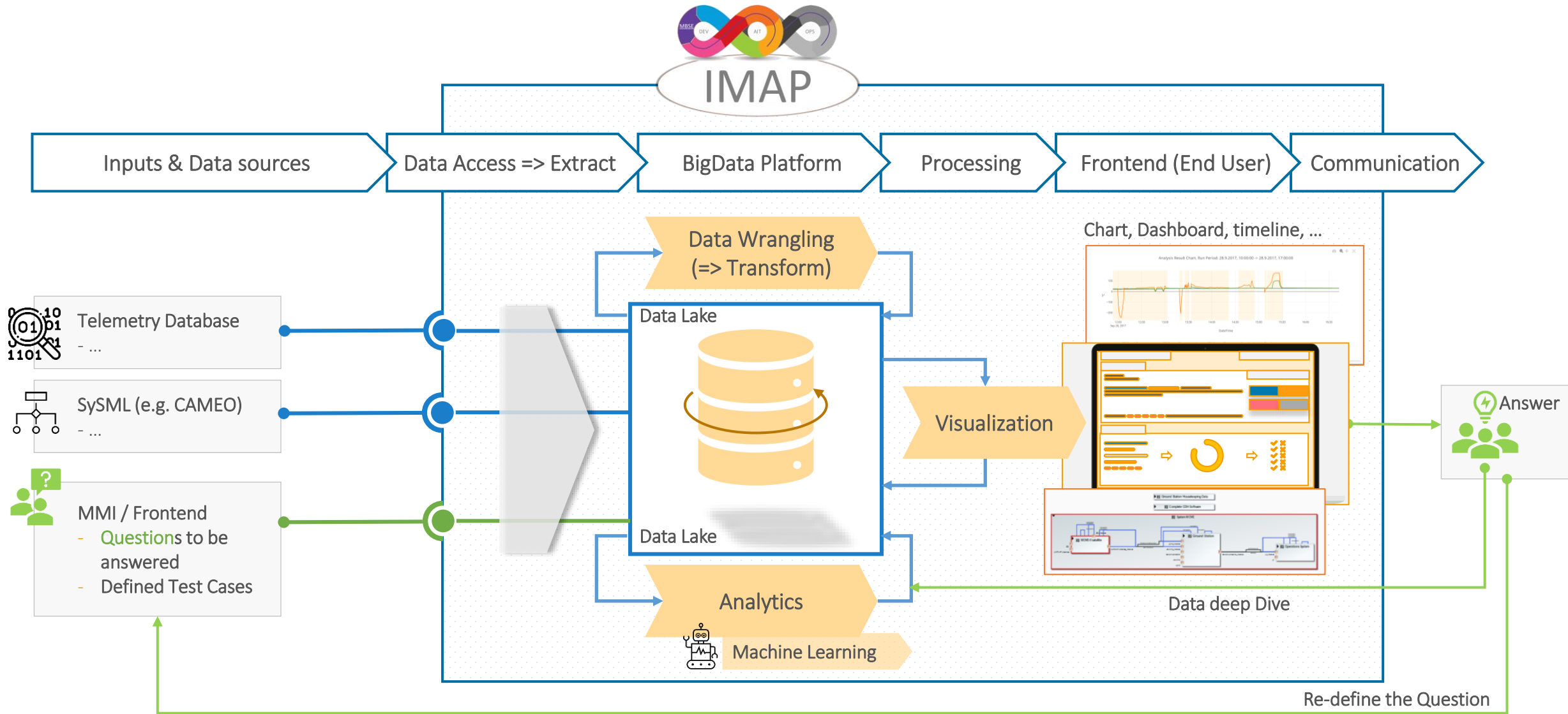


# Integrated MBSE Analytics Platform - IMAP

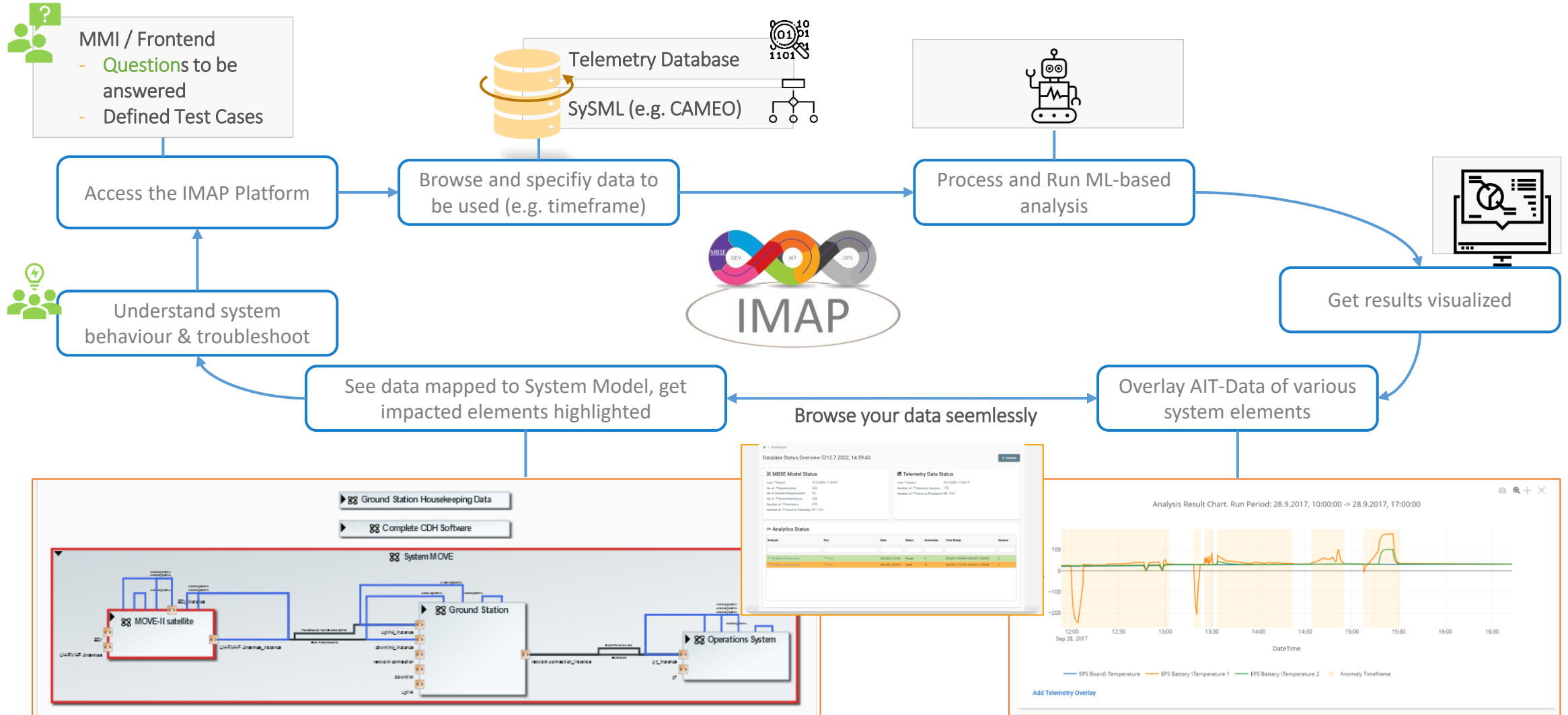
- Integrate MBSE and AIT with MachineLearning based anomaly detection into one platform
- Developed as technology demonstrator in an ESA funded research project within the OSIP activity
- Aligned with ESA’s MBSE strategy  
<https://essr.esa.int/project/mb4se-model-based-for-system-engineering>
- Aligned with results and user stories from Expert interviews in Spring 2021
- Project completion July 2022



