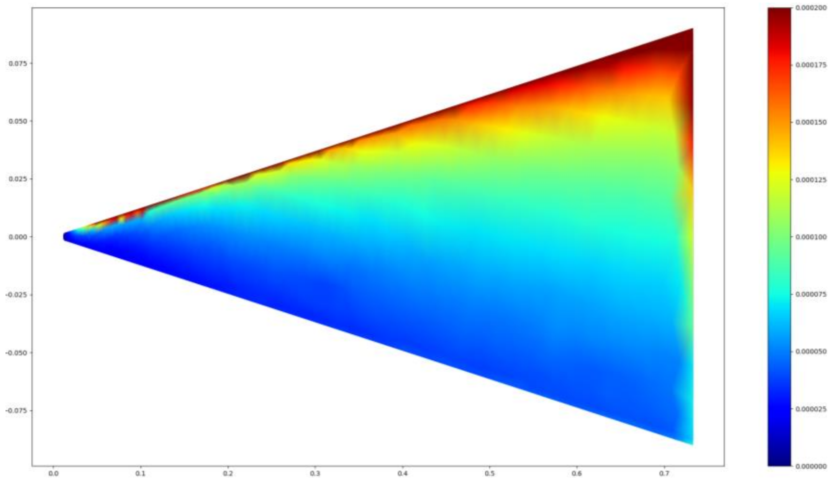
PFAT

Post Flight Analysis Toolset for ESA Missions

Final Presentation



PFAT FP
Videoconference
08/02/2022

Agenda

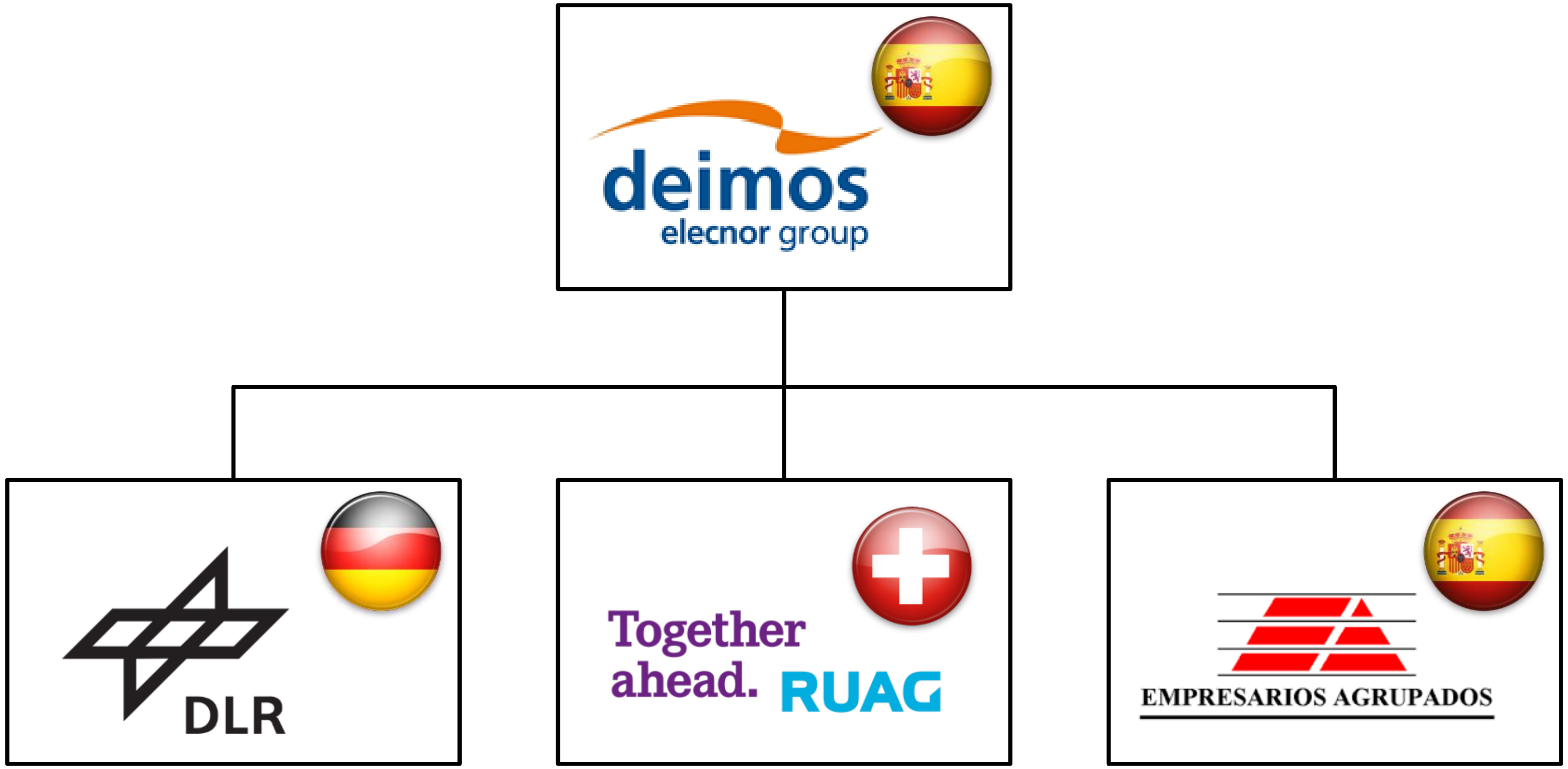
- Project History
- Software Development
- PFAT Capabilities
- Validation Campaign
- Achievements and Future Works
- Conclusions

1 Project History

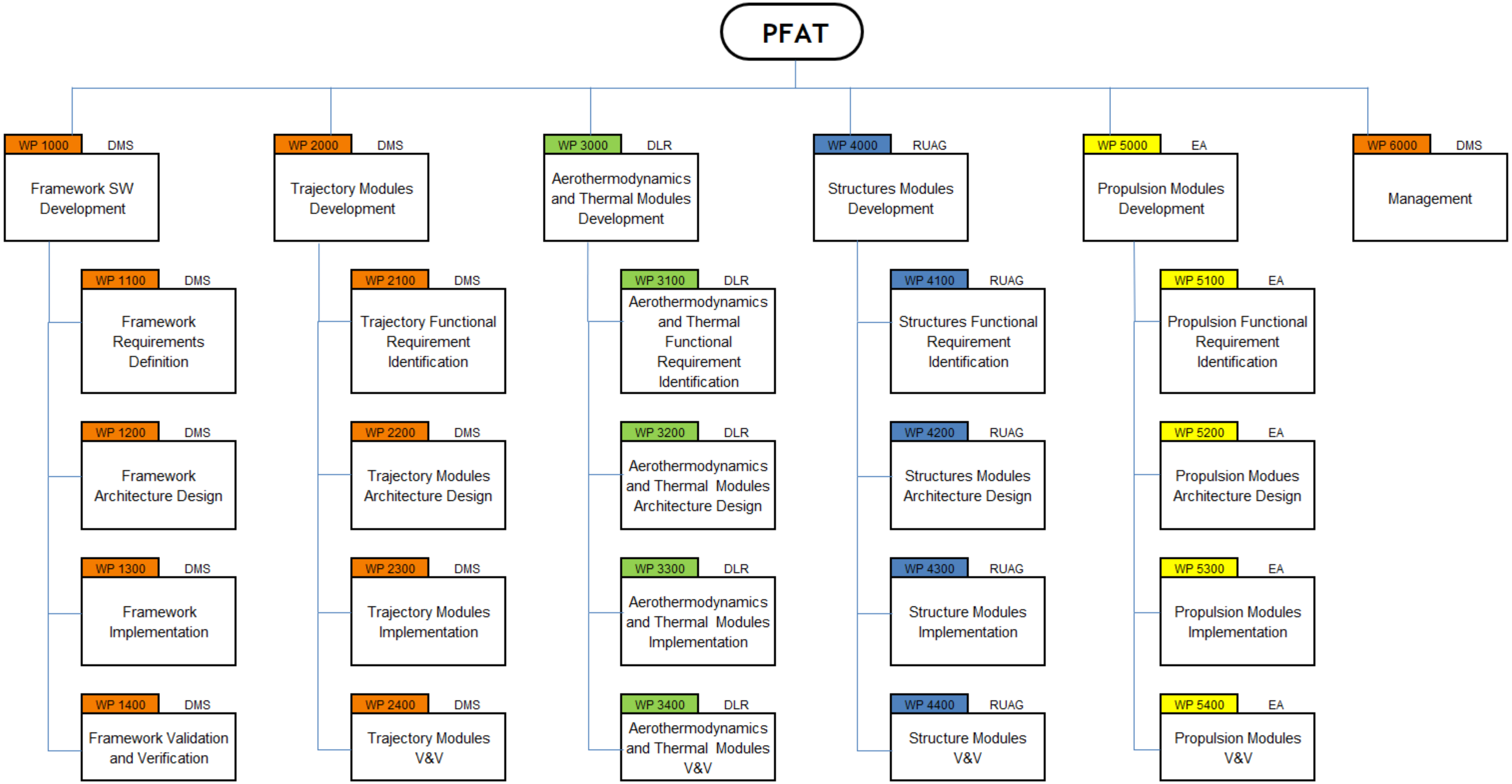
Objectives

- ❑ Develop a **Post-Flight Analysis Toolkit** (PFAT) SW for use in ESA missions and interoperable with other engineering software tools commonly used by ESA
- ❑ Implement **post-flight algorithms** and **analysis tools** to support different engineering domains: **propulsion**, **aerothermodynamics**, **structures/separation**, and **trajectories**
- ❑ Make use of **standard exchange formats**
- ❑ **Automatic** generation of post-flight analysis **reports**
- ❑ **Open-source**

Project Team

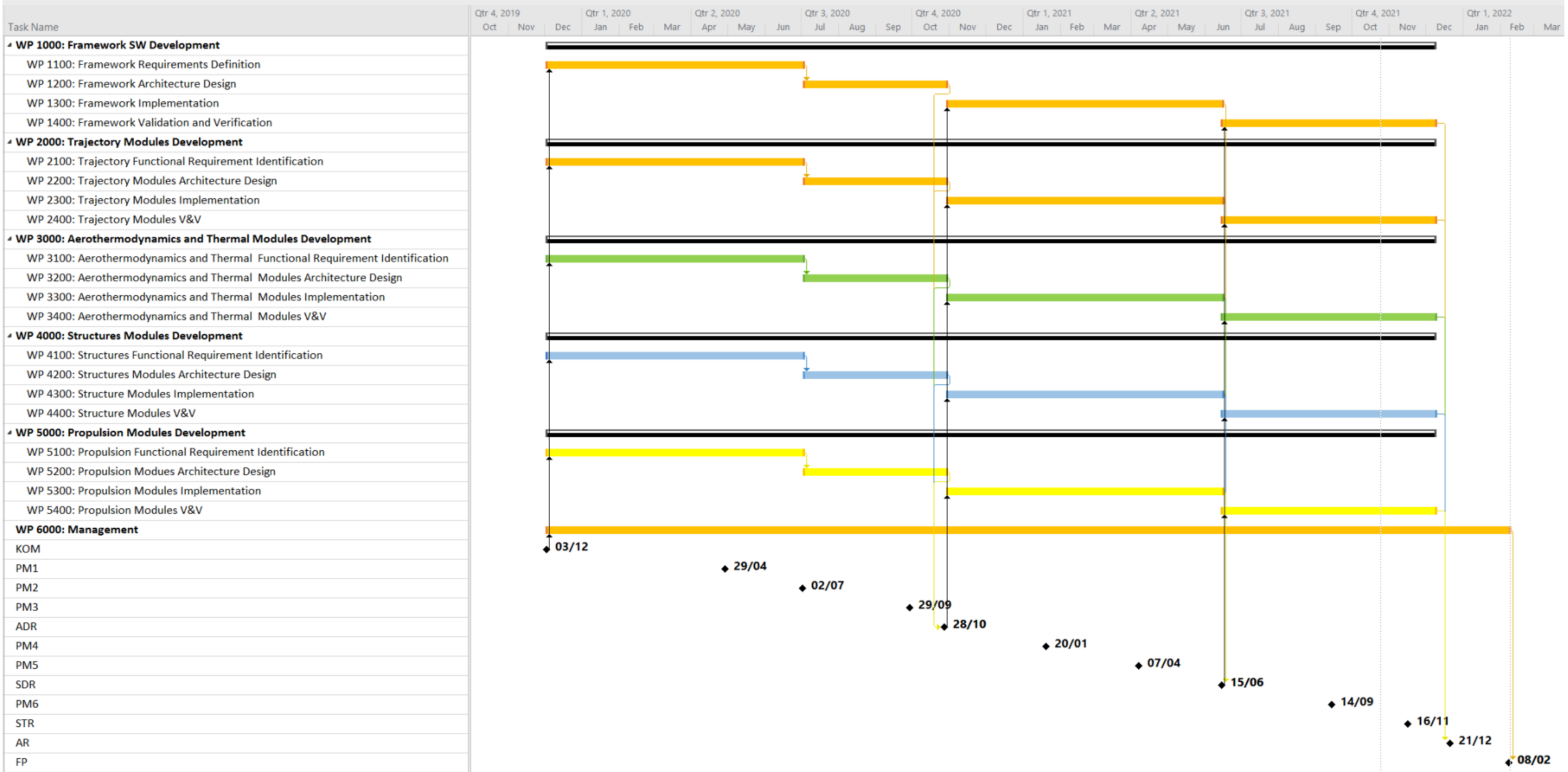


WBS



PFAT
WP 1000 DMS
Framework SW Development
WP 1100 DMS
Framework Requirements Definition
WP 1200 DMS
Framework Architecture Design
WP 1300 DMS
Framework Implementation
WP 1400 DMS
Framework Validation and Verification
WP 2000 DMS
Trajectory Modules Development
WP 2100 DMS
Trajectory Functional Requirement Identification
WP 2200 DMS
Trajectory Modules Architecture Design
WP 2300 DMS
Trajectory Modules Implementation
WP 2400 DMS
Trajectory Modules V&V
WP 3000 DLR
Aerothermodynamics and Thermal Modules Development
WP 3100 DLR
Aerothermodynamics and Thermal Functional Requirement Identification
WP 3200 DLR
Aerothermodynamics and Thermal Modules Architecture Design
WP 3300 DLR
Aerothermodynamics and Thermal Modules Implementation
WP 3400 DLR
Aerothermodynamics and Thermal Modules V&V
WP 4000 RUAG
Structures Modules Development
WP 4100 RUAG
Structures Functional Requirement Identification
WP 4200 RUAG
Structures Modules Architecture Design
WP 4300 RUAG
Structure Modules Implementation
WP 4400 RUAG
Structure Modules V&V
WP 5000 EA
Propulsion Modules Development
WP 5100 EA
Propulsion Functional Requirement Identification
WP 5200 EA
Propulsion Modues Architecture Design
WP 5300 EA
Propulsion Modules Implementation
WP 5400 EA
Propulsion Modules V&V
WP 6000 DMS
Management

Schedule


Task Name
Qtr 4, 2019
Oct
Nov
Dec
Qtr 1, 2020
Jan
Feb
Mar
Qtr 2, 2020
Apr
May
Jun
Qtr 3, 2020
Jul
Aug
Sep
Qtr 4, 2020
Oct
Nov
Dec
Qtr 1, 2021
Jan
Feb
Mar
Qtr 2, 2021
Apr
May
Jun
Qtr 3, 2021
Jul
Aug
Sep
Qtr 4, 2021
Oct
Nov
Dec
Qtr 1, 2022
Jan
Feb
Mar
WP 1000: Framework SW Development
WP 1100: Framework Requirements Definition
WP 1200: Framework Architecture Design
WP 1300: Framework Implementation
WP 1400: Framework Validation and Verification
WP 2000: Trajectory Modules Development
WP 2100: Trajectory Functional Requirement Identification
WP 2200: Trajectory Modules Architecture Design
WP 2300: Trajectory Modules Implementation
WP 2400: Trajectory Modules V&V
WP 3000: Aerothermodynamics and Thermal Modules Development
WP 3100: Aerothermodynamics and Thermal Functional Requirement Identification
WP 3200: Aerothermodynamics and Thermal Modules Architecture Design
WP 3300: Aerothermodynamics and Thermal Modules Implementation
WP 3400: Aerothermodynamics and Thermal Modules V&V
WP 4000: Structures Modules Development
WP 4100: Structures Functional Requirement Identification
WP 4200: Structures Modules Architecture Design
WP 4300: Structure Modules Implementation
WP 4400: Structure Modules V&V
WP 5000: Propulsion Modules Development
WP 5100: Propulsion Functional Requirement Identification
WP 5200: Propulsion Modues Architecture Design
WP 5300: Propulsion Modules Implementation
WP 5400: Propulsion Modules V&V
WP 6000: Management
KOM
03/12
PM1
29/04
PM2
02/07
PM3
29/09
ADR
28/10
PM4
20/01
PM5
07/04
SDR
15/06
PM6
14/09
STR
16/11
AR
21/12
FP
08/02

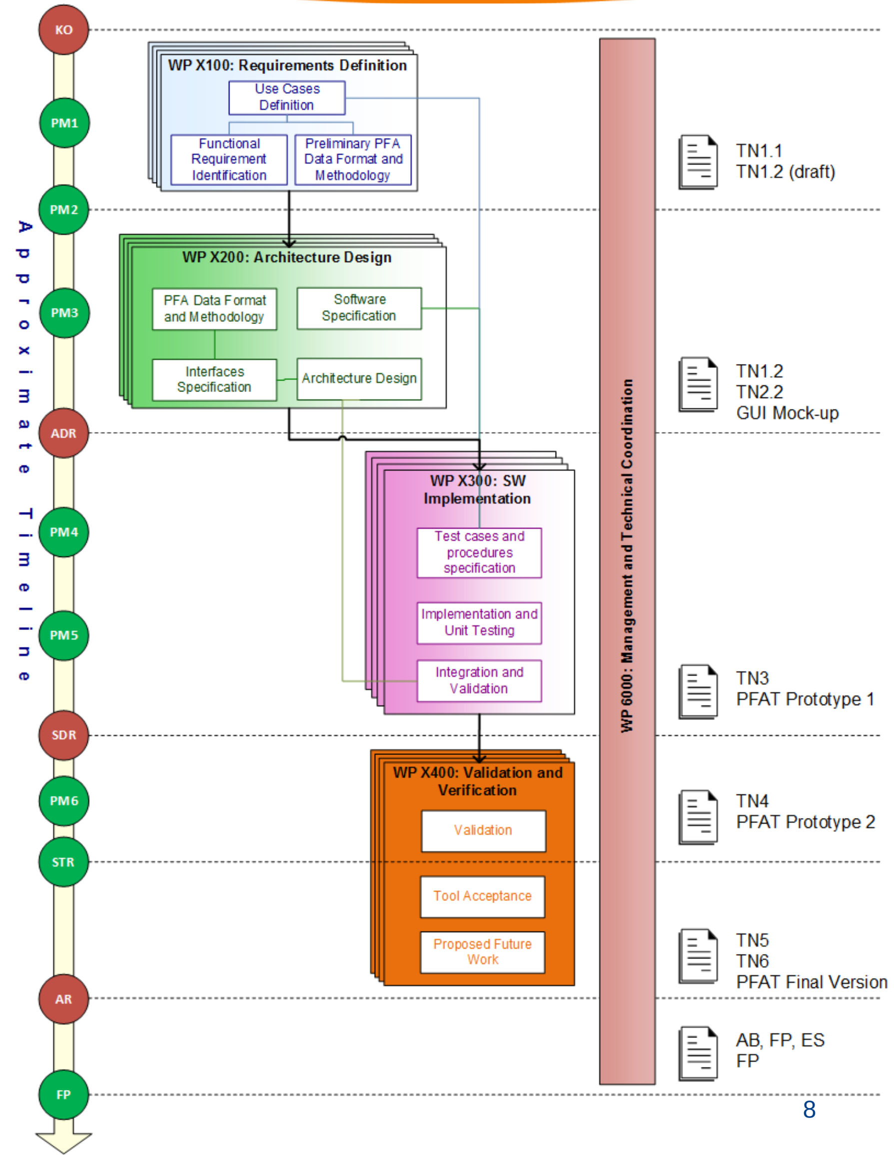
Work Logic

Requirement Definition: use cases definition, functional requirements and preliminary data formats and methodologies derivation