# IMAP – Overall Process

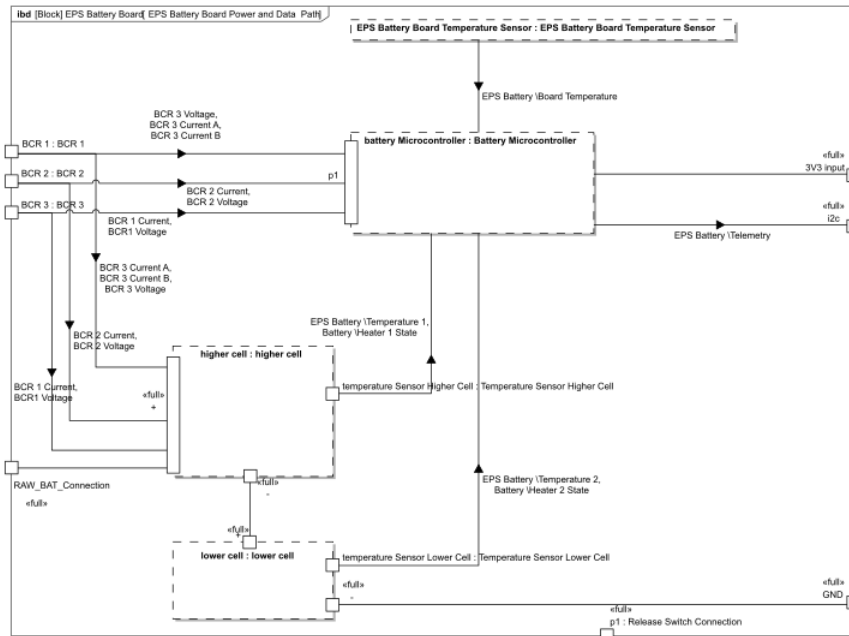
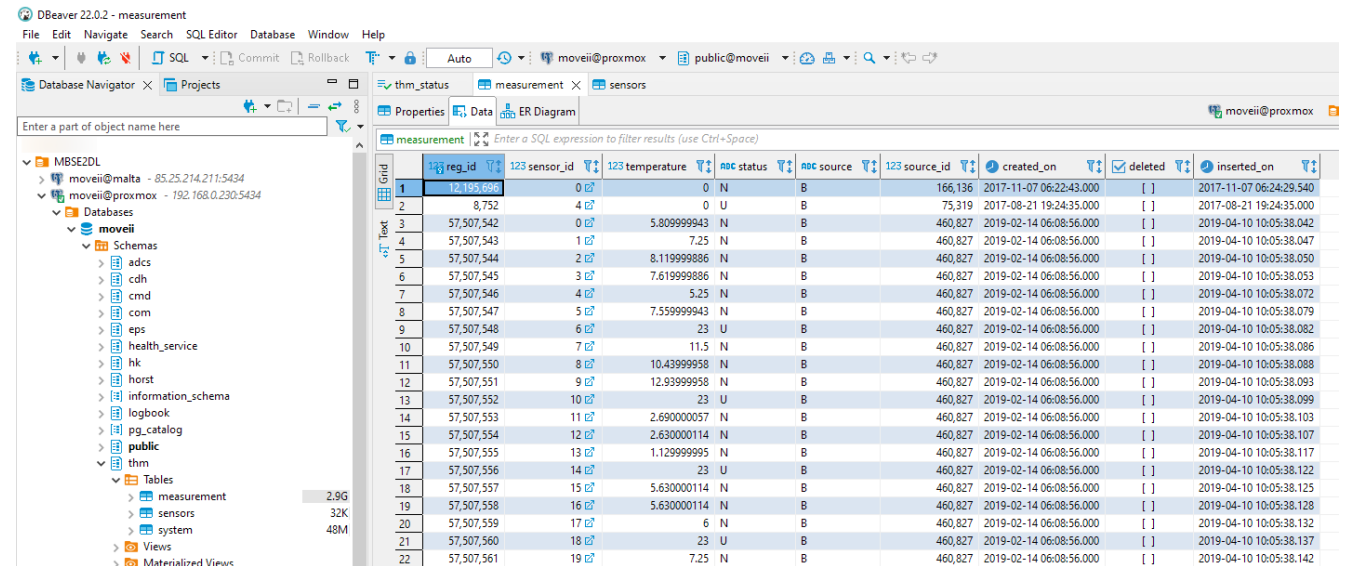


# IMAP – Integrated MBSE to AIT Workflow



# Demonstration Data

- All demonstration data is completely based on the operational [MOVE-II CubeSat mission](https://docs.mbse2dl.org/overview/demo-data/) which has been developed by the Chair of Astronautics at TU Munich

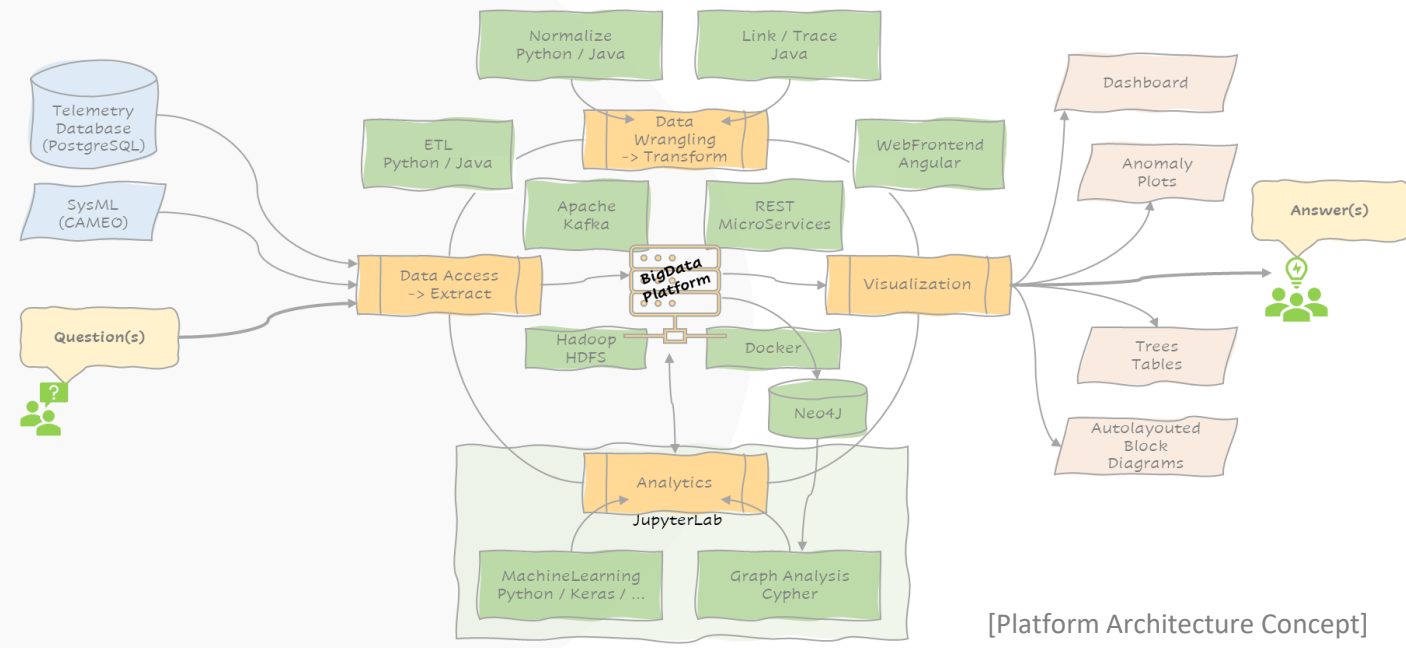
reg_id	sensor_id	123 temperature	abc status	abc source	123 source_id	created_on	deleted	inserted_on
1	12,195,698	0	N	B	166,136	2017-11-07 06:22:43.000	[ ]	2017-11-07 06:24:29.540
2	8,752	4	U	B	75,319	2017-08-21 19:24:35.000	[ ]	2017-08-21 19:24:35.000
3	57,507,542	0	N	B	460,827	2019-02-14 06:08:56.000	[ ]	2019-04-10 10:05:38.042
4	57,507,543	1	N	B	460,827	2019-02-14 06:08:56.000	[ ]	2019-04-10 10:05:38.047
5	57,507,544	2	N	B	460,827	2019-02-14 06:08:56.000	[ ]	2019-04-10 10:05:38.050
6	57,507,545	3	N	B	460,827	2019-02-14 06:08:56.000	[ ]	2019-04-10 10:05:38.053
7	57,507,546	4	N	B	460,827	2019-02-14 06:08:56.000	[ ]	2019-04-10 10:05:38.072
8	57,507,547	5	N	B	460,827	2019-02-14 06:08:56.000	[ ]	2019-04-10 10:05:38.079
9	57,507,548	6	N	B	460,827	2019-02-14 06:08:56.000	[ ]	2019-04-10 10:05:38.082
10	57,507,549	7	N	B	460,827	2019-02-14 06:08:56.000	[ ]	2019-04-10 10:05:38.086
11	57,507,550	8	N	B	460,827	2019-02-14 06:08:56.000	[ ]	2019-04-10 10:05:38.088
12	57,507,551	9	N	B	460,827	2019-02-14 06:08:56.000	[ ]	2019-04-10 10:05:38.093
13	57,507,552	10	N	B	460,827	2019-02-14 06:08:56.000	[ ]	2019-04-10 10:05:38.099
14	57,507,553	11	N	B	460,827	2019-02-14 06:08:56.000	[ ]	2019-04-10 10:05:38.103
15	57,507,554	12	N	B	460,827	2019-02-14 06:08:56.000	[ ]	2019-04-10 10:05:38.107
16	57,507,555	13	N	B	460,827	2019-02-14 06:08:56.000	[ ]	2019-04-10 10:05:38.117
17	57,507,556	14	N	B	460,827	2019-02-14 06:08:56.000	[ ]	2019-04-10 10:05:38.122
18	57,507,557	15	N	B	460,827	2019-02-14 06:08:56.000	[ ]	2019-04-10 10:05:38.125
19	57,507,558	16	N	B	460,827	2019-02-14 06:08:56.000	[ ]	2019-04-10 10:05:38.128
20	57,507,559	17	N	B	460,827	2019-02-14 06:08:56.000	[ ]	2019-04-10 10:05:38.132
21	57,507,560	18	N	B	460,827	2019-02-14 06:08:56.000	[ ]	2019-04-10 10:05:38.137
22	57,507,561	19	N	B	460,827	2019-02-14 06:08:56.000	[ ]	2019-04-10 10:05:38.142