Architecture Design: software specifications from the functional requirements, interfaces specification, data formats and methodologies consolidation, and tool architecture design

SW Implementation: detailed design, analysis and calculation algorithms specification, test cases and procedures specification, SW implementation, unit and integration testing

V&V: verification (calculations are numerically correct) and validation (PFA against the test cases scenarios derived from the use cases)



Meeting Plan

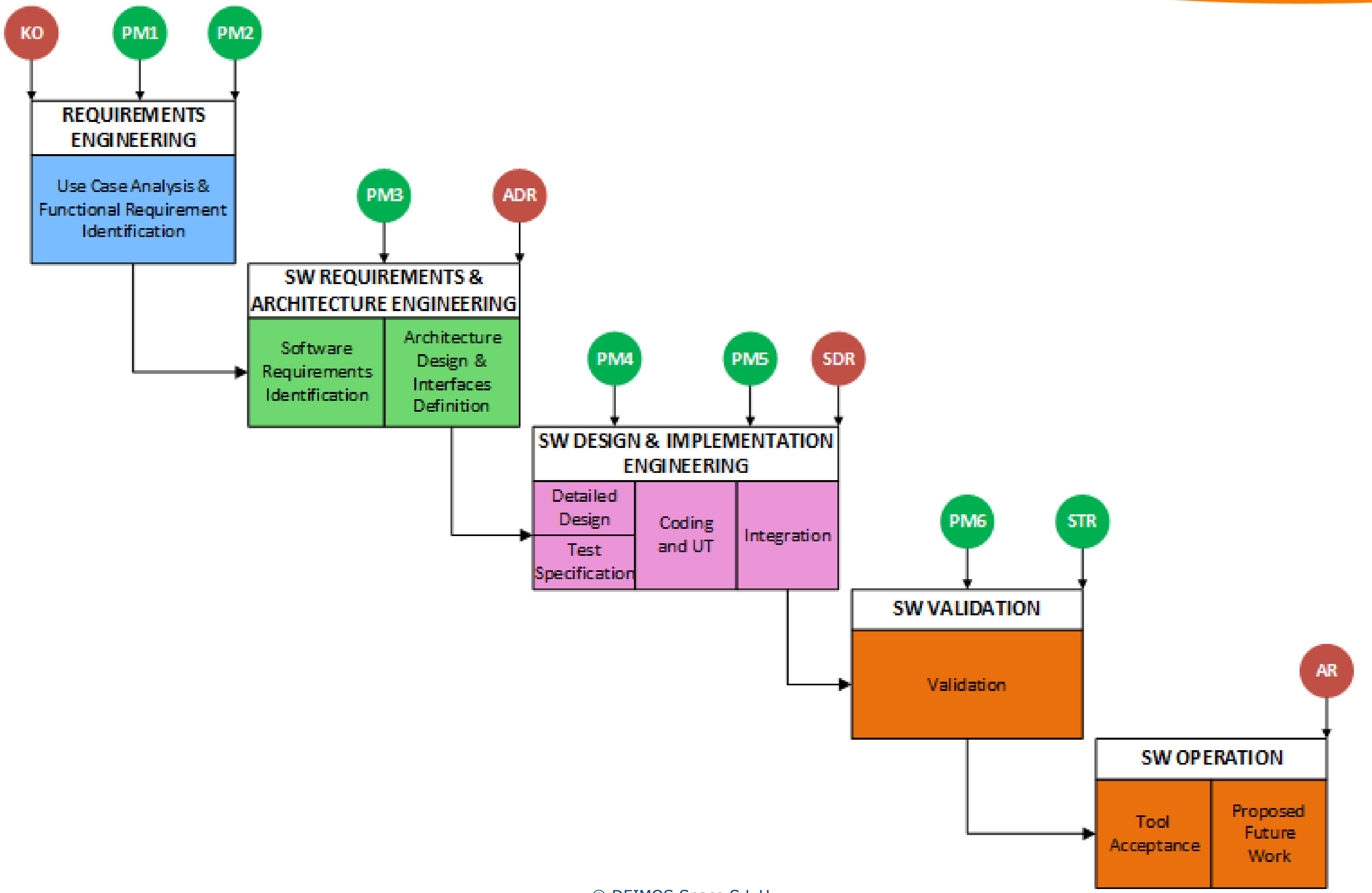
Meeting	Schedule Date	Location	Schedule Date	Meeting Date	DMS	DLR	RUAG	EAI
Kick-off Meeting	T0	ESTEC	03/12/19	03/12/19	Y	TC	TC	TC
Progress Meeting 1	T0+3m	Teleconference	04/03/20	29/04/19	TC	TC	TC	TC
Progress Meeting 2	T0+6m	Teleconference	04/06/20	02/07/20	TC	TC	TC	TC
Progress Meeting 3	N/A	Teleconference	N/A	29/09/20	TC	TC	TC	TC
Architecture Design Review	T0+9m	Teleconference	06/09/20	28/10/20	TC	TC	TC	TC
Progress Meeting 4	T0+12m	Teleconference	04/12/20	20/01/21	TC	TC	TC	TC
Progress Meeting 5	T0+15m	Teleconference	04/03/21	07/04/21	TC	TC	TC	TC
Software Development Review	T0+18m	Teleconference	10/06/21	15/06/21	TC	TC	TC	TC
Progress Meeting 6	N/A	Teleconference	N/A	14/09/21	TC	TC	TC	TC
Software Test Review	T0+22m	Teleconference	05/10/21	16/11/21	TC	TC	TC	TC
Acceptance Review	T0+24m	Teleconference	09/12/21	21/12/21	TC	TC	TC	TC
Final Presentation	T0+24m	Teleconference	09/12/21	08/02/22	TC	TC	TC	TC

Deliverable Item List

ID	Title	Due for	Comments
TN1.1	Use Case Analysis & Functional Requirement Identification	PM1	Updated at STR
TN1.2	Post Flight Analysis Data Format and Methodology	PM1 (draft), ADR (final)	Updated at STR
TN2	Architecture Design	ADR	Updated at STR
TN3	Software Development Document	SDR	Upgrade of TN1.2, TN2
TN4	Validation and Testing Document	AR	Pre-delivered at STR
TN5	Software Manual & Tutorials	AR	
TN6	Future Developments Roadmap	AR	
TDP	Technical Data Package	FP	Pre-delivered at AR
AB	Abstract	FP	Pre-delivered at AR
TAS	Technology Achievement Summary	FP	Pre-delivered at AR
FP	Final Presentation	FP	Pre-delivered at AR
ESR	Executive Summary Report	FP	Pre-delivered at AR
FR	Final Report	FP	Pre-delivered at AR
GUI	GUI non-functional mock-up	ADR (within TN2)	
SW1	PFAT Software (Prototype)	SDR	
SW2	PFAT Software (Prototype)	STR	
SW3	PFAT Final Software	AR	

2 Software Development

PFAT Life Cycle



System Requirements and Software Specifications

34 Use Cases

↕

43 Functional Requirements

↕

58 Software Specifications

Architectural Design

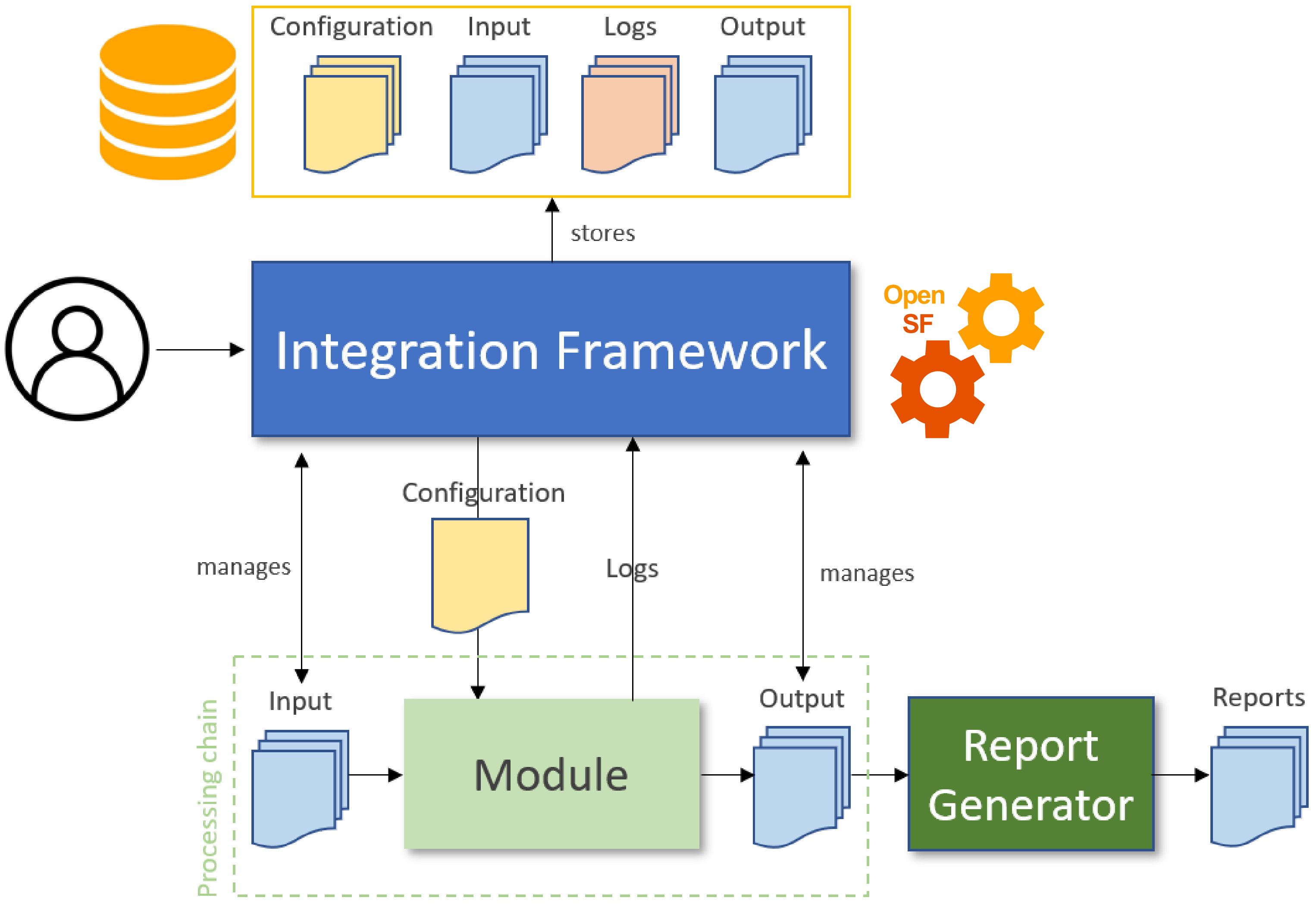
- PFA algorithms identification and specifications
- External interfaces identification and specifications
- High-level system design, allocating algorithms/interfaces

Multi OS desktop SW with four different functional components

Functional components

- **Computational core**: six Python modules, related with the four engineering domains plus data processing and an auxiliary module with common functionalities.
- **Executable modules**: interfaced with the GUI/CLI, exposing the functionalities
- **Graphical User Interface**: relying on the ESA openSF integration framework
- **Common Data Structure**: the glue between the different modules

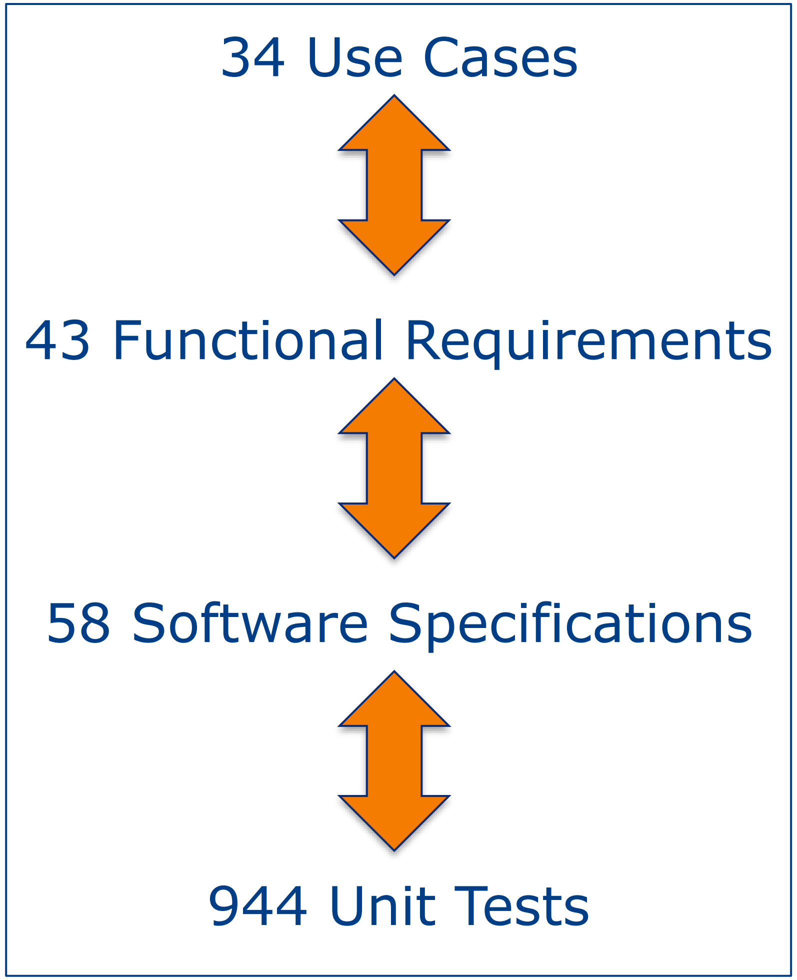
PFAT System Decomposition



Unit Testing

Comprehensive verification with unit testing

- 944 unit tests on 761 functions
- 87% code coverage
  - *94% excluding the two Python UI*
- Tests executed in
  - *Windows 10*
  - *Linux Ubuntu 20.04 LTS*
  - *macOS 11.0 Big Sur*



3 PFAT Capabilities

Common Data Structure

Common interface to manage any data set used in PFAT

- Agnostic data management among different engineering domains
- Reduces the time that the user needs to familiarize with PFAT
- Maximizes the reusability of the functions/algorithms, which is especially relevant for the more generic algorithms such as the data processing ones
- Based on Pandas
- Store data series with one independent and multiple dependent variables or data series
- Stores points coordinates, the connectivity information, the element type, and other data
- Stores metadata
- Exposes a programmatic API
- I/O to and from binary, to JSON, to CSV files

Generic functionalities on the CDS data

- Concatenation
- Gap detection, gap filling
- Noise quantification
- Radar altitude estimation
- IMU/INS/GNSS processing
- Data replacement
- Measurements removal by index or by interval
- Data smoothing by moving average, moving polynomial or moving exponential
- Frequency resampling, up-sampling or down-sampling

Other generic functionalities

- I/O to and from binary, to JSON, to CSV files and from a series of external formats
  - *Telemetry in Tabular format*
  - *External tools: OEM/AEM, TAU, TDMS, Tecplot*
- CDS diff (whole or subset, with or without resampling)
- Plot generation (GUI enabled)
  - *By regular expression, By point identifier, or By label and point identifier*
  - *Interactive HTML, PNG, JPG*

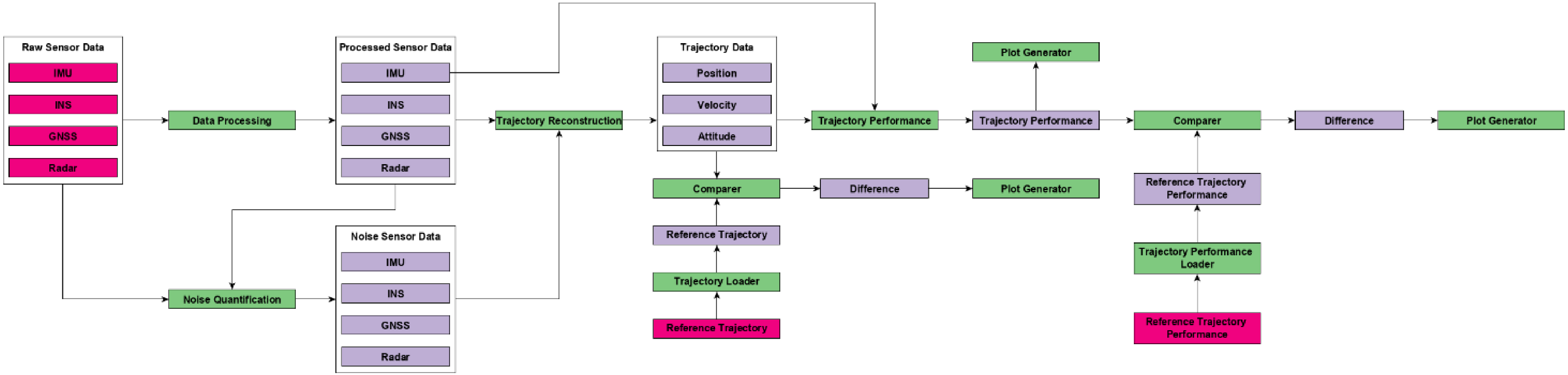
Trajectories Domain

Telemetry data loading, pre-processing and noise quantification

Trajectory reconstruction by means of a multiplicative extended Kalman filter

Reference trajectory loader and performances calculation

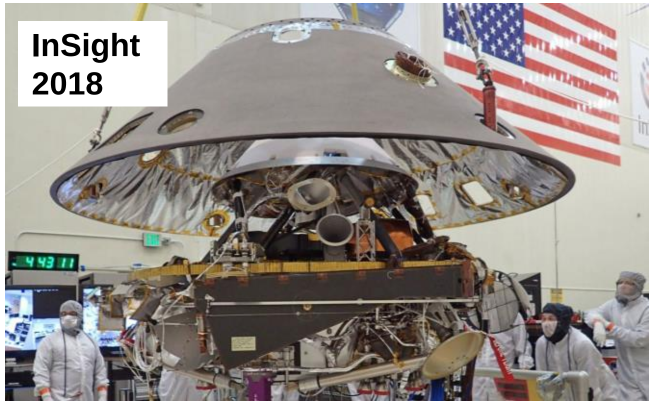
Post-processing and report generations



Trajectory Reconstruction – Post-flight Analysis Test Case

Use available **InSight** mission flight data for Mars re-entry and descent trajectory