- For details: See <https://docs.mbse2dl.org/overview/demo-data/>



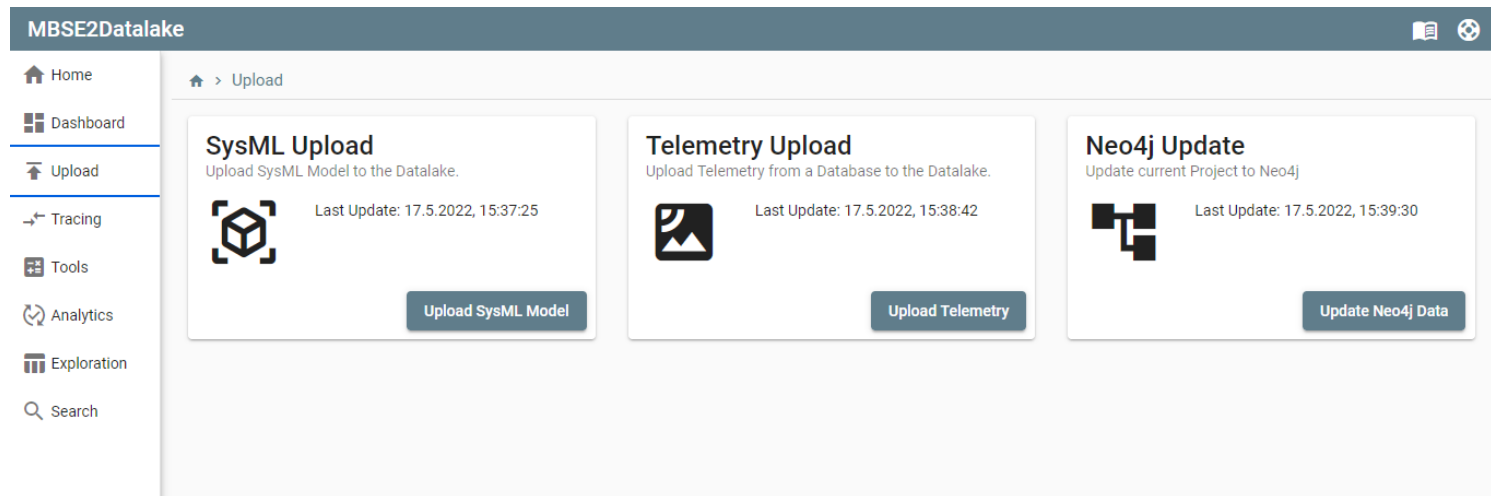
# The Platform

- Scalable for BigData
  - State of the art building blocks: Docker based micro services, Hadoop Filesystem, Kafka Messaging, ...
- Formalized
  - Semantic DataLake concept, underlying Ontology modeled in OWL – based on CIP metamodel, MBSE storage in RDF
- Flexible
  - Analytics functions implemented in industry standard Jupyter Notebooks – for expert users
  - Integrated into end-user-friendly web frontend
- Extensible