- True trajectory is unknown, but the EKF results can be compared with the literature and in-house tool
- Flight dataset to completely validate PFAT data-fusion (IMU + GNSS pos/vel + RDA)
  - IMU measurement → public
  - Radar measurement → from literature
  - GNSS measurement → landing site measurement from literature



	InSight
IMU measurements	public
radar measurements	Karlgaard Fig. 5
GNSS measurements	Karlgaard
IMU reconstruction in literature	yes
EKF reconstruction in literature	yes

Staged approach towards **IMU + Radar + Landing Site** reconstruction

CASE 1 - IMU only reconstruction

- Validate IMU data integration and uncertainty propagation vs in-house tool
- Reference trajectory loader and comparer used
- Trajectory performance used to computed derived variables and uncertainty
- Reference trajectory performance loader

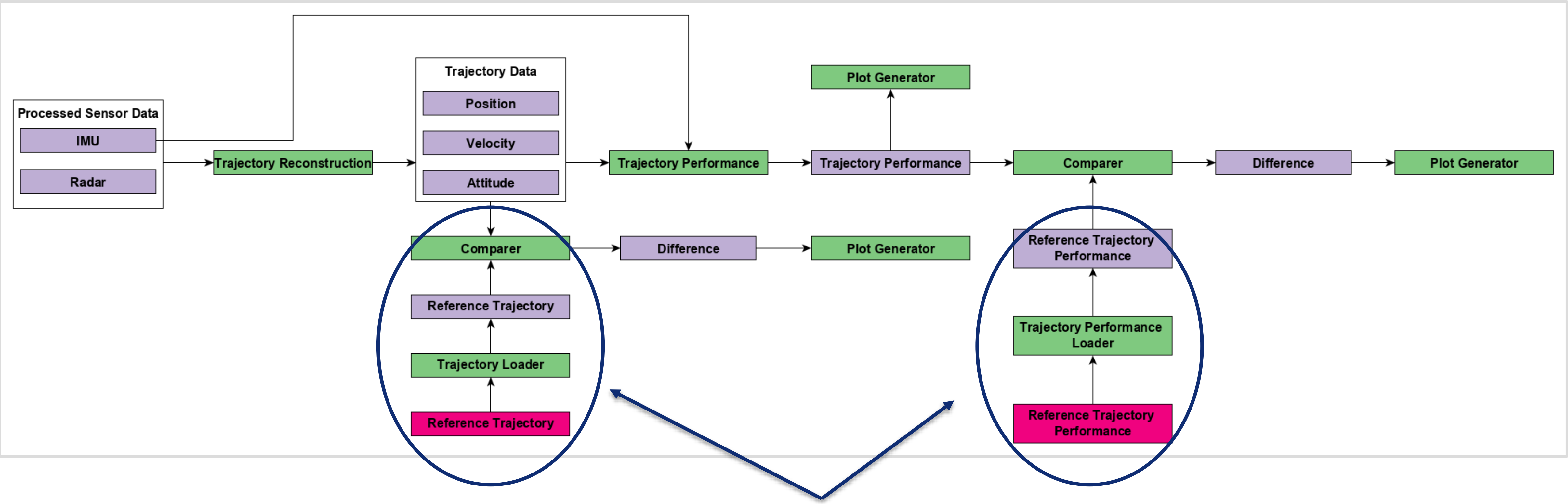
CASE 2 - IMU + Radar + Landing Site reconstruction

a) **Forward only** reconstruction to analyze data fusion

b) **Forward and backward** reconstruction comparing with Karlgaard

  - Trajectory reconstruction + Trajectory performance
  - Reference solution from literature is only graphical

IMU Forward Reconstruction Processing chain



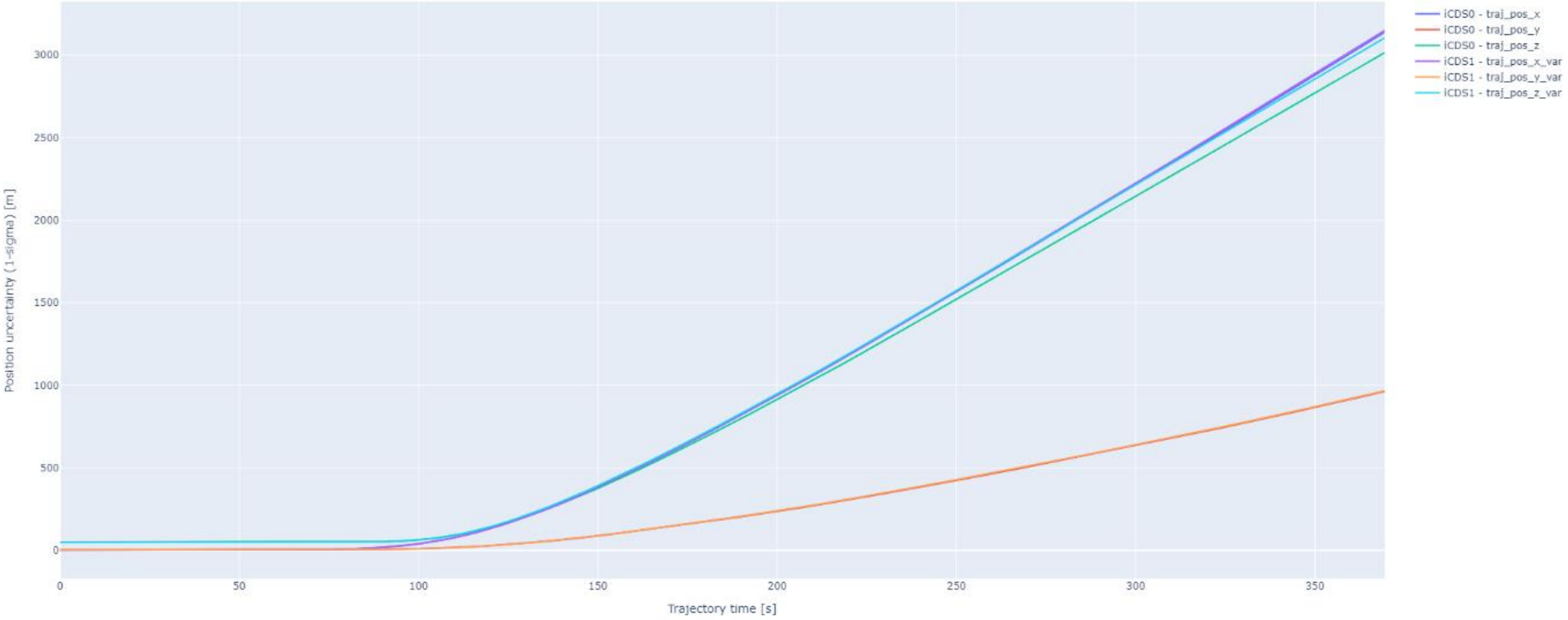
In-house tool solution provides a source of comparison for trajectory and uncertainty evolution

Comparison with reference reconstruction from external tool

- Propagated state/attitude is near-exact
  - Maximum averaged error in position / velocity: ~1e-7 / ~1e-5
- Uncertainty evolution will present differences
  - Different approaches for uncertainty propagation: Kalman filter vs. Monte Carlo
  - Similar trends are expected, but not perfectly matching results

CASE 1 - IMU Forward Reconstruction

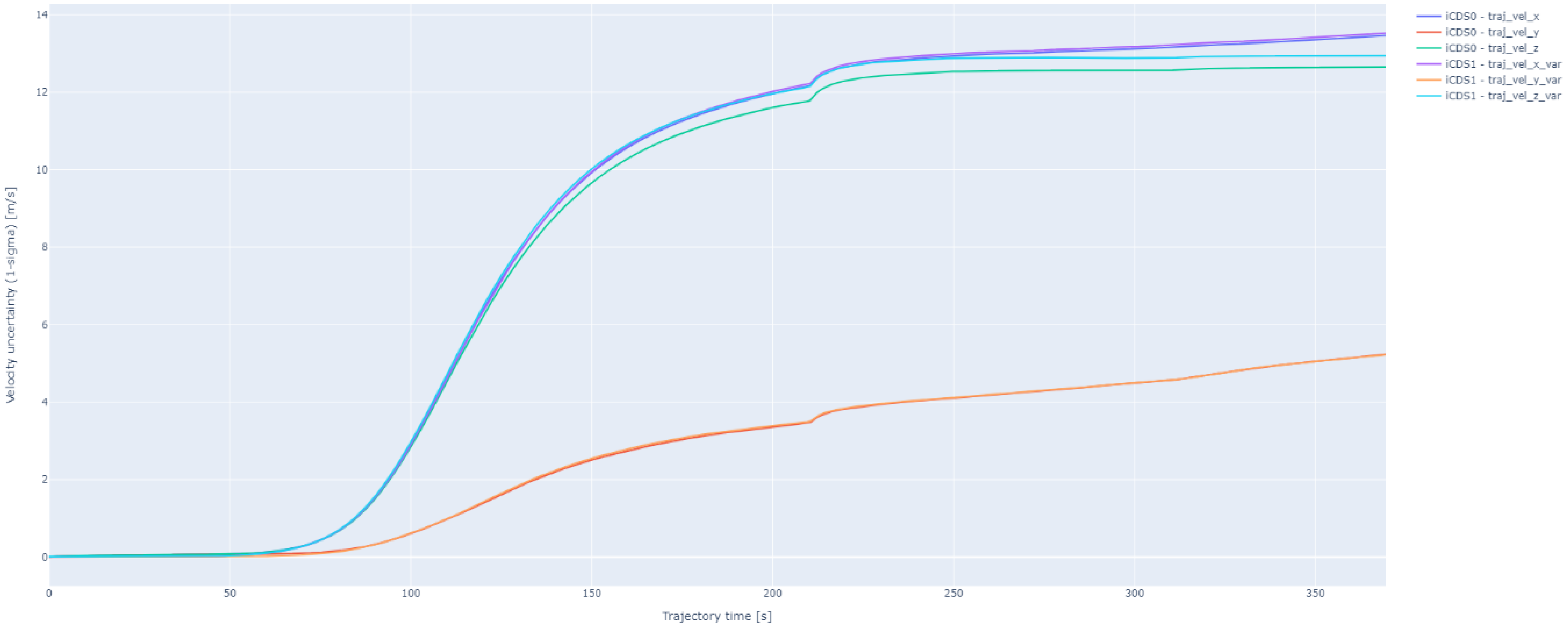
Reconstructed and reference position uncertainty evolution



- iCDS0: reconstructed
- iCDS1: reference

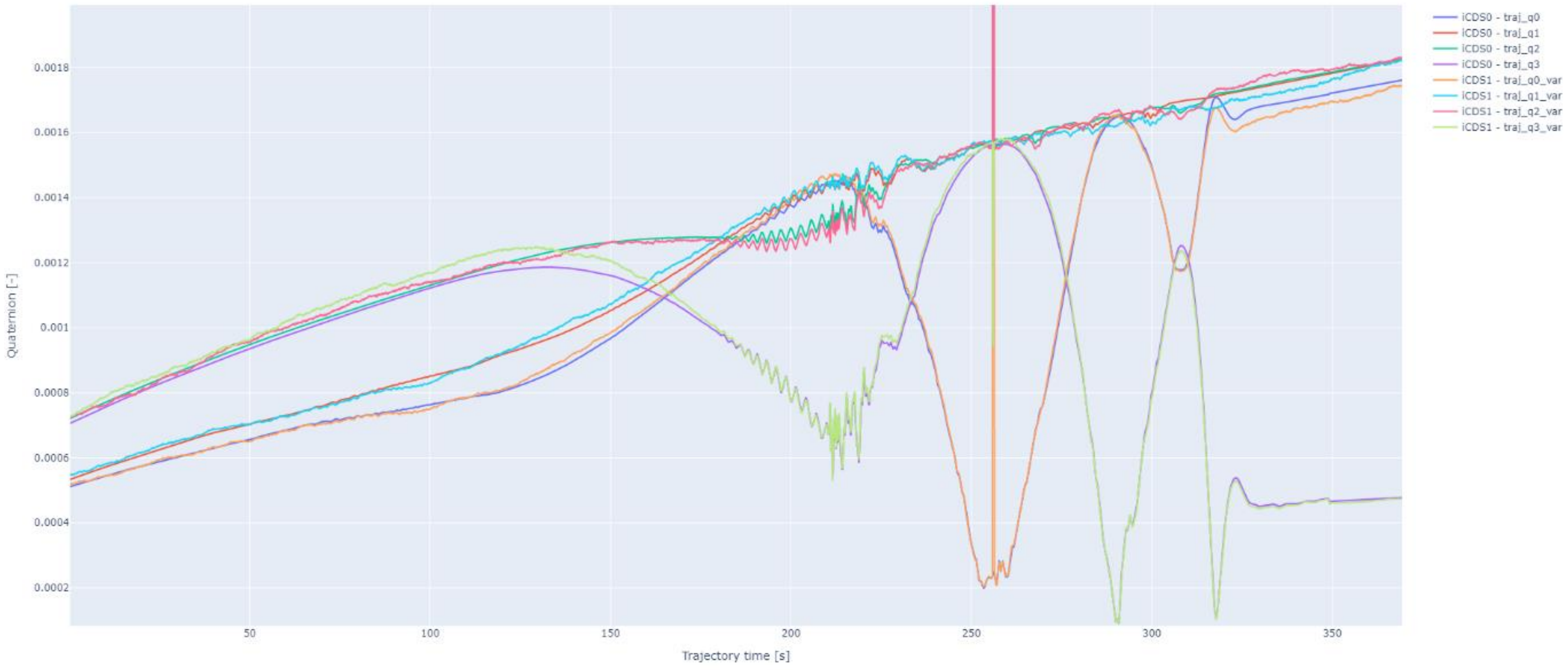
CASE 1 - IMU Forward Reconstruction

Reconstructed and reference velocity uncertainty evolution



- iCDS0: reconstructed
- iCDS1: reference

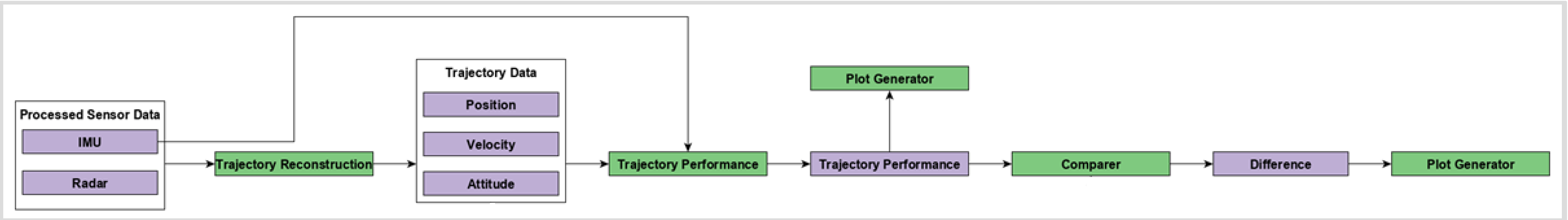
Reconstructed and reference attitude uncertainty evolution



- iCDS0: reconstructed
- iCDS1: reference

IMU + Radar + Landing Site Reconstruction Processing chain

- No numerical reference solution is given for this case
  - Comparers are not included
- Trajectory performance is mainly included to assess altitude evolution



CASE 2a - IMU + Radar + Landing Site Forward Reconstruction

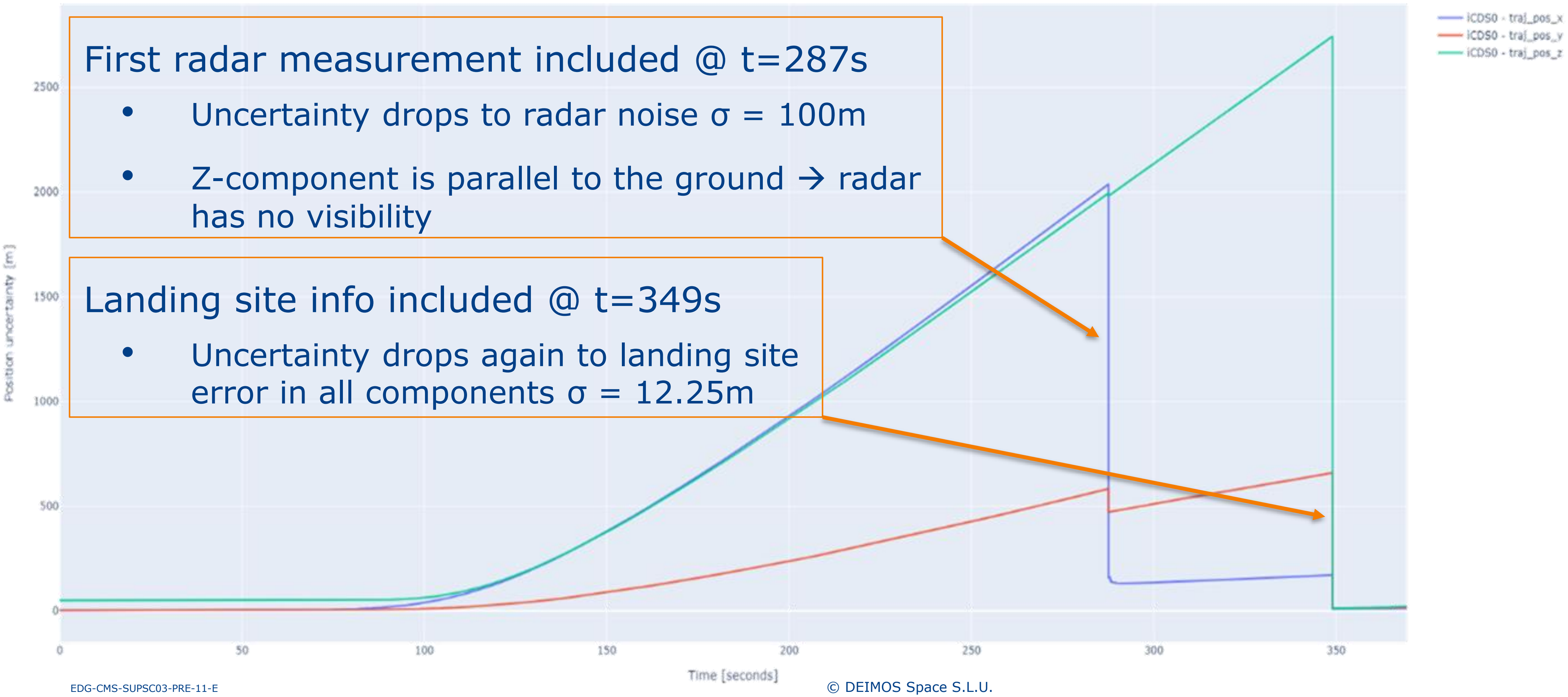
Reconstructed position uncertainty evolution

First radar measurement included @ t=287s

- Uncertainty drops to radar noise σ = 100m
- Z-component is parallel to the ground → radar has no visibility

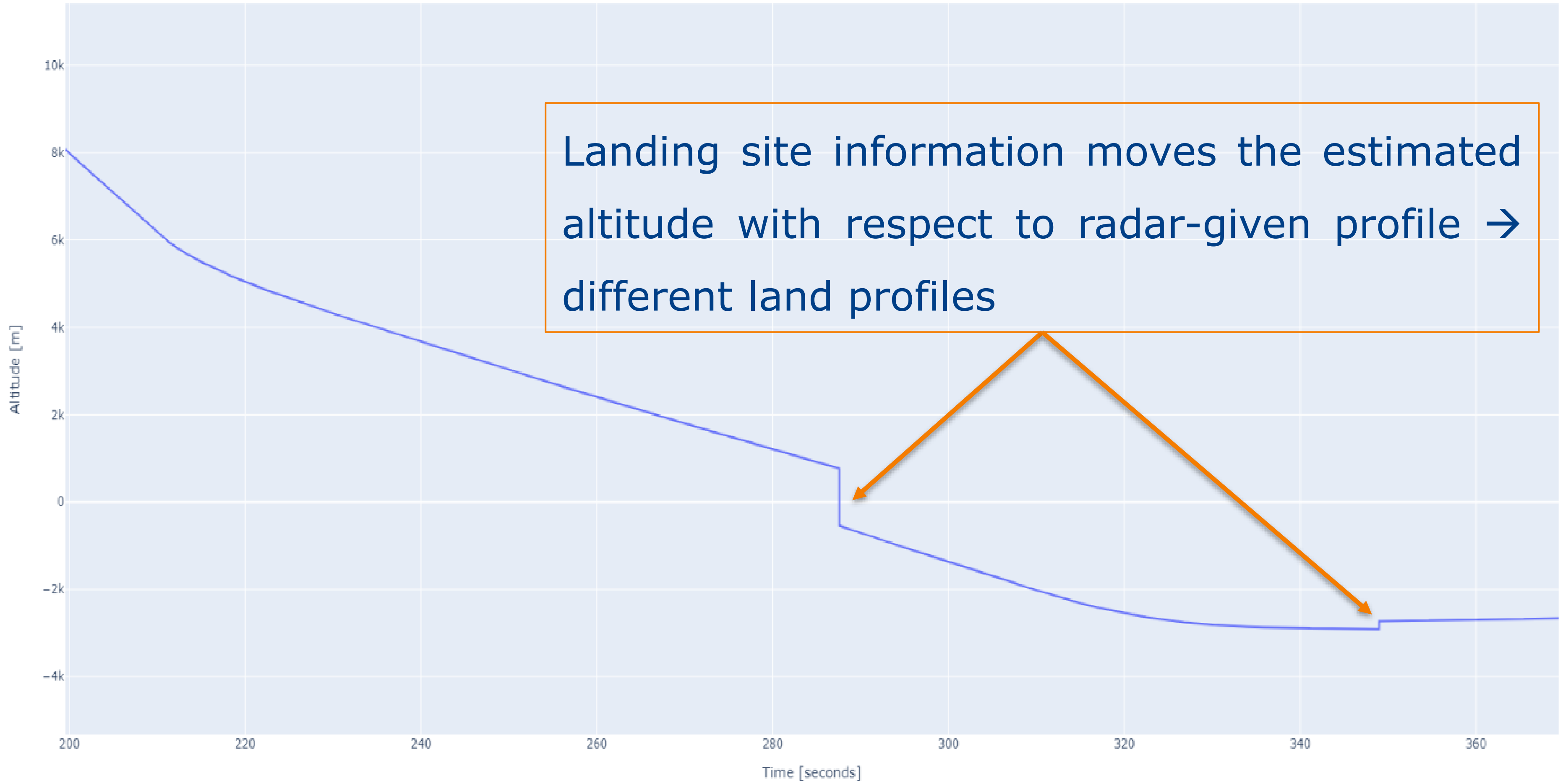
Landing site info included @ t=349s

- Uncertainty drops again to landing site error in all components σ = 12.25m



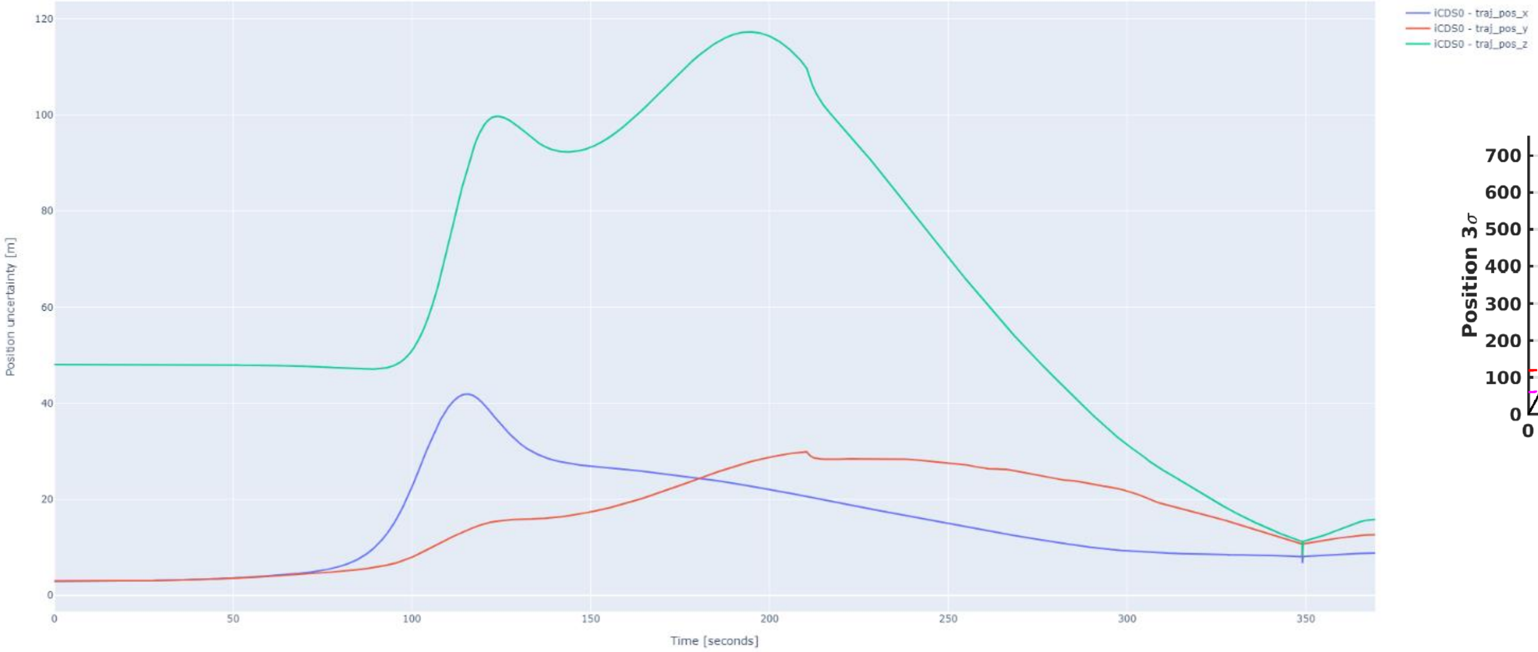
CASE 2a - IMU + Radar + Landing Site Forward Reconstruction

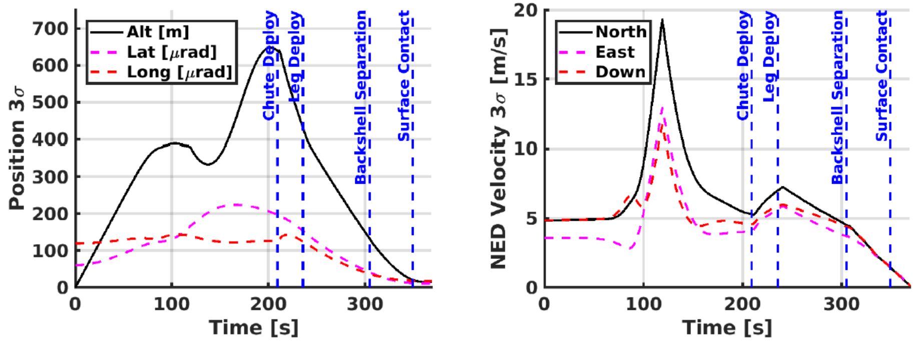
Reconstructed altitude evolution



Reconstructed position evolution uncertainty

- Uncertainty profile is similar (peaks at high dynamic pressure correspond in time) and uncertainty values are close (reference plot is 3-σ, PFAT plot is 1-σ)
- Differences are mainly due to different initial uncertainty and measurements noise modelling





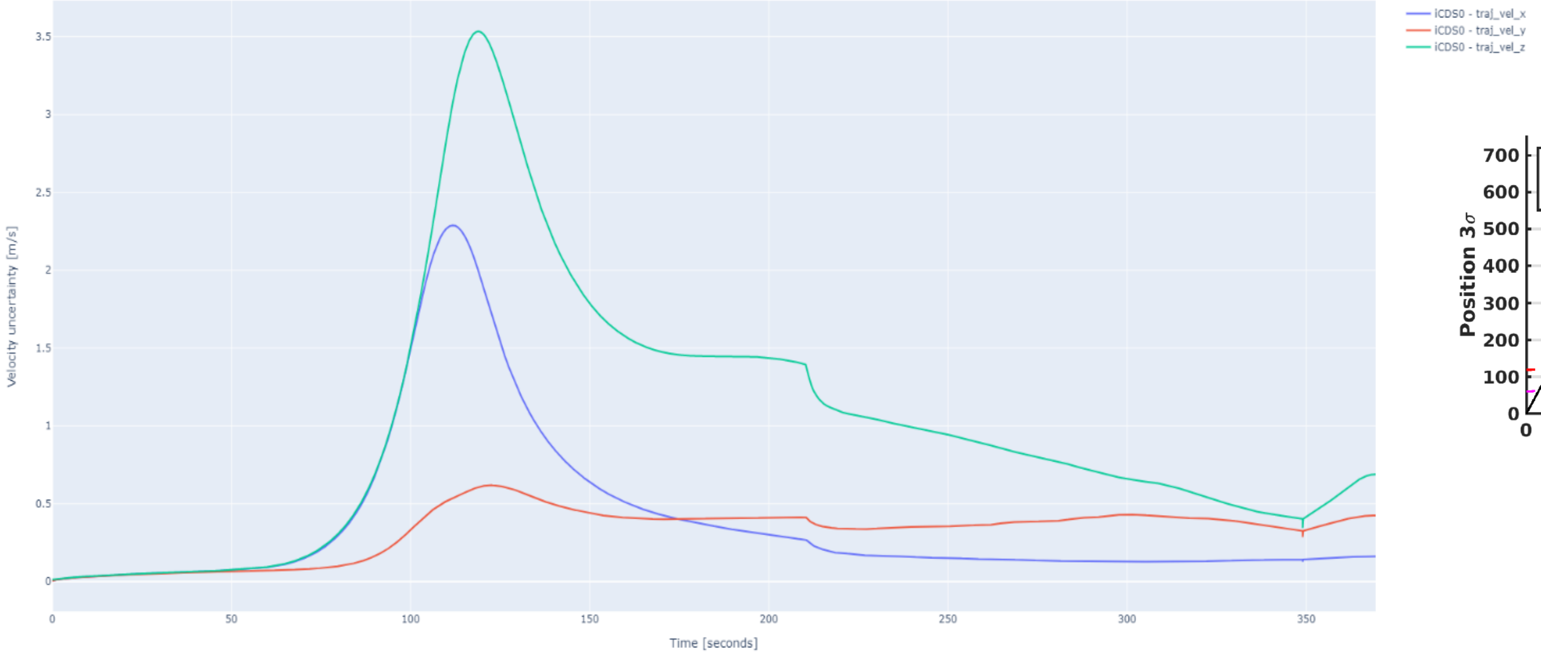
(a) Position (b) Velocity

Fig. 10 Reconstructed Position and Velocity Uncertainties

CASE 2b - IMU + Radar + Landing Site Forward&Backward Reconstruction

Reconstructed velocity evolution uncertainty

- Uncertainty profile is similar (peaks at high dynamic pressure correspond in time) and uncertainty values are close (reference plot is 3-σ, PFAT plot is 1-σ)
- Differences are mainly due to different initial uncertainty and measurements noise modelling



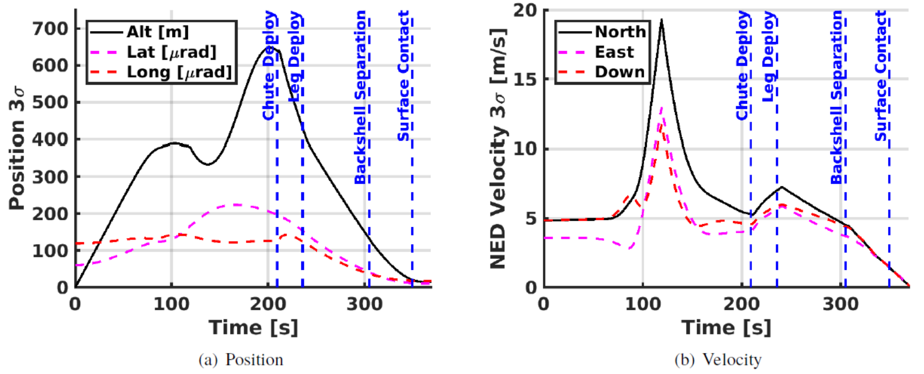
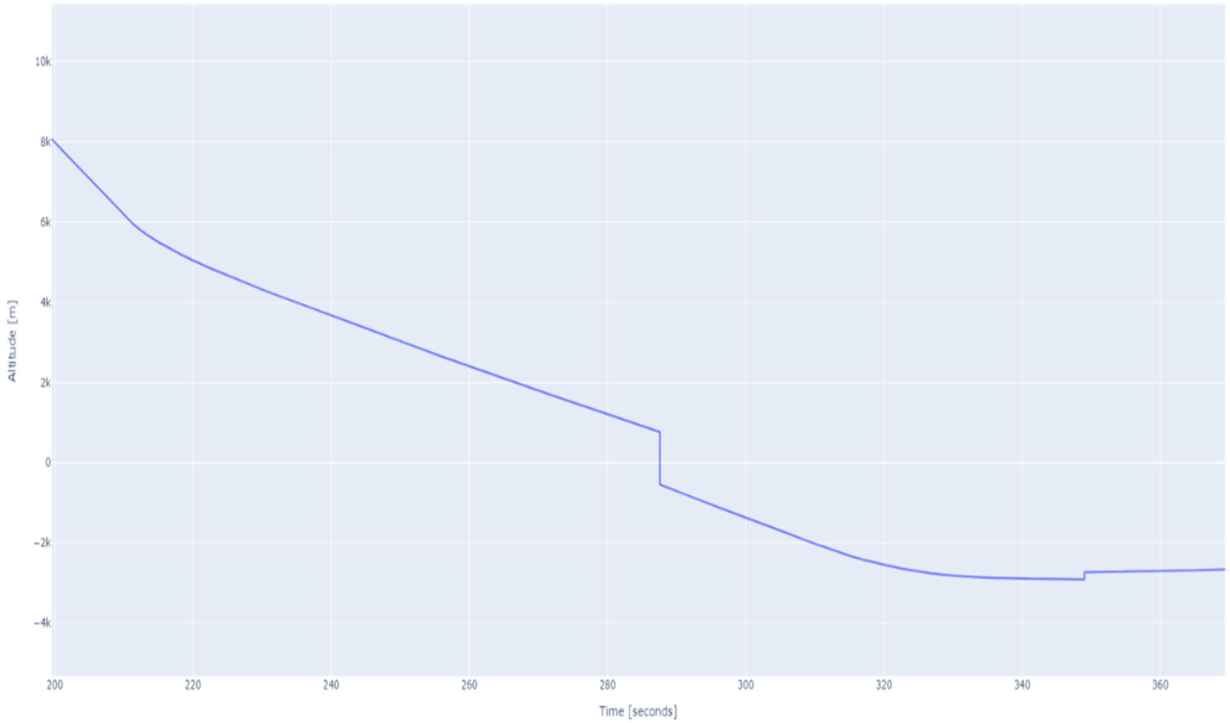


Fig. 10 Reconstructed Position and Velocity Uncertainties

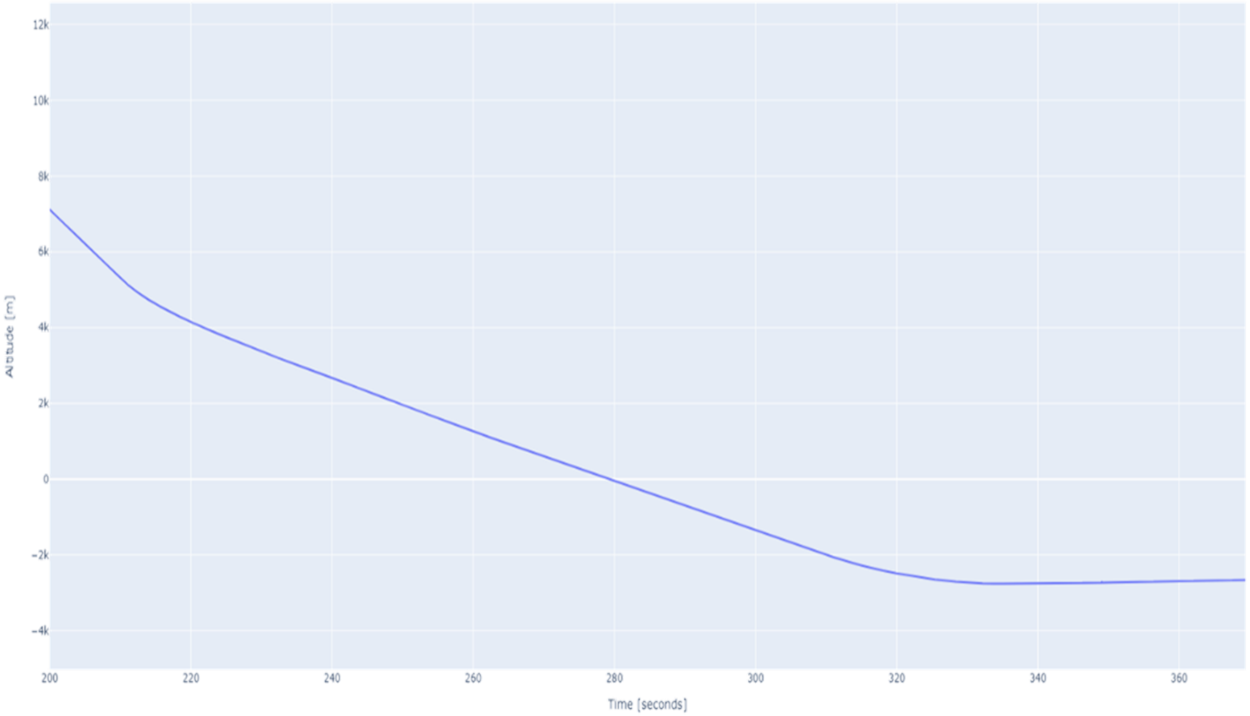
Reconstructed altitude evolution

- Altitude profile is smoothed thanks to the forward + backward approach that includes information “from the future”