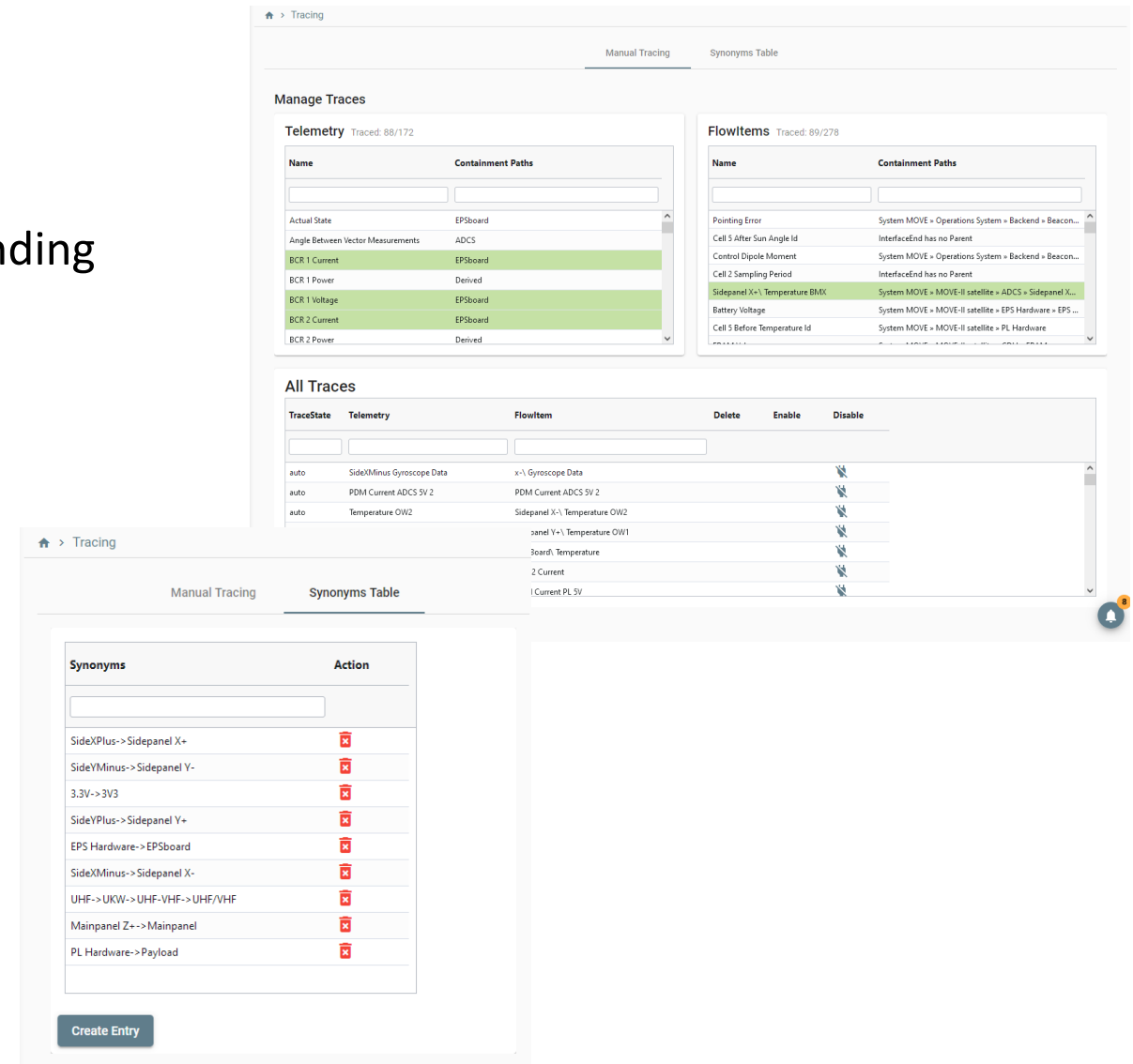
# Importing Data

- In the technology demonstrator, data imports are specifically tailored for MOVE-II data (MagicDraw SysML model and PostgreSQL Telemetry Database)
- These 2 importers will need to be adapted or rewritten for different MBSE models and Telemetry Databases!



# Tracing MBSE to Telemetry Data

- Traces between MBSE FlowItems and Telemetry sensors can be detected
  - Automatically (to a certain degree, depending on Model and Sensor structure)
  - By manual drag&drop
  
- By defining a list of synonymous terms



The screenshot displays the 'Tracing' application interface, divided into two main sections: 'Manual Tracing' and 'Synonyms Table'.

**Manual Tracing Section:**

- Manage Traces:** Contains two panels:
  - Telemetry Traced: 88/172:** A table listing telemetry items and their containment paths.
 

Name	Containment Paths
Actual State	EPSboard
Angle Between Vector Measurements	ADCS
BCR 1 Current	EPSboard
BCR 1 Power	Derived
BCR 1 Voltage	EPSboard
BCR 2 Current	EPSboard
BCR 2 Power	Derived
  - FlowItems Traced: 89/278:** A table listing flow items and their containment paths.
 

Name	Containment Paths
Pointing Error	System MOVE » Operations System » Backend » Beacon...
Cell 5 After Sun Angle Id	InterfaceEnd has no Parent
Control Dipole Moment	System MOVE » Operations System » Backend » Beacon...
Cell 2 Sampling Period	InterfaceEnd has no Parent
Sidepanel X-1 Temperature BMX	System MOVE » MOVE-II satellite » ADCS » Sidepanel X...
Battery Voltage	System MOVE » MOVE-II satellite » EPS Hardware » EPS ...
Cell 5 Before Temperature Id	System MOVE » MOVE-II satellite » PL Hardware
- All Traces:** A table listing all detected traces with columns for TraceState, Telemetry, FlowItem, Delete, Enable, and Disable.
 

TraceState	Telemetry	FlowItem	Delete	Enable	Disable
auto	SideXMinus Gyroscope Data	x-1 Gyroscope Data			
auto	PDM Current ADCS SV 2	PDM Current ADCS SV 2			
auto	Temperature OW2	Sidepanel X-1 Temperature OW2			
		anel Y-1 Temperature OW1			
		3board Temperature			
		2 Current			
		1 Current PL SV			

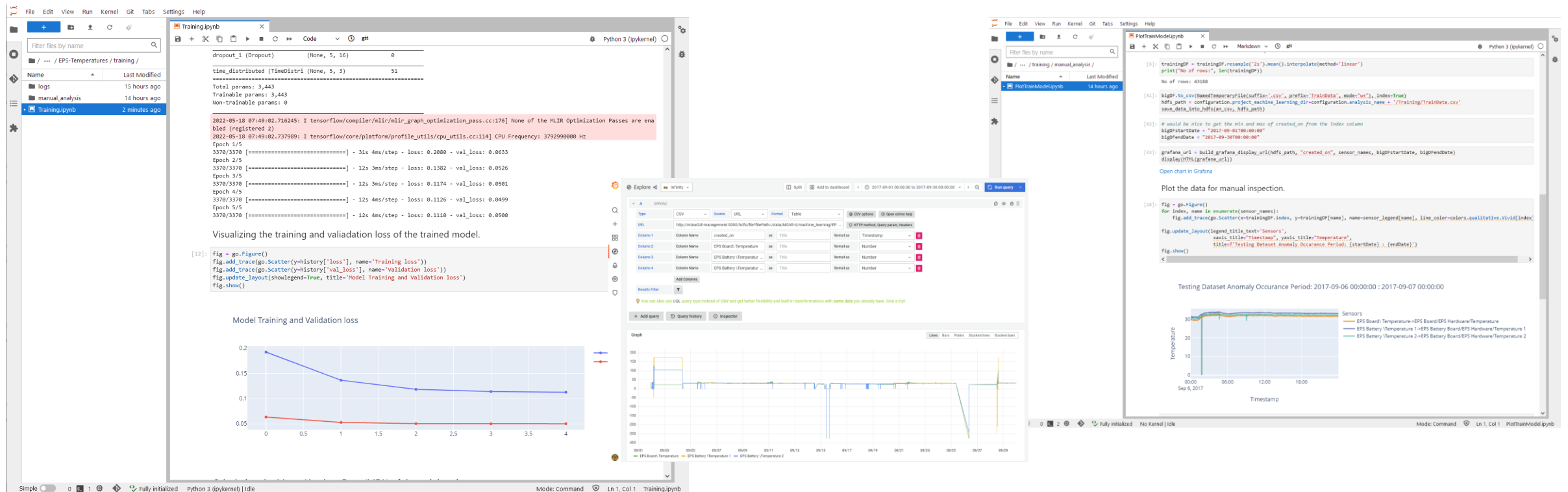
**Synonyms Table Section:**

- Synonyms Table:** A table defining synonymous terms for tracing.
 

Synonyms	Action
SideXPlus->Sidepanel X+	[X]
SideYMinus->Sidepanel Y-	[X]
3.3V->3V3	[X]
SideYPlus->Sidepanel Y+	[X]
EPS Hardware->EPSboard	[X]
SideXMinus->Sidepanel X-	[X]
UHF->UKW->UHF-VHF->UHF/VHF	[X]
Mainpanel Z+>Mainpanel	[X]
PL Hardware->Payload	[X]
- Create Entry:** A button to add new synonym entries.