Forward only



Forward & Backward



Aerothermodynamic Domain

❑ Data loading

- Read data from flight tests, ground tests and simulations
- Read data series, grid data and meta data
- Supported file formats
  - ➢ *TDMS files (data series)*
  - ➢ *Tecplot files (data series and grid data)*
  - ➢ *TAU files (grid data)*
- Automatic data scaling for TDMS files (here: linear scaling)
- Load just selected zones or variables
- Load specific parts of TAU data (surface, sublayer, flow field)

Data harmonization

- Interpolation of data series
- Interpolation of grid data
- Interpolation of data sets

Basic data processing

- Filtering (here: Savitzky–Golay filter)
- Arithmetic calculations (data series and grid data, from multiple data sets, including meta data)
- Statistical analysis of data series and data sets (e.g.: mean, variance, interpercentile)
- Data reduction of data series (here: power spectral density and probability density function)

Aerothermodynamic Domain

❑ Specific data processing

- Complex computations of data series (here: solve 1D heat equation)
- Surface integrals (e.g. pressure to forces)
- Boundary layer integrals (e.g. momentum thickness)

❑ Uncertainties

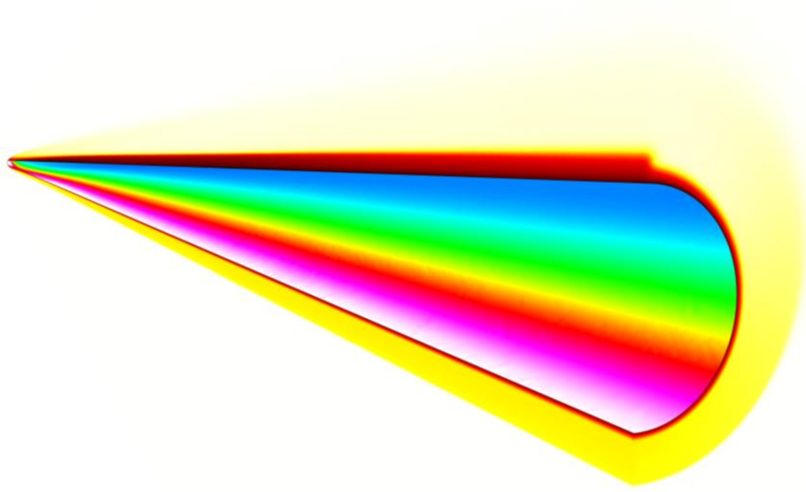
- Uncertainties calculator (based on equations and/or statistical analysis)
- Uncertainties propagation (here: Monte Carlo algorithm)

Aerothermodynamic Domain

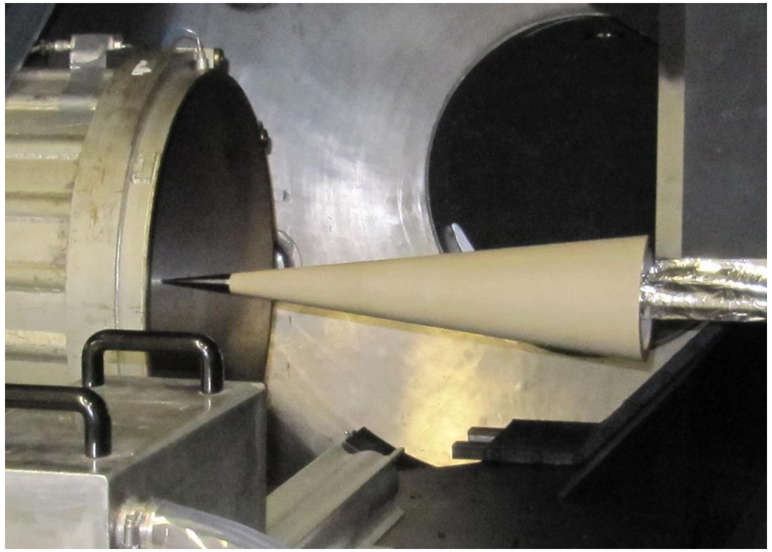
- Modules may serve as blue prints for similar algorithms
- If feasible modules check and process units with a specific submodule
- Parameters of algorithms can be adjusted by user
- Modules can be combined freely to create complex processing chains

Aerothermodynamic Domain

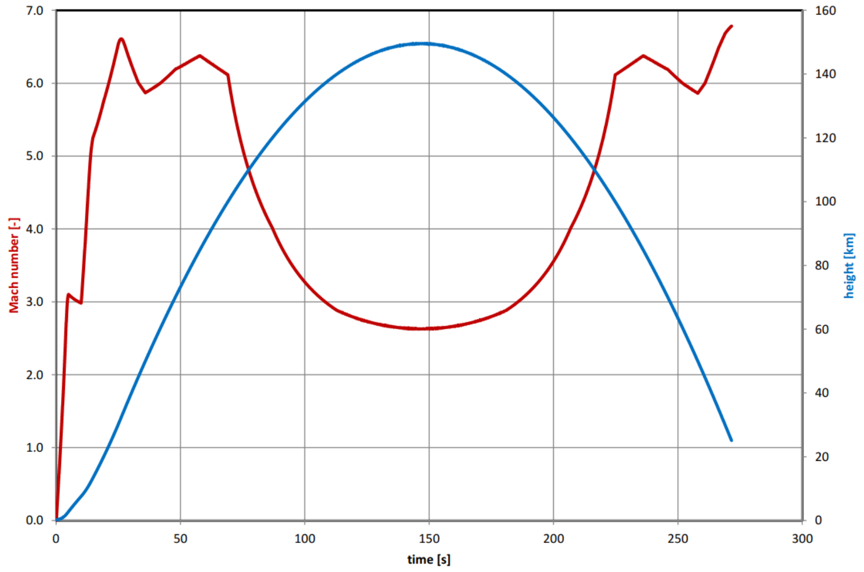
❑ Validation done with ROTEX-fake data



Simulation data



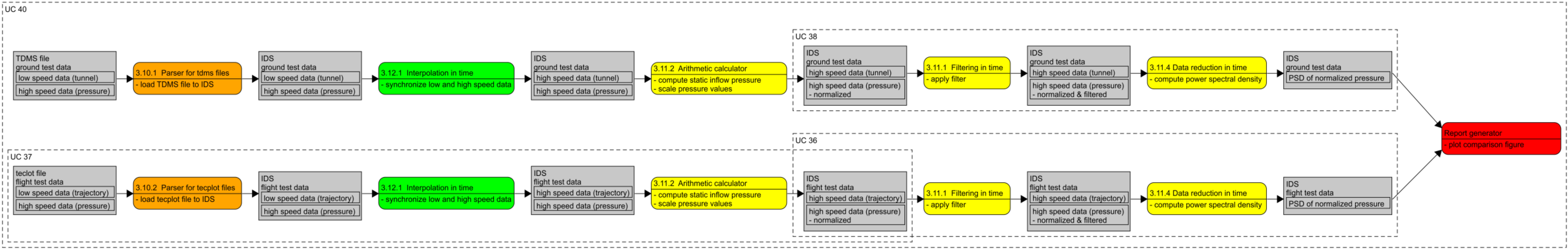
Ground test data



Flight test data

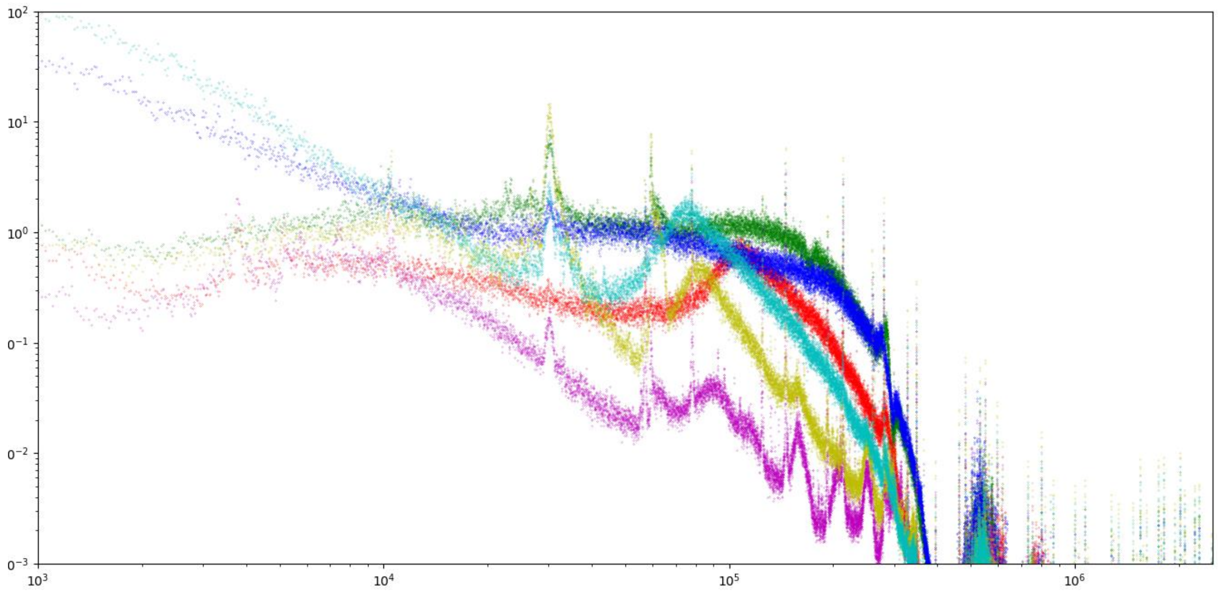
PSD pressure processing chain

- Read data series from tecplot files.
- Read data series from tdms files including automatic scaling of sensor data
- Interpolation in time / synchronization of data sets
- Perform arithmetic calculations with data series
- Filtering in time (Savitzky-Golay)
- Spectral analysis (PSD)

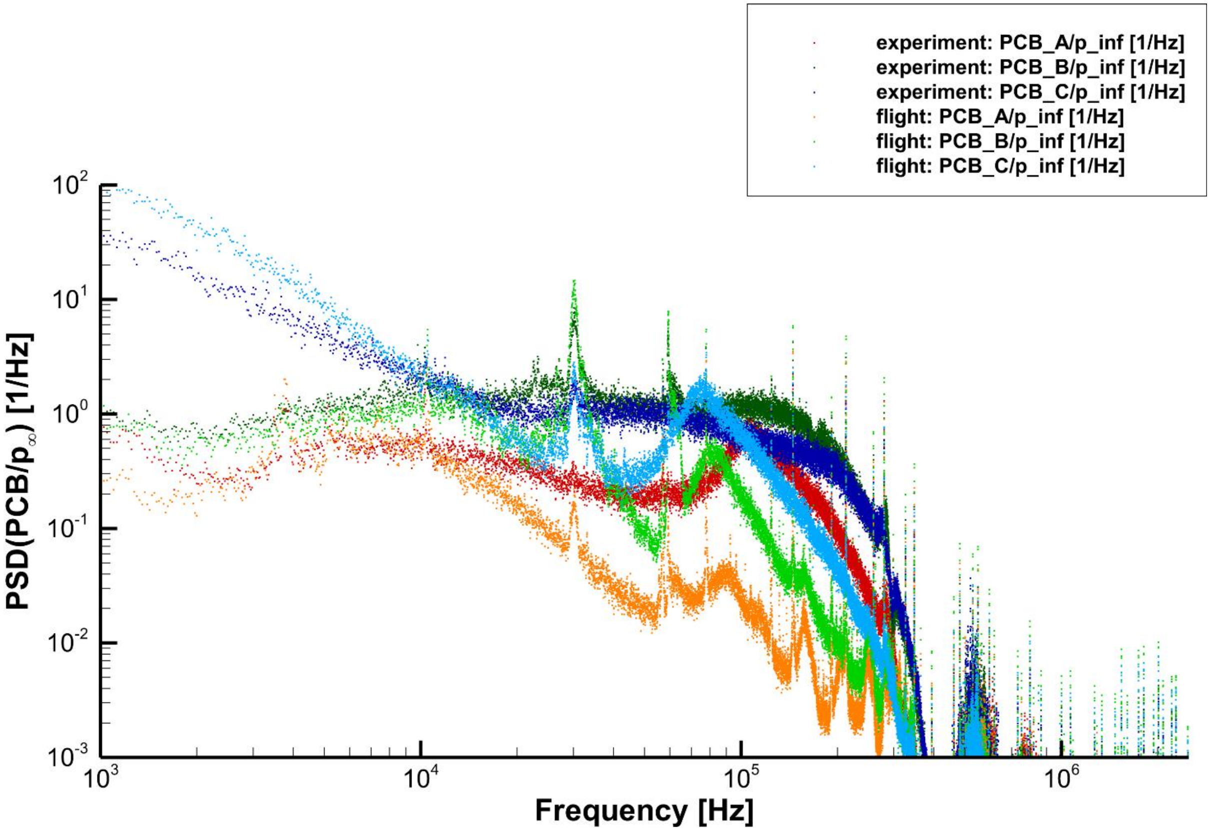


PSD pressure processing chain

Output by PFAT

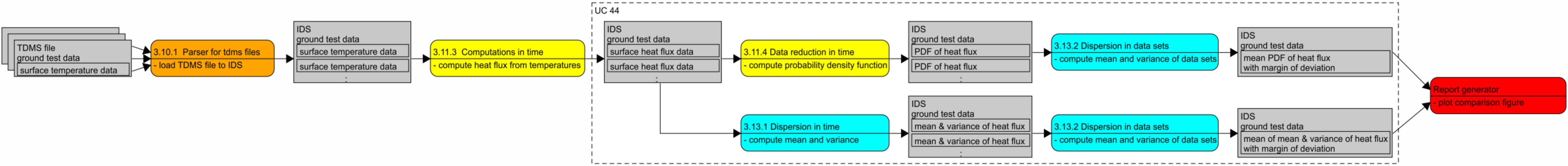


Output by Labview



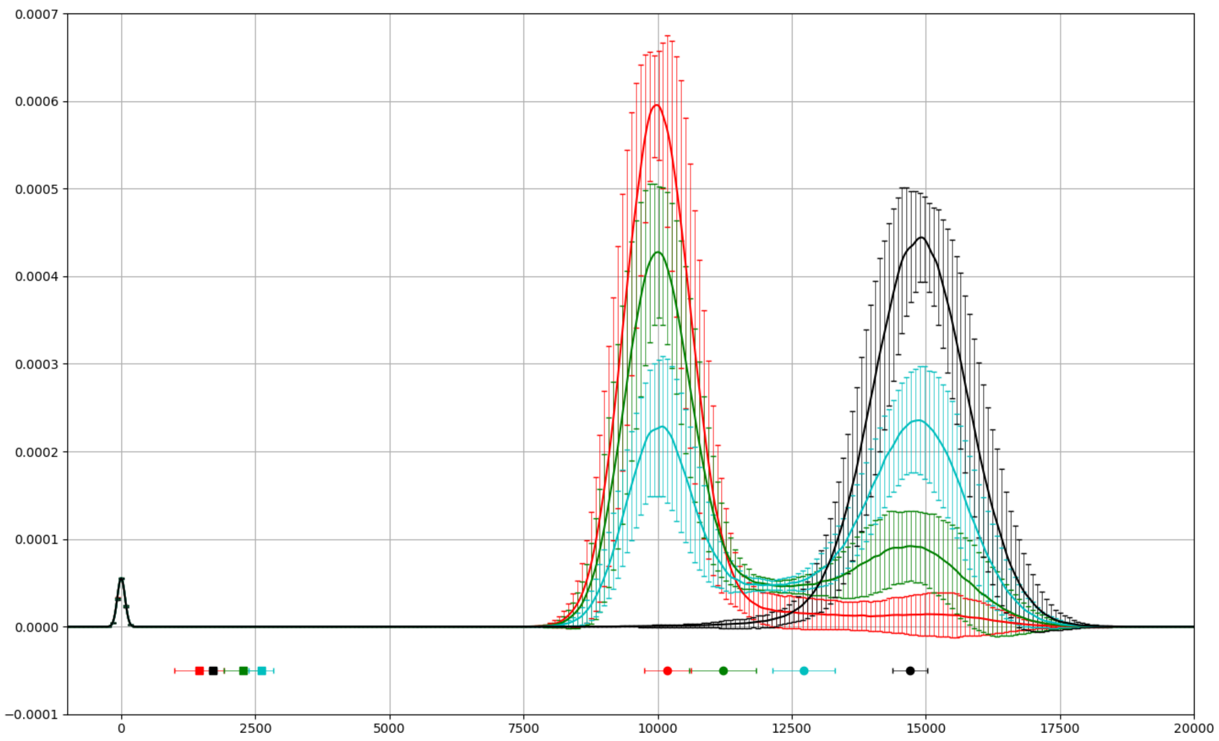
PDF heat-flux processing chain

- Read data series from tdms files
- Complex computations in time (1D heat equation)
- Statistical values of data series
- Statistical analysis (PDF)
- Statistical values of data sets

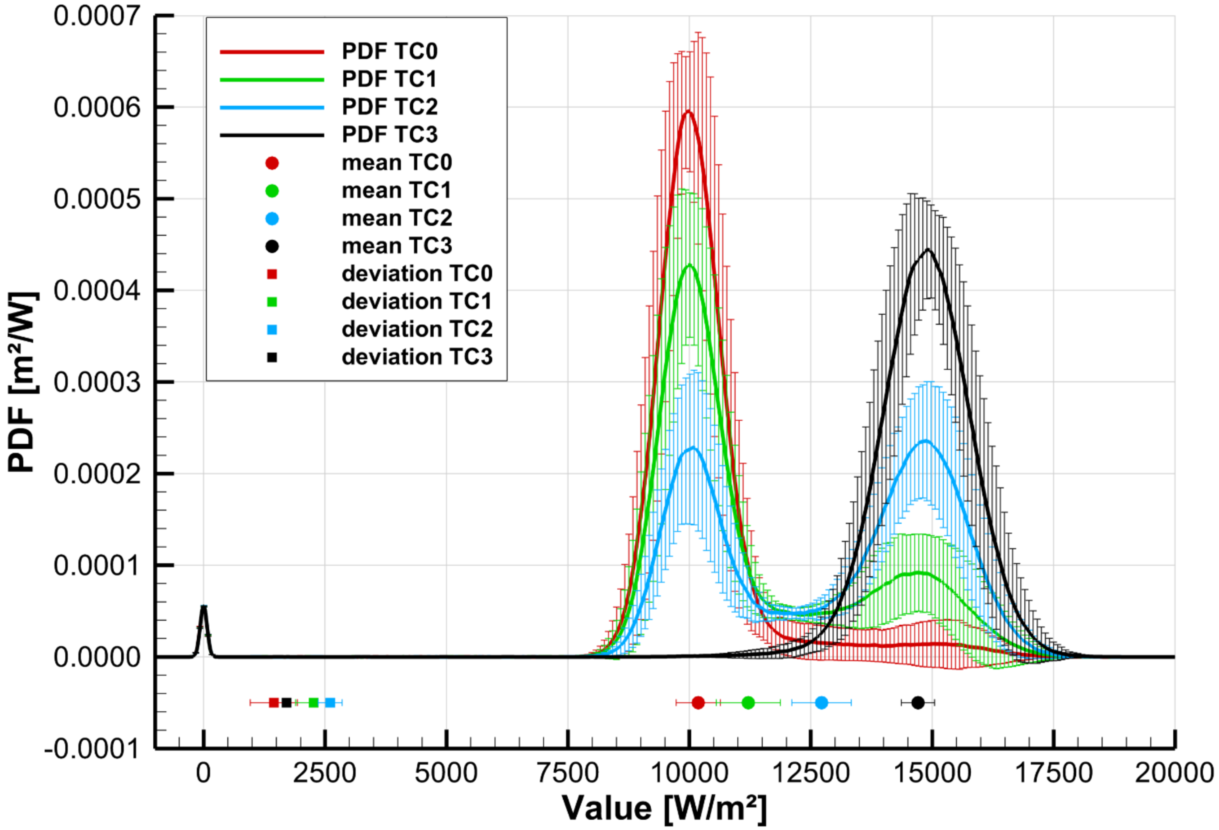


PDF heat-flux processing chain

Output by PFAT

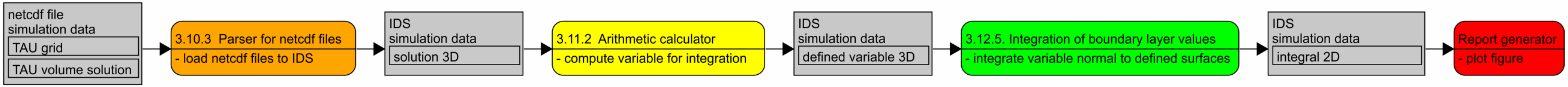


Output by Labview



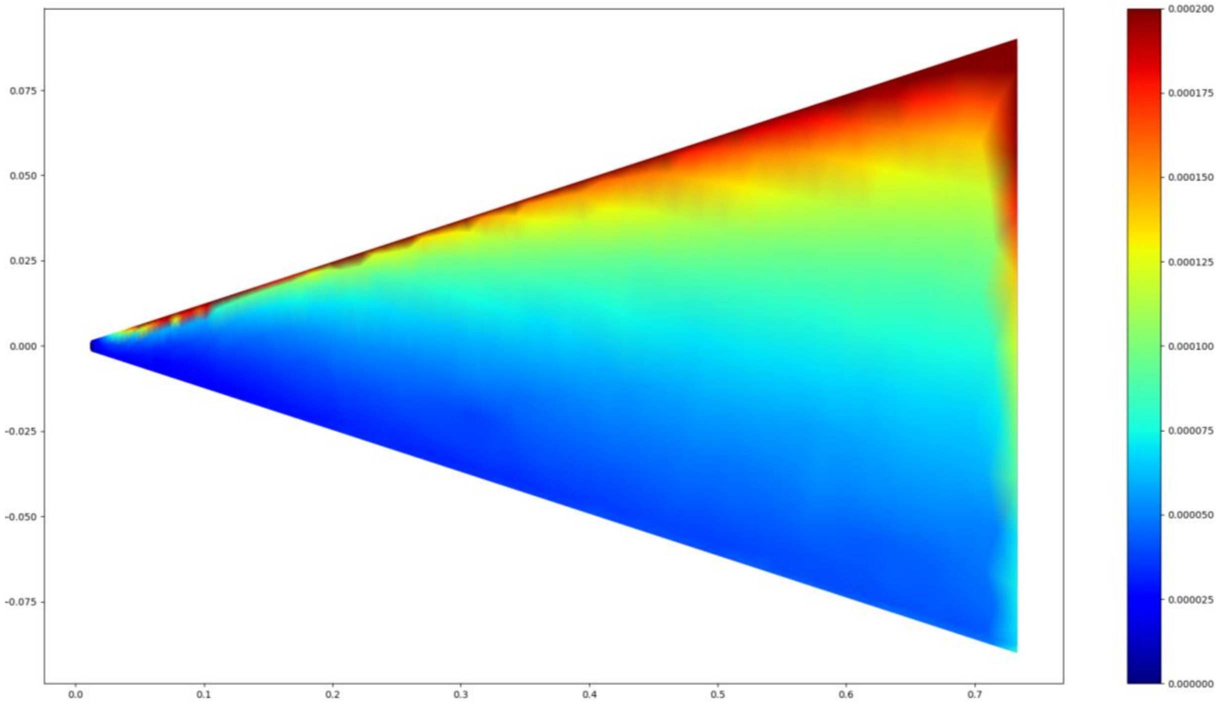
Boundary layer integral processing chain

- Read 3D volume data from netcdf files including grid information
- Perform arithmetic calculations with grid data
- Calculate boundary layer integrals

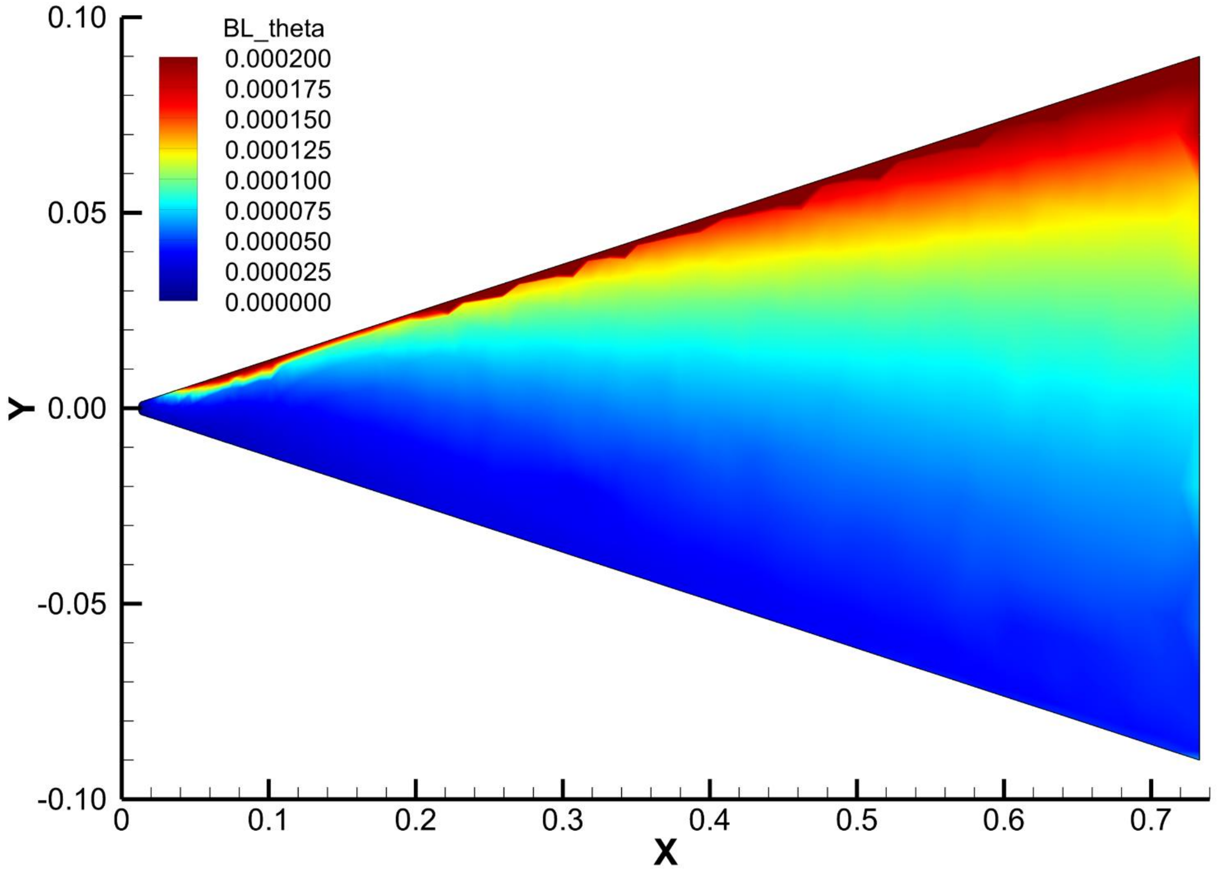


Boundary layer integral processing chain

Output by PFAT

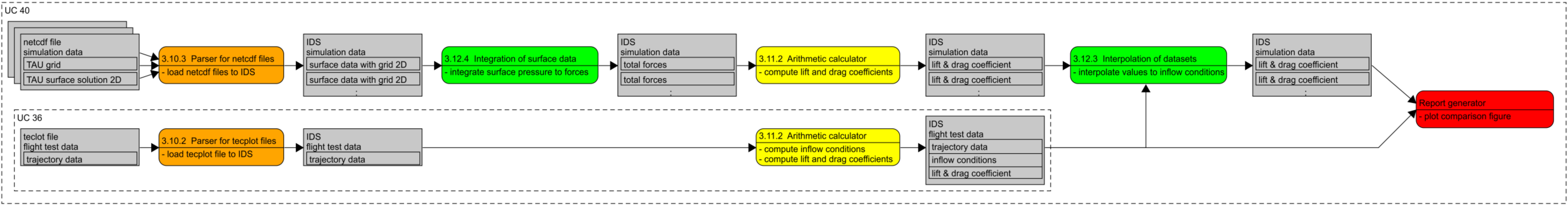


Output by Fortran



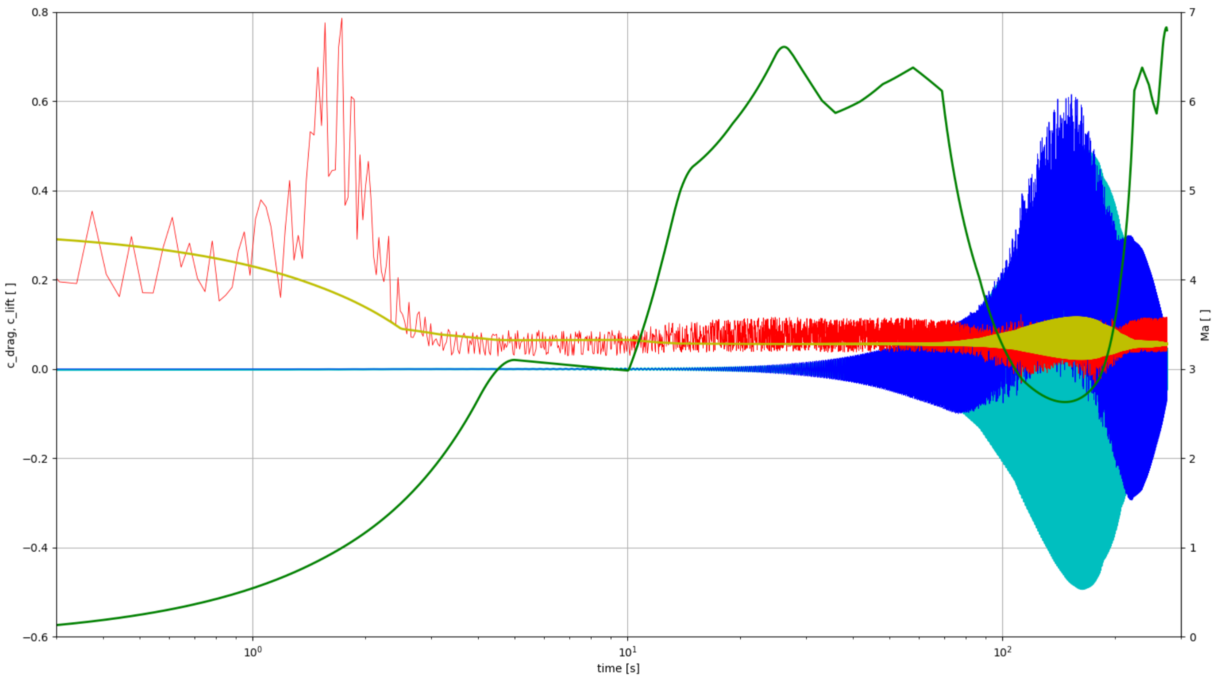
Lift and drag processing chain

- Read 3D surface data from netcdf files including grid information
- Calculate surface integrals
- Perform interpolation of data sets

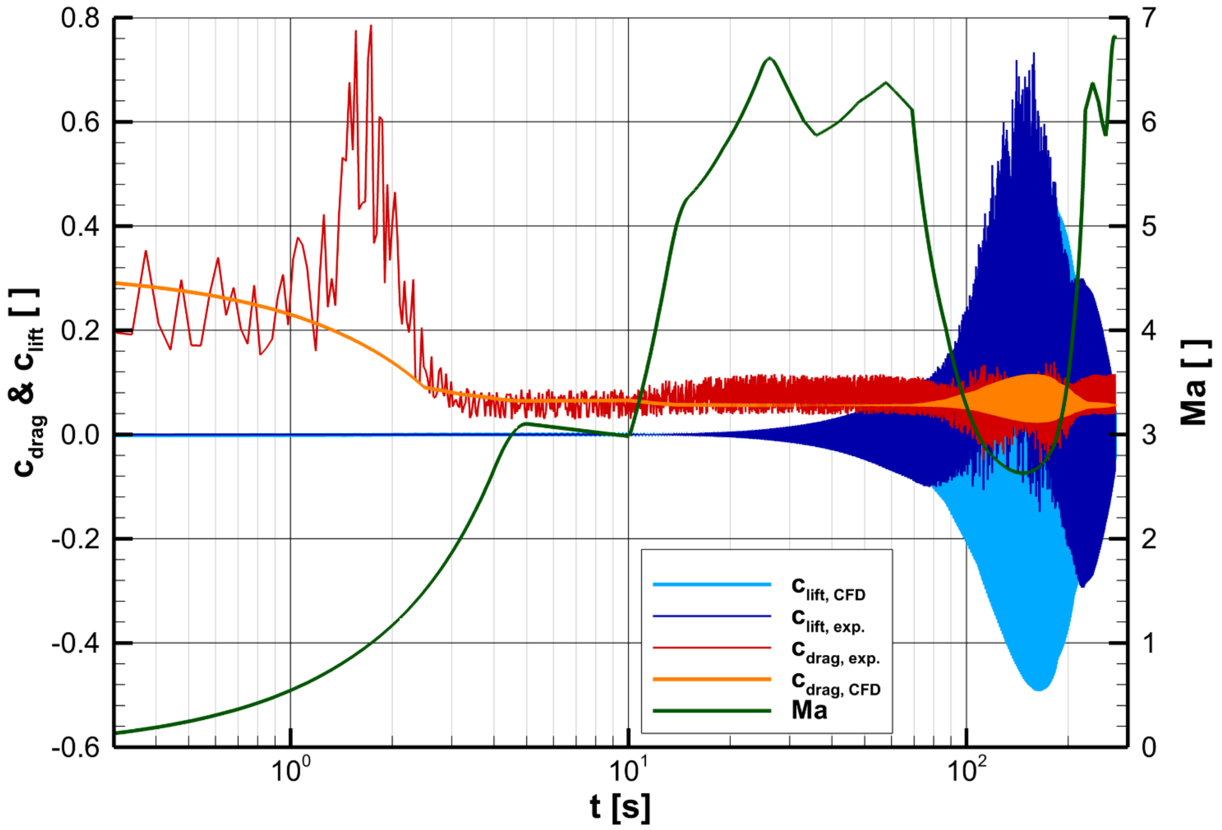


Lift and drag processing chain

Output by PFAT

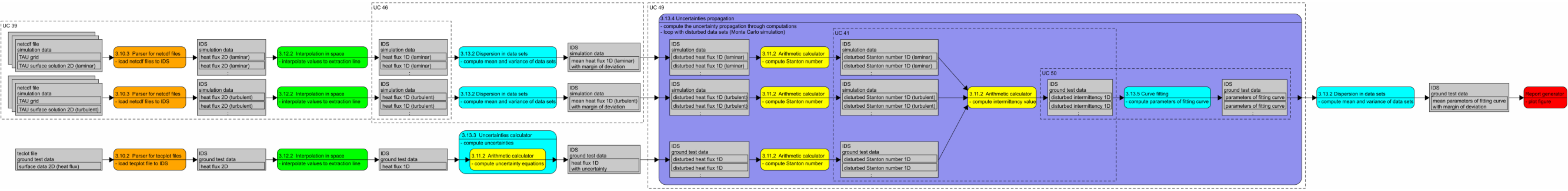


Output by Tecplot and Excel



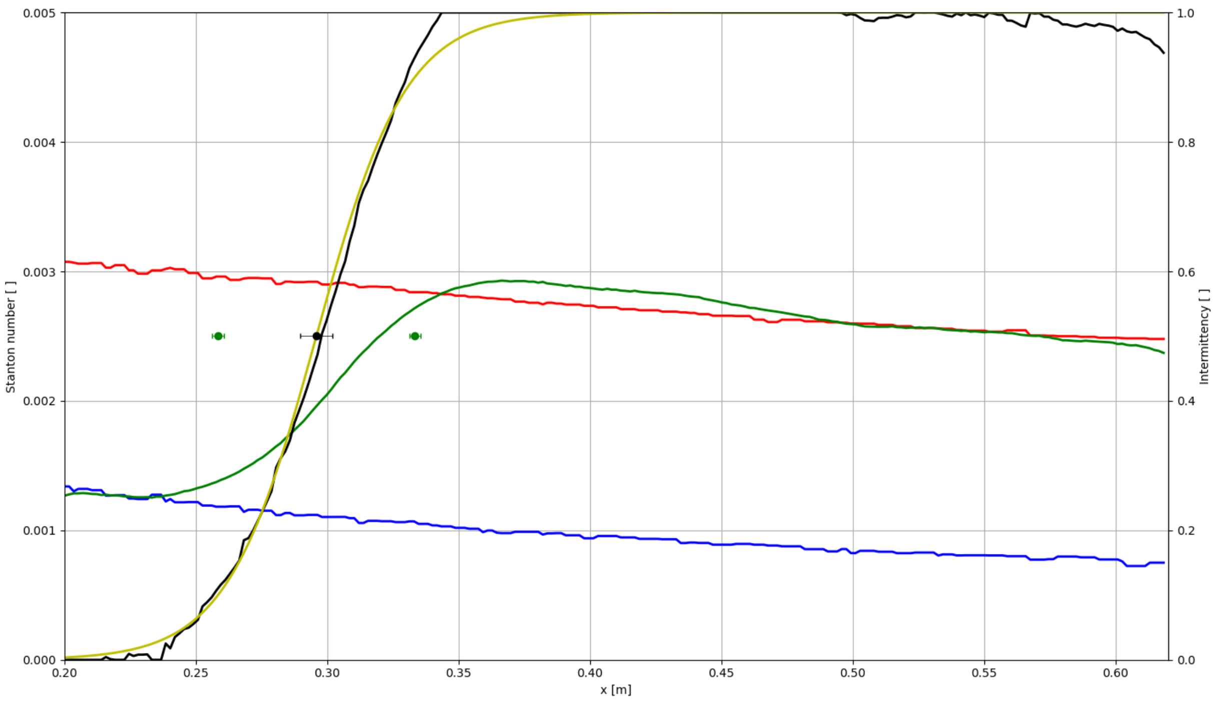
Intermittency processing chain

- Read 3D surface data from netcdf and tecplot files including grid information
- Perform interpolation in space
- Perform arithmetic calculations combining data sets
- Compute uncertainties based on data sets or equations
- Perform curve fitting
- Compute error propagations