# Feature: ML based automated Anomaly Detection

- On-platform setup of MachineLearning models and AnomalyDetection with JupyterLab for expert users/data analysts – no need to care about the technical infrastructure
- Results flow into DataLake for storage and end-user friendly front-ends



The image displays a workflow for ML-based anomaly detection. It starts with a JupyterLab notebook titled 'Training.ipynb' showing the training of a model. The notebook output includes training and validation loss metrics over 5 epochs:

Epoch	Time	Loss	Val Loss
1/5	3370/3370	0.2080	0.0633
2/5	3370/3370	0.1382	0.0526
3/5	3370/3370	0.1174	0.0501
4/5	3370/3370	0.1126	0.0499
5/5	3370/3370	0.1110	0.0500

The notebook also includes code to visualize the training and validation loss:

```

fig = go.Figure()
fig.add_trace(go.Scatter(y=history['loss'], name='Training loss'))
fig.add_trace(go.Scatter(y=history['val_loss'], name='Validation loss'))
fig.update_layout(showlegend=True, title='Model Training and Validation loss')
fig.show()

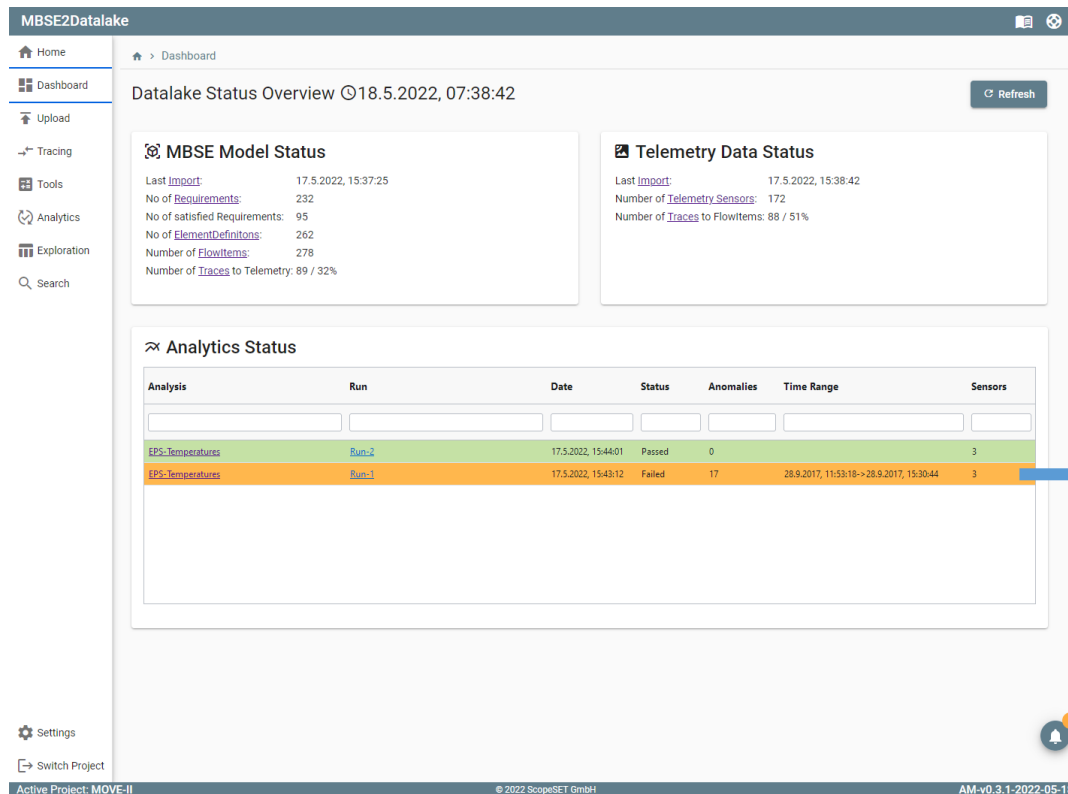
```

The next screenshot shows a Databricks Data Explorer interface displaying a table of data with columns for 'Columns Name', 'created\_on', 'Title', 'Timestamp', and 'Sensor'. The table contains four rows of data.

The final screenshot shows a Grafana dashboard titled 'Testing Dataset Anomaly Occurrence Period: 2017-09-06 00:00:00 - 2017-09-07 00:00:00'. The dashboard displays a line chart showing 'Temperature' over 'Timestamp' for various sensors. The sensors listed are: EPS Board/ Temperature -> EPS Board/ EPS Hardware/ Temperature, EPS Battery/ Temperature 1 -> EPS Battery Board/ EPS Hardware/ Temperature 1, and EPS Battery/ Temperature 2 -> EPS Battery Board/ EPS Hardware/ Temperature 2. The chart shows a significant spike in temperature around 06:00 on Sep 6, 2017.

# Feature: MBSE and Analytics – Overview

- All data in one place
- From Dashboard to ML based analysis results...



**MBSE2DataLake**

Dashboard

Datalake Status Overview 18.5.2022, 07:38:42

**MBSE Model Status**

Last Import: 17.5.2022, 15:37:25  
 No of Requirements: 232  
 No of satisfied Requirements: 95  
 No of ElementDefinitions: 262  
 Number of FlowItems: 278  
 Number of Traces to Telemetry: 89 / 32%

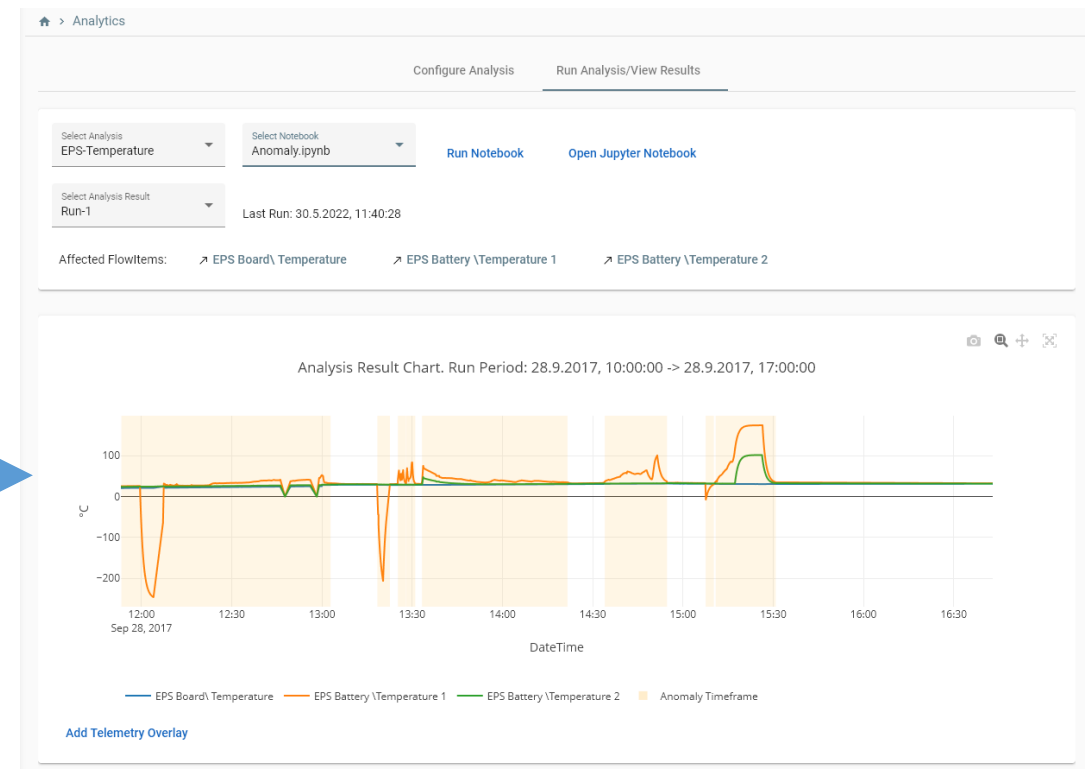
**Telemetry Data Status**

Last Import: 17.5.2022, 15:38:42  
 Number of Telemetry Sensors: 172  
 Number of Traces to FlowItems: 88 / 51%