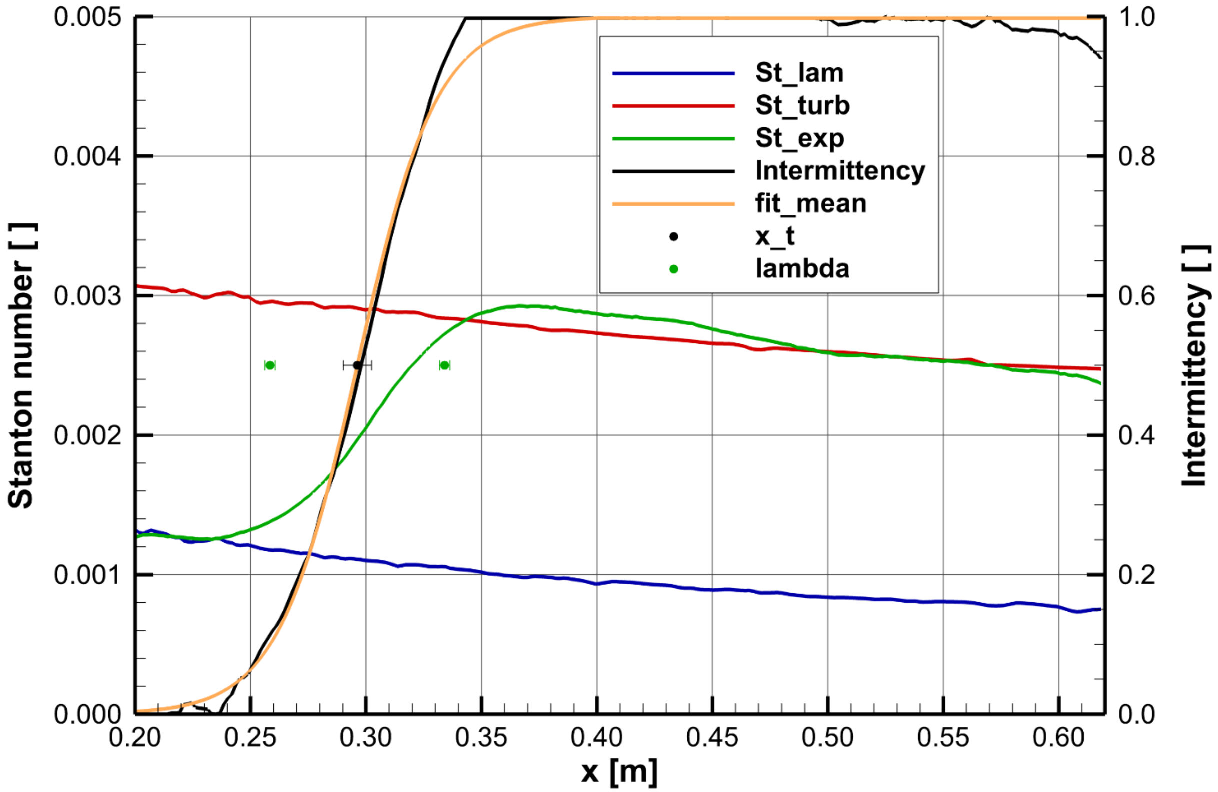


Intermittency processing chain

Output by PFAT



Output by Tecplot and Excel

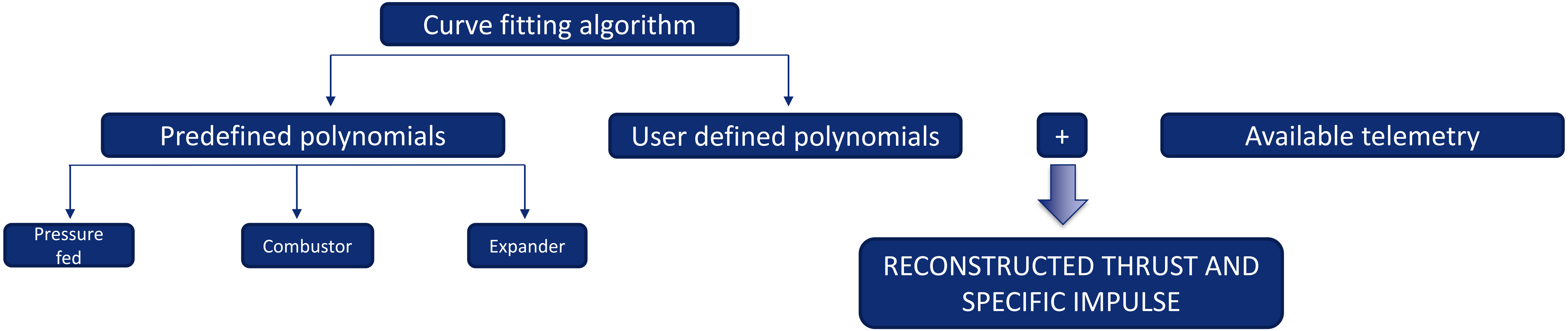


Propulsion Domain

Main goal: Compare in-flight propulsion data measurements with simulation results

e.g.: Chamber pressure, chamber temperature, pumps rotational speed, tanks pressure…

- Engine performance (thrust and specific impulse) are of interest in post-flight analysis, but NOT present in telemetry database
  - *Solution: performance reconstruction module:*



EMPRESARIOS AGRUPADOS

STEPS IN PFAT FOR REACHING THE GOAL

1. Creation of a customized output file from ESPSS models, adjusting the sample frequency and reporting only the parameters of interest.
2. Load the numerical data from ESPSS output file and rename the simulation parameters
3. Pre-process the raw measurements of the propulsion telemetry data
4. Reconstruct the propulsion performance from the processed telemetry data and the ESPSS simulation output
5. Comparison of the reconstructed performance with ESPSS simulation output
6. Plot generation

POST-FLIGHT ANALYSIS CAMPAIGN

Four different tests were included in the validation campaign of the propulsion domain modules.

Remarks:

- All of them have the same structure
- Variability among tests appears in 2 key elements:
  - Deck models
  - Fitting expression used for performance reconstruction

EMPRESARIOS AGRUPADOS

POST-FLIGHT ANALYSIS CAMPAIGN

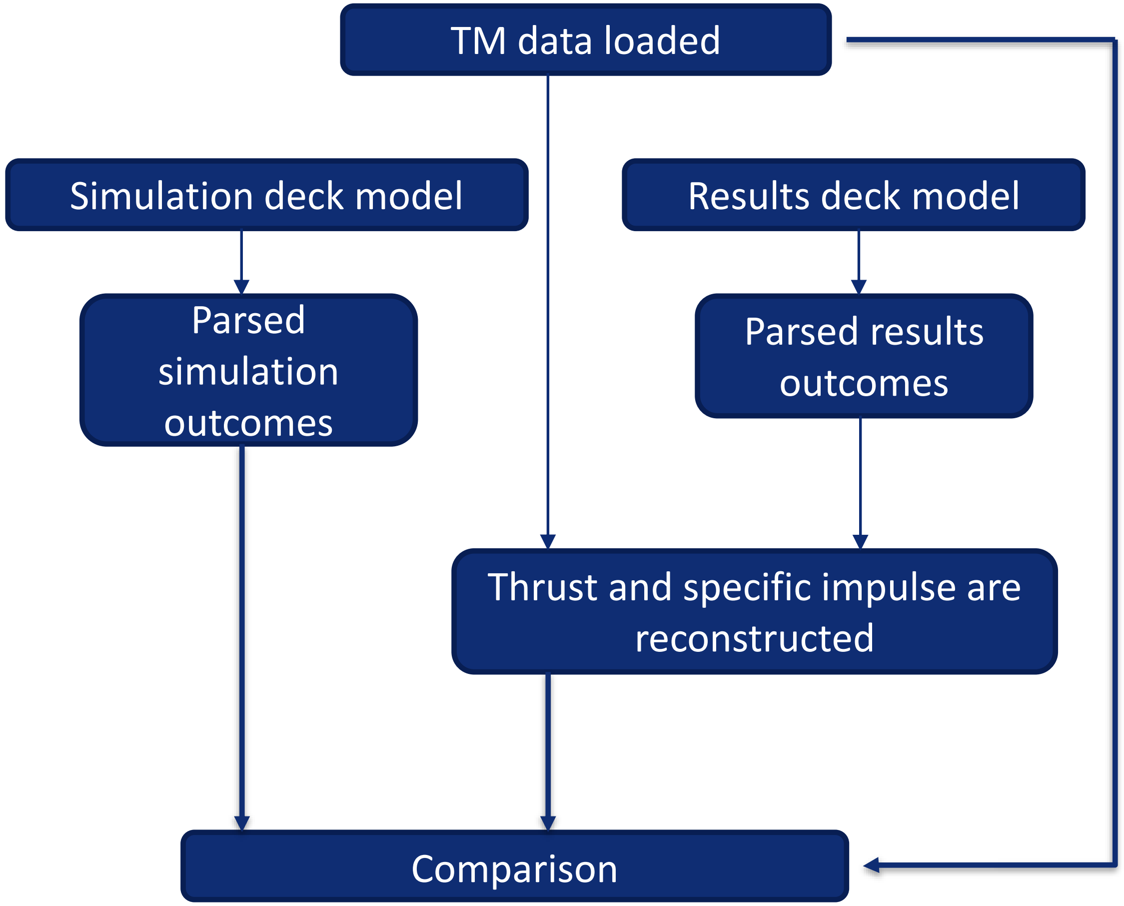
On-board telemetry data available is loaded

Two deck models are executed using the *ESPSS output* module

The outcomes of the deck models are mapped to PFAT standard variable *ESPSS parser* module

The parser output enters into the *performance reconstruction* module to reconstruct the thrust and specific impulse using the telemetry data

The reconstructed thrust and specific impulse are compared with the ones computed by the simulation deck model, generating plots



Propulsion Domain

EMPRESARIOS AGRUPADOS

SINGLE COMBUSTOR TEST

Predefined fitting expression

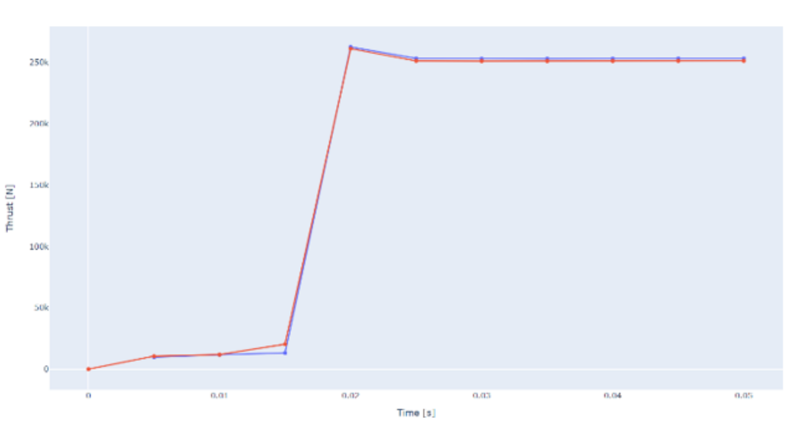
$$Thrust = a_1 P_c + a_2 P_{ext} + a_3$$

$$Isp = b_1 \sqrt{P_c} + b_2 \sqrt{P_{ext}} + b_3$$

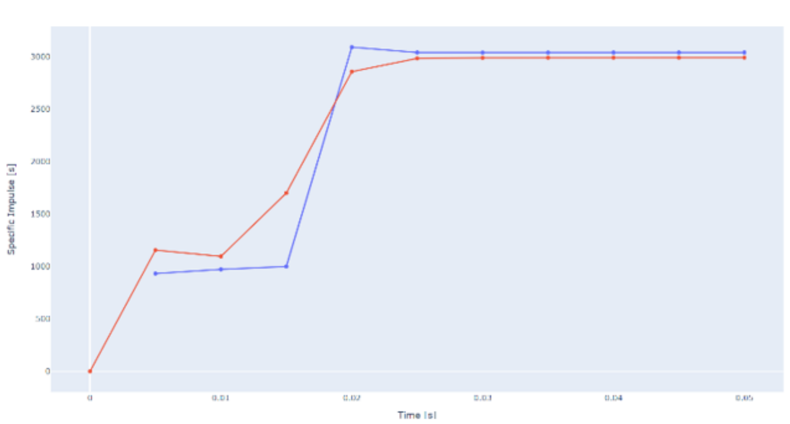
Conclusions

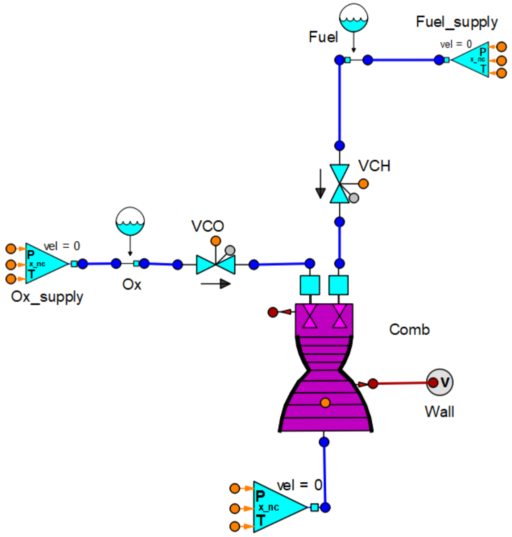
Thrust reconstructed with good accuracy.
More dispersion for impulse reconstruction when starting up.

Thrust



Specific Impulse





* Reconstruction vs. simulation

Propulsion Domain

EMPRESARIOS AGRUPADOS

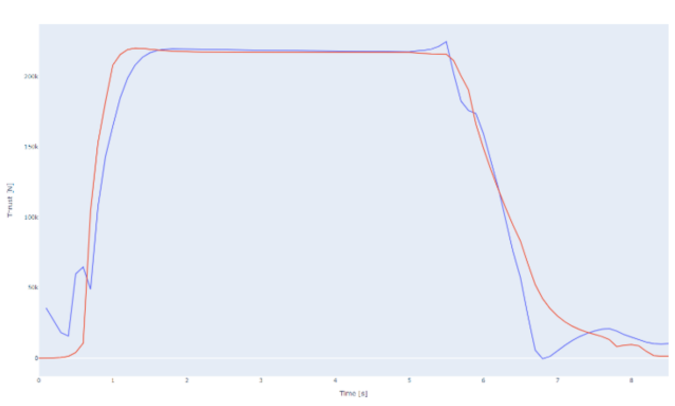
EXPANDER ENGINE TEST

Predefined fitting expression

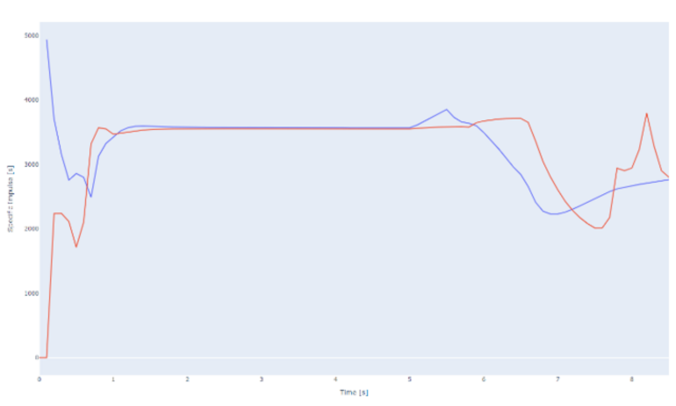
$$Thrust = a_1 P_{oxy}^2 + a_2 P_{red}^2 + a_3 P_{ext}^2 + a_4 N_{oxy}^2 + a_5 N_{red}^2 + a_6 P_{oxy} P_{red} + a_7 P_{oxy} P_{ext} + a_8 N_{oxy} P_{oxy} + a_9 N_{red} P_{oxy} + a_{10} P_{red} P_{ext} + a_{11} N_{oxy} P_{red} + a_{12} N_{red} P_{red} + a_{13} P_{ext} N_{oxy} + a_{14} P_{ext} N_{red} + a_{15} N_{oxy} N_{red} + a_{16} P_{oxy} + a_{17} P_{red} + a_{18} P_{ext} + a_{19} N_{oxy} + a_{20} N_{red} + a_{21}$$

$$Isp = a_1 P_{oxy} + a_2 P_{red} + a_3 P_{ext} + a_4 N_{oxy} + a_5 N_{red} + a_6 \sqrt{P}_{oxy} \sqrt{P}_{red} + a_7 \sqrt{P}_{oxy} \sqrt{P}_{ext} + a_8 \sqrt{N}_{oxy} \sqrt{P}_{oxy} + a_9 \sqrt{N}_{red} \sqrt{P}_{oxy} + a_{10} \sqrt{P}_{red} \sqrt{P}_{ext} + a_{11} \sqrt{N}_{oxy} \sqrt{P}_{red} + a_{12} \sqrt{N}_{red} \sqrt{P}_{red} + a_{13} \sqrt{P}_{ext} \sqrt{N}_{oxy} + a_{14} \sqrt{P}_{ext} \sqrt{N}_{red} + a_{15} \sqrt{N}_{oxy} \sqrt{N}_{red} + a_{16} \sqrt{P}_{oxy} + a_{17} \sqrt{P}_{red} + a_{18} \sqrt{P}_{ext} + a_{19} \sqrt{N}_{oxy} + a_{20} \sqrt{N}_{red} + a_{21}$$

Thrust



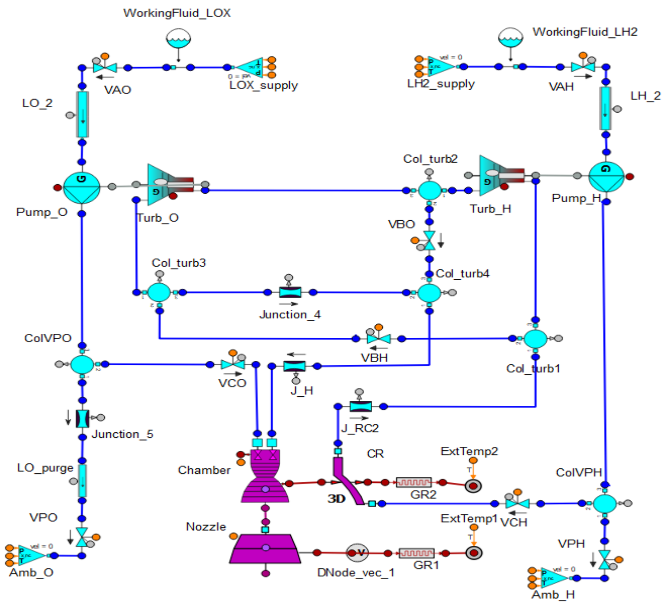
Specific Impulse



* Reconstruction vs. simulation

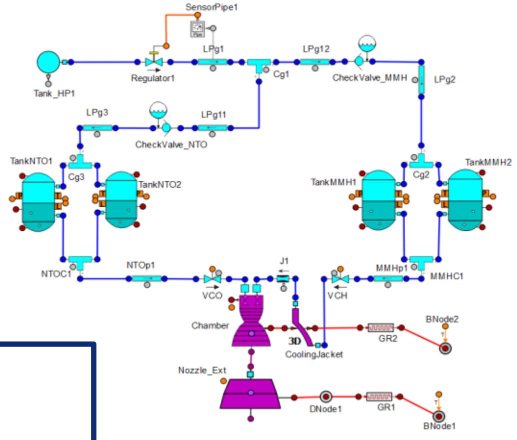
Conclusions

Strong transient regions (startup and shutdown) have more dispersion.
Reconstructed thrust follows the tendency of pumps speed.
Reconstructed curves are not forced to start from 0 or end in 0.