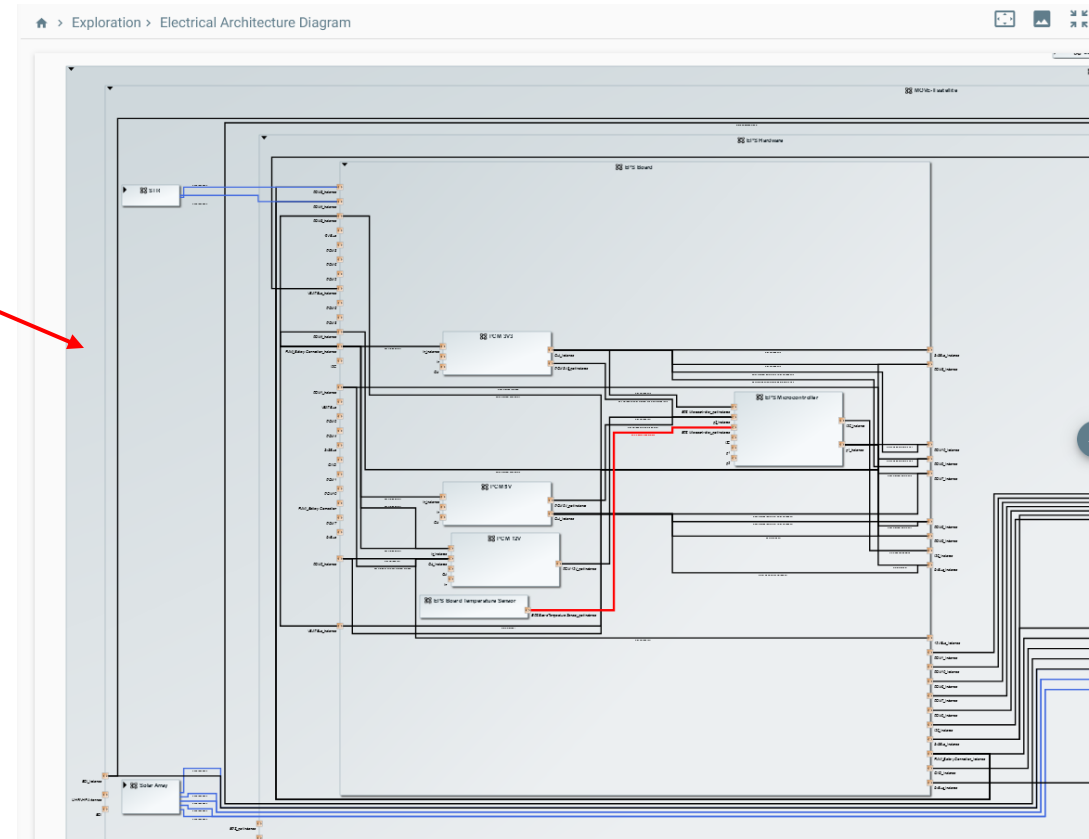
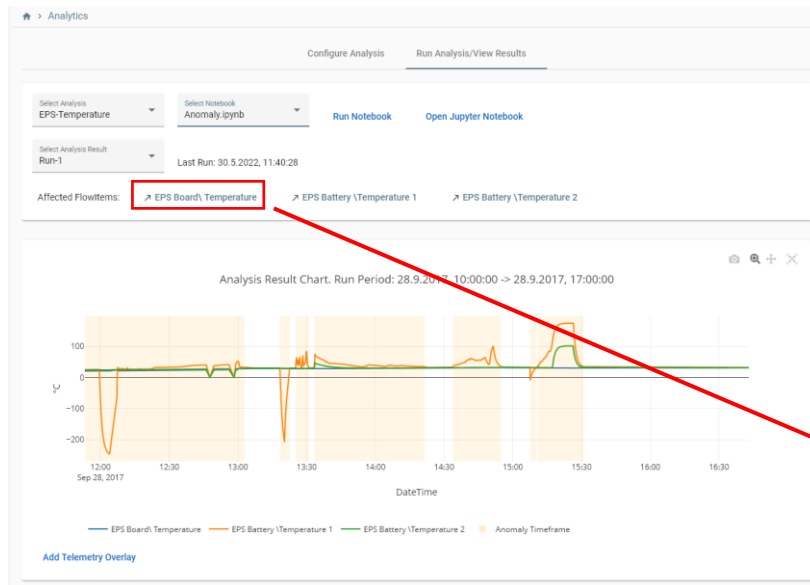
**Analytics Status**

Analysis	Run	Date	Status	Anomalies	Time Range	Sensors
EPS-Temperatures	Run-2	17.5.2022, 15:44:01	Passed	0		3
EPS-Temperatures	Run-1	17.5.2022, 15:43:12	Failed	17	28.9.2017, 11:53:18->28.9.2017, 15:30:44	3

Settings | Switch Project | Active Project: MOVE-II | © 2022 ScopeSET GmbH | AM-v0.3.1-2022-05-18

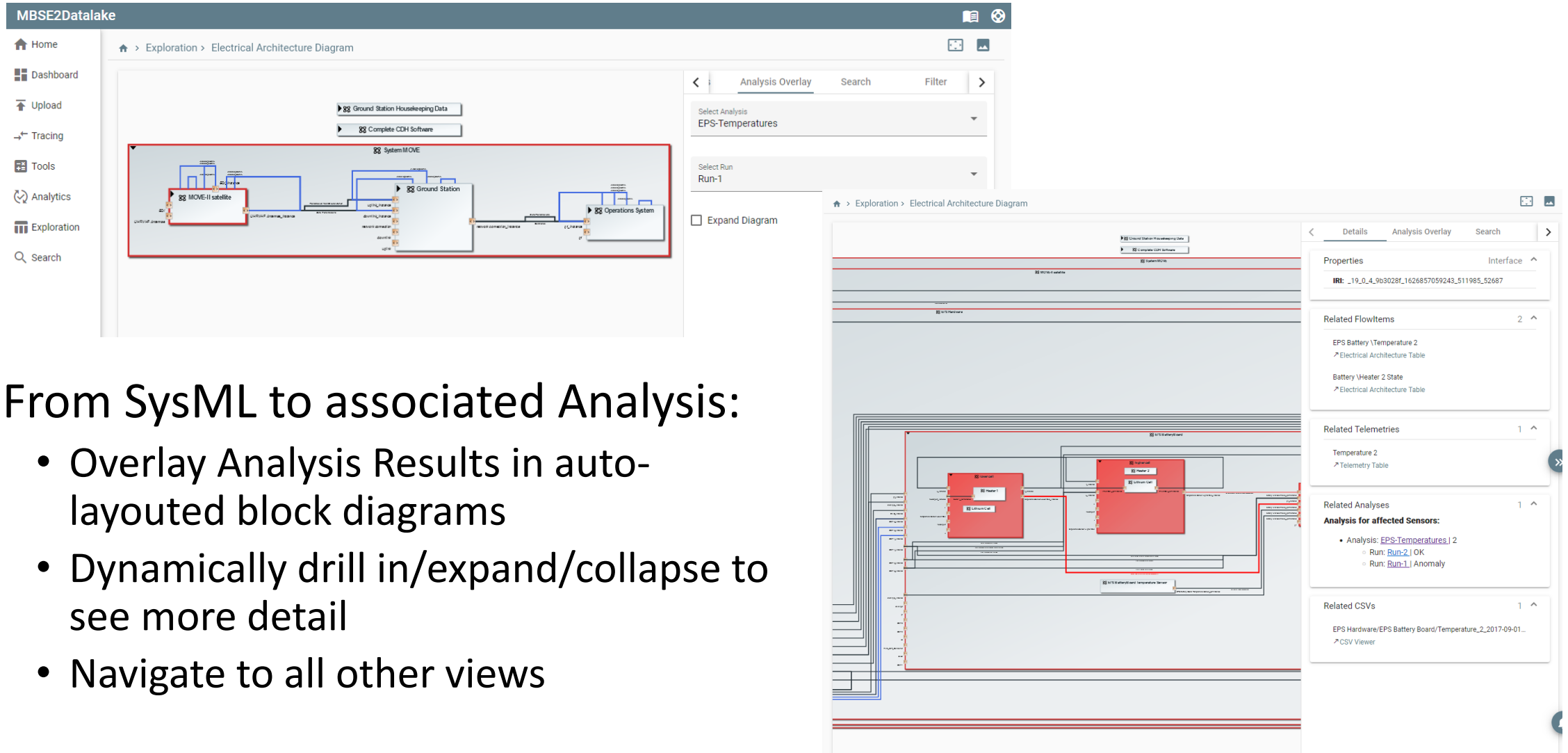


# Feature: Block Diagrams with Failure Highlighting



- Navigate from detected Anomaly to affected FlowItem in auto-laid out Block Diagram

# Feature: Block Diagrams as Entry Points

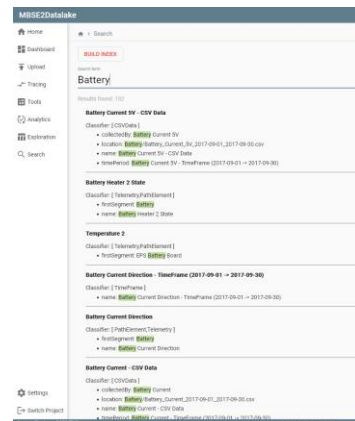


The screenshot displays the MBSE2DataLake interface. The main window shows an 'Electrical Architecture Diagram' with a red box highlighting a specific component, 'MOVE-II satellite'. An 'Analysis Overlay' panel is visible, showing 'EPS-Temperatures' analysis results for 'Run-1'. A detailed view of the 'EPS-Temperature' component is shown on the right, with a sidebar containing 'Properties', 'Related FlowItems', 'Related Telemetries', 'Related Analyses', and 'Related CSVs'.

- From SysML to associated Analysis:
  - Overlay Analysis Results in auto-laid out block diagrams
  - Dynamically drill in/expand/collapse to see more detail
  - Navigate to all other views

# Feature: Assistance Functions

- MBSE centric views, Trees, Tables
- Drill-in capable autolayouting block diagram with Search, Filters, ...
- Integration with state-of-the-art Chart Visualization (Grafana)
- Global search



Exploration > Requirement Table

### All Requirements

Req Id	Name	Derived From	Lower Reqs	Satisfied
126	% EPS-08	1	0	0
127	% EPS-09	1	0	0
128	% EPS-10	1	0	0
179	%EPS-00.2	1	0	0
72	%EPS-07	2	0	0
157	EPS-00.1	2	0	0
158	EPS-00.2A	1	0	0
35	EPS-01	1	0	2
36	EPS-02	1	1	2
36.1	EPS-02.1	1	0	2
37	EPS-03	1	3	3
37.2	EPS-03.1	1	0	0
37.3	EPS-03.2	1	0	0
37.1	EPS-03.3	2	0	2
38	EPS-04	1	0	1
39	EPS-05	2	0	0
40	EPS-06	1	0	0
41.1	EPS-07.1	1	0	1
41	EPS-07A	1	1	0
4	EPS-11	1	2	2
4.1	EPS-11.1	1	0	1

#### Requirement Details

**Id:** 35  
**Name:** EPS-01  
**Description:** The EPS shall be able to handle the maximum occurring input power in orbit.

**Derived From Requirements:**

- [EPS-00.1](#)

**Satisfied by ElementDefinitions:**

- lower cell: [EA Diagram PS Table PS Tree](#)  
**Analysis for affected Sensors:**
  - FlowItem: [EPS Battery \Temperature 2](#)
    - Analysis: [EPS-Temperatures | 2](#)
      - Run: [Run-2](#) | OK
      - Run: [Run-1](#) | Anomaly
- higher cell: [EA Diagram PS Table PS Tree](#)  
**Analysis for affected Sensors:**
  - FlowItem: [EPS Battery \Temperature 1](#)
    - Analysis: [EPS-Temperatures | 2](#)
      - Run: [Run-2](#) | OK
      - Run: [Run-1](#) | Anomaly

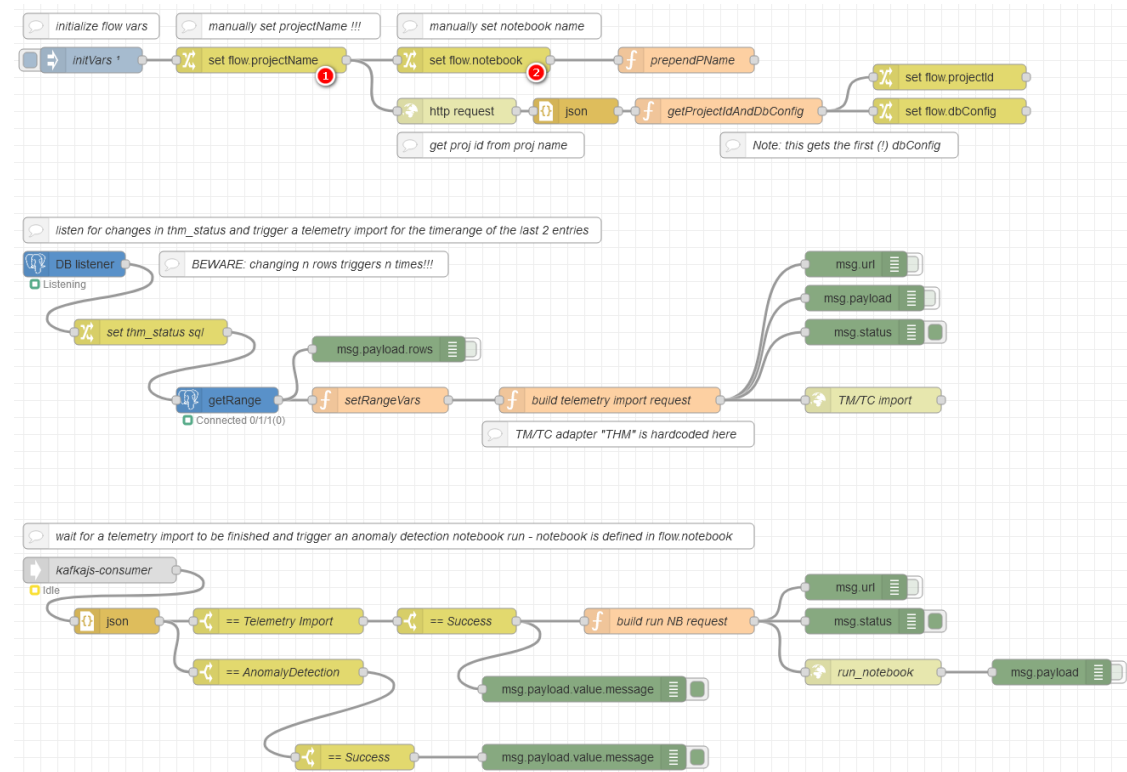


# Feature: Orchestration of Data Ingestion

- Automated data ingestion and triggering of Analytics functions

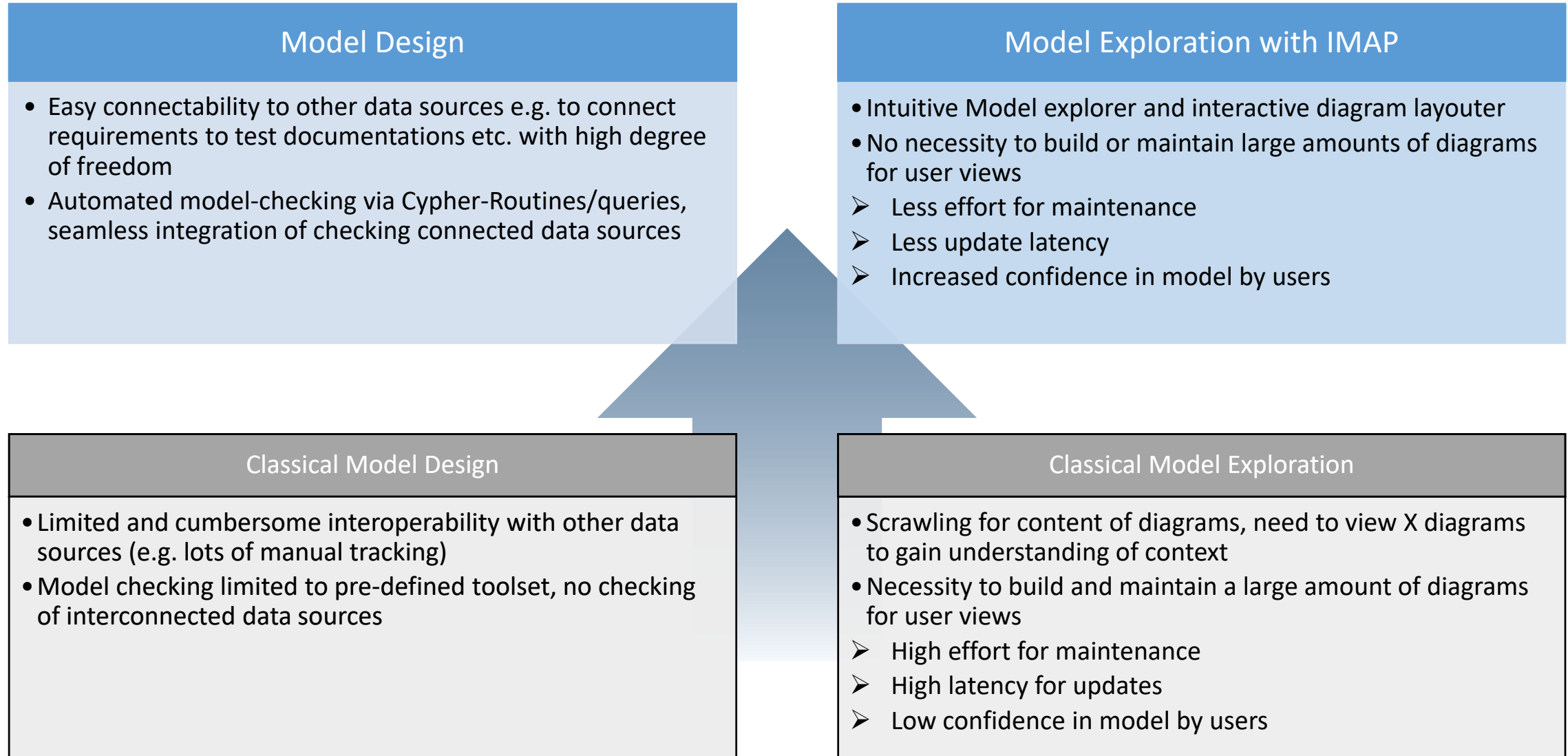
- Node-Red flow diagram:

- Model the ingestion workflow based on specific input data access and trigger tailored Analysis scripts
- Result will be displayed in standard dashboards and views



# User Feedback

## Workflow Comparison – Design Phase



# User Feedback

## Workflow Comparison – Testing and Operations

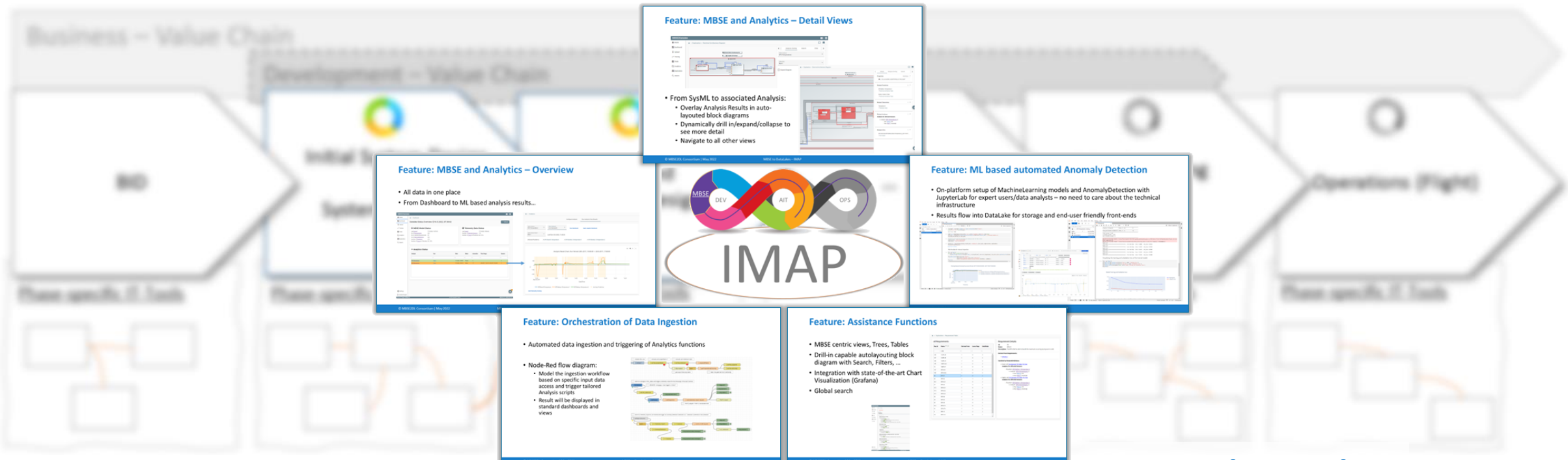
### IMAP Integration Debugging Process

- Fault detection facilitated by AI
- Fault tracing simplified by automated model analysis
- Proposal of „best guess“ for components likely at fault based on deductions from Model and telemetry

### Classical Integration Debugging Process

- Reliant on key personnel
- Requires holistic understanding of the system
- Trial and error based on intuition

# Summary And Final Hypothesis



Additional resources  
in early phases

Pay out in Testing  
and Operations

Resources  
Spent



# Deployment and Demo

- Available online on <https://www.mbsd2dl.org>
- Including extensive documentation at <https://docs.mbsd2dl.org>
- Questions and Support: [amueller@scopeset.de](mailto:amueller@scopeset.de)

# Feedback on Solution Demo(s)



Stephan Finkel

3DSE, SE-Principal - Defence,  
Space and Automotive

"... many of our customers use (MBSE) system models extensively in the early phases of complex system developments, for example in naval shipbuilding and automotive engineering. However, there is often a gap in the use of models, which is due to time, organisational, but also competence-related reasons. A solution like IMAP can become a valuable building block in the E2E tool landscape. In addition, we see the opportunity that such a tool could both strengthen the end-to-end SE idea and enable the desire of many customers to shorten and agilise projects."



Patrik Krause

3DSE, Project Lead AMELIE,  
Advanced SE for SMEs

"... in our experience, SMEs in particular find it difficult to implement systems engineering holistically. This is due on the one hand to the extensive methodology, and on the other hand to the fact that people are often involved in projects in multiple roles. In the field of medical device development, we often have teams of < 15 people. We see the potential of the IMAP solution in the fact that a system expert can create the test cases and train the ML-system for approval-relevant test scenarios. The highly qualified person can then take on other tasks and only provide support in the event of an anomaly, then also with direct reference to the system model."

# Feedback on Solution Demo(s)



Rupert Amann

Head of Space Development  
and Co-Founder of OroraTech

"... after an internal presentation of the MBSE2DL system, we hereby express our interest in the technology. The fast and clear failure analysis, as well as the possibility to automatically generate analyses and graphs for reports from the DataLake, are technologies with which we recognise a considerable added value for our satellite development. We see huge potential in machine-learning based telemetry analysis for the mission operations of our future satellite constellations."



Felix Firmbach

WARR MOVE Student Lead

"After learning about the IMAP tool with the MBSE2DL system, we are certain that it would be highly beneficial in the development and operation of our satellites. By directly coupling a SysML model with telemetry data, existing SysML models are enhanced and put at the forefront of development. The automatic analysis and the display of which systems are affected by an identified anomaly in the SysML model have the potential to significantly enhance and speed up problem discovery and resolution. Both the development and operation of satellites would greatly benefit from this. Furthermore, the tool uses well-known technologies such as Python with JupyterLab and Grafana. This makes it easier to get started with the tool."