Propulsion Domain

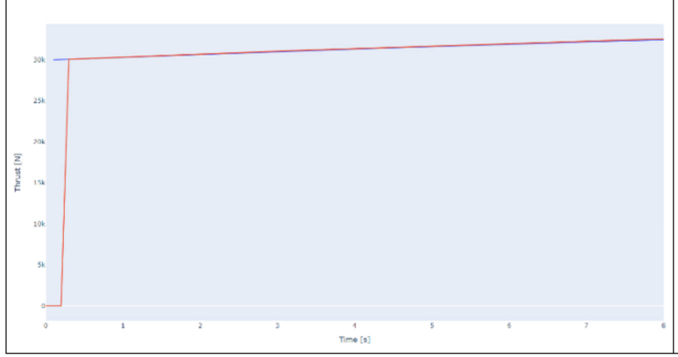
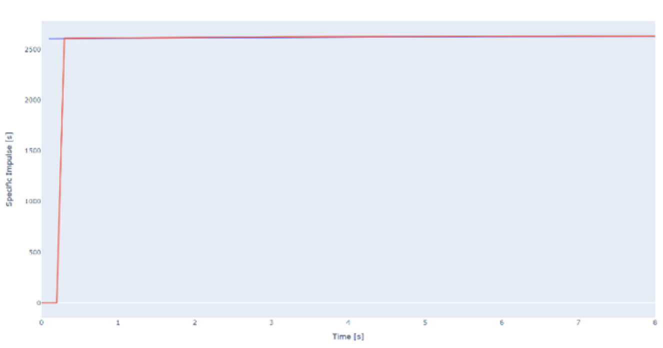
PRESSURE FED ENGINE TESTS



Predefined fitting expression

$$Thrust = a_1P_{oxy}^2 + a_2P_{red}^2 + a_3P_{oxy}P_{red} + a_4P_{oxy} + a_5P_{red} + a_6$$

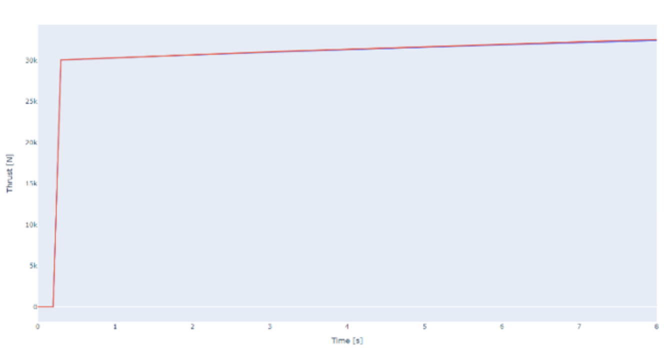
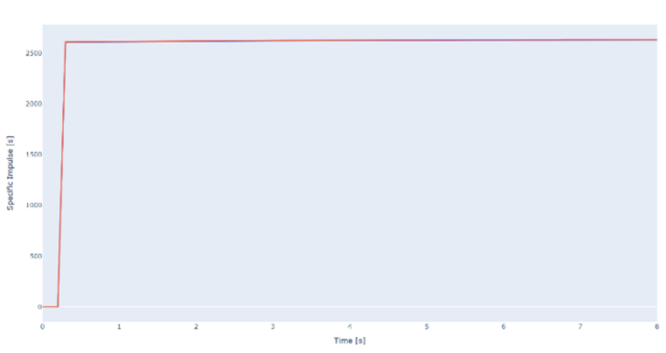
$$Isp = b_1P_{oxy} + b_2P_{red} + b_3\sqrt{P_{oxy}P_{red}} + b_4\sqrt{P_{oxy}} + b_5\sqrt{P_{red}} + b_6$$

Thrust	Specific Impulse
	

User defined fitting expression

$$Thrust = a_1P_{oxy}^2 + a_2P_{red}^2 + a_3P_{oxy}P_{red} + a_4P_{oxy} + a_5P_{red} + a_6 + a_7P_c$$

$$Isp = b_1P_{oxy} + b_2P_{red} + b_3\sqrt{P_{oxy}P_{red}} + b_4\sqrt{P_{oxy}} + b_5\sqrt{P_{red}} + b_6 + b_7\sqrt{P_c}$$

Thrust	Specific Impulse
	

Conclusions

Reconstruction with very good accuracy.
Thrust and impulse near TIME = 0 do not capture the startup.
Very good accuracy in all points of the simulation.
Chamber pressure is able to capture the startup process of the engine.

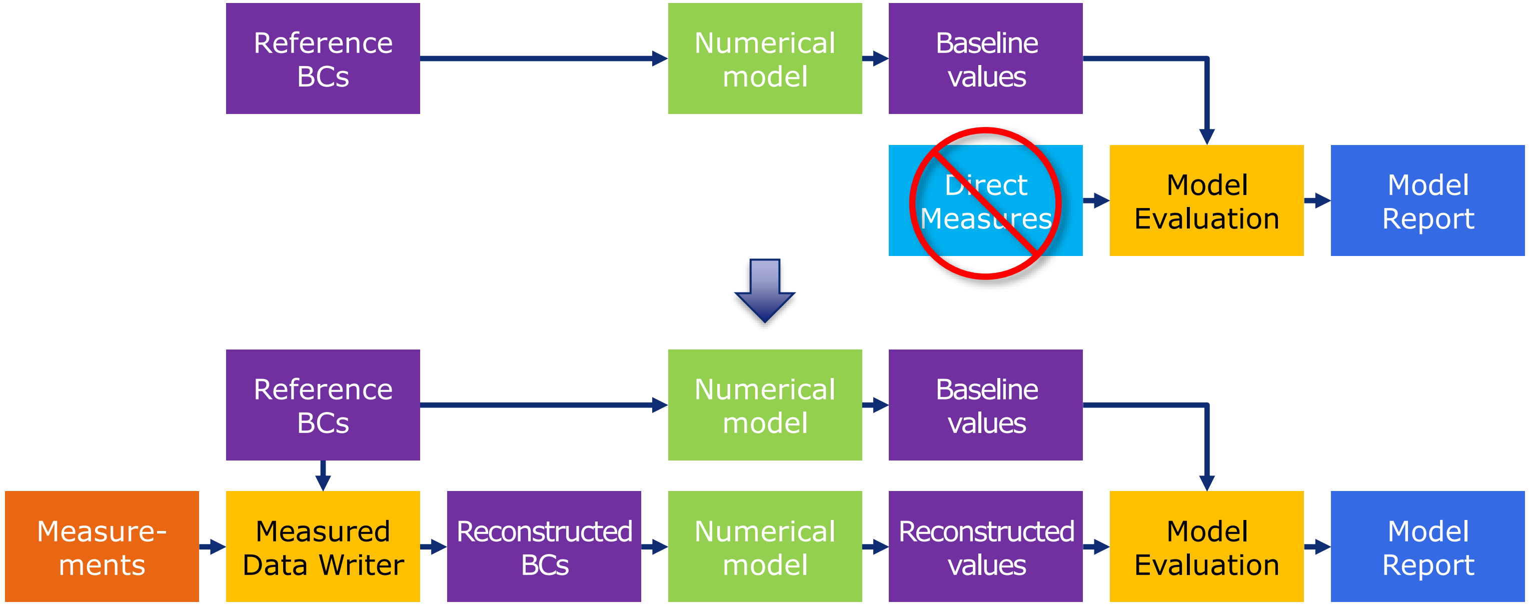
* Reconstruction vs. simulation

Structural, Separation, Acoustic Domains

Together ahead. RUAG

- **Goal:** evaluate the behavior in the mechanical and acoustic domains
- **Constraint:** no direct measurements (material strains / sound pressure in the payload bay)

➔ **Approach:** use other sensors/reconstructed data as boundary conditions; simulate new behavior, and compare to baseline

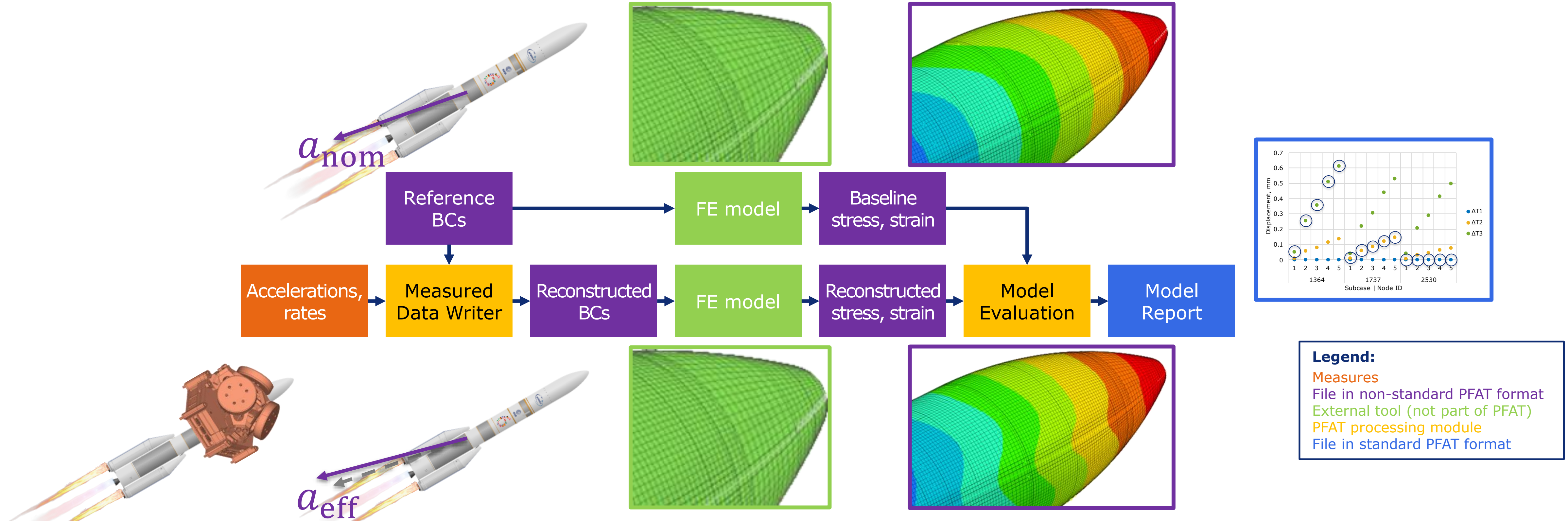


Legend:
Measures
File in non-standard PFAT format
External tool (not part of PFAT)
PFAT processing module
File in standard PFAT format

Structural domain (static): goals and implementation

Together ahead. RUAG

- **“Direct” values of interest:** structural displacements, stresses, strains
- **Available measurements:** launch vehicle accelerations and rates
- **Reconstructed Boundary Conditions:** accelerations, rates

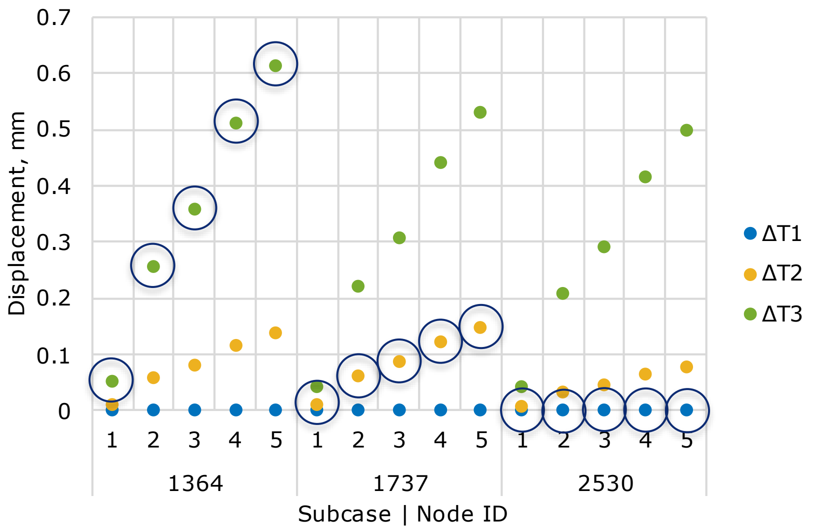


Structural domain (static): verification and results

Together ahead. RUAG

- **Measured data:** generated synthetically

➔ **Evaluate:** correct creation of model based on measured data ✔
correct post-processing of model results ✔

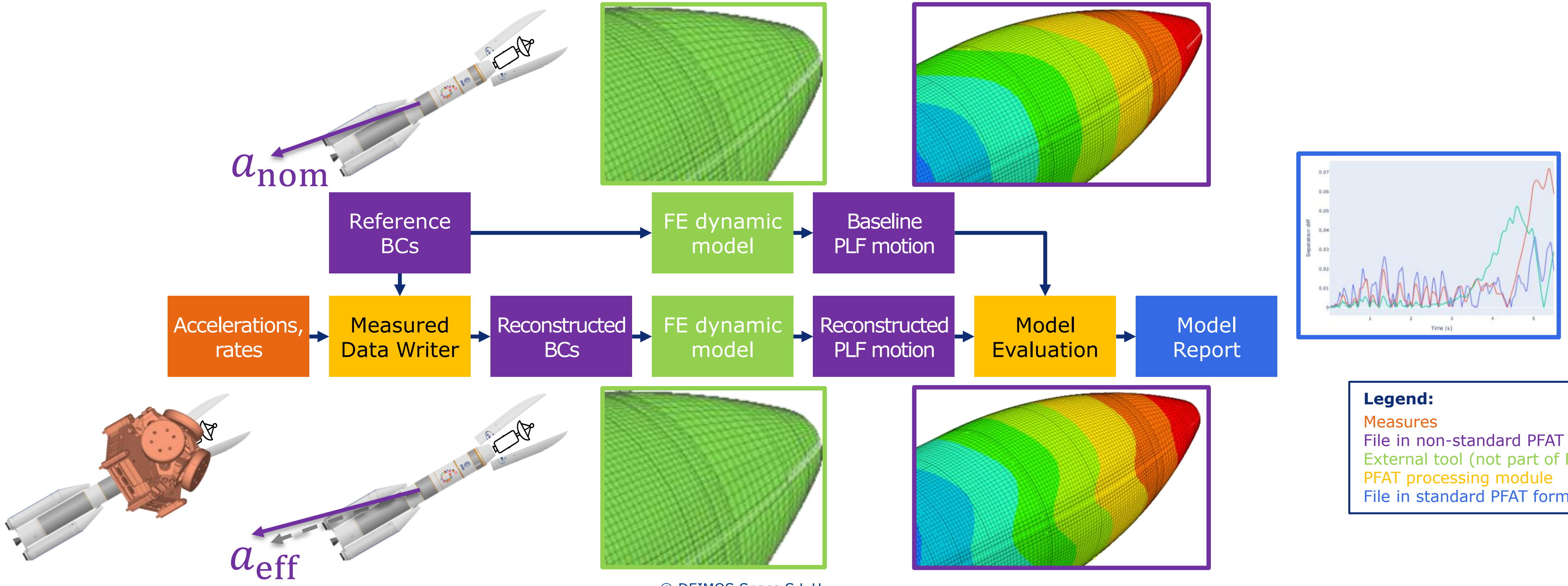


Displacements									
Subc.	Average difference, m			Maximum difference, m			Node ID with max diff.		
	T1	T2	T3	T1	T2	T3	T1	T2	T3
1	7.247E-08	1.013E-05	4.567E-05	1.663E-07	1.231E-05	5.112E-05	2530	1737	1364
2	3.624E-07	5.065E-05	2.284E-04	8.317E-07	6.155E-05	2.556E-04	2530	1737	1364
3	5.073E-07	7.091E-05	3.197E-04	1.164E-06	8.616E-05	3.578E-04	2530	1737	1364
4	7.247E-07	1.013E-04	4.567E-04	1.663E-06	1.231E-04	5.112E-04	2530	1737	1364
5	8.697E-07	1.216E-04	5.481E-04	1.996E-06	1.477E-04	6.134E-04	2530	1737	1364

Structural domain (separation): goals and implementation

Together ahead. RUAG

- **"Direct" values of interest:** separation kinematics (nodal displacements)
- **Available measurements:** launch vehicle accelerations and rates
- **Reconstructed Boundary Conditions:** accelerations and rates at separation start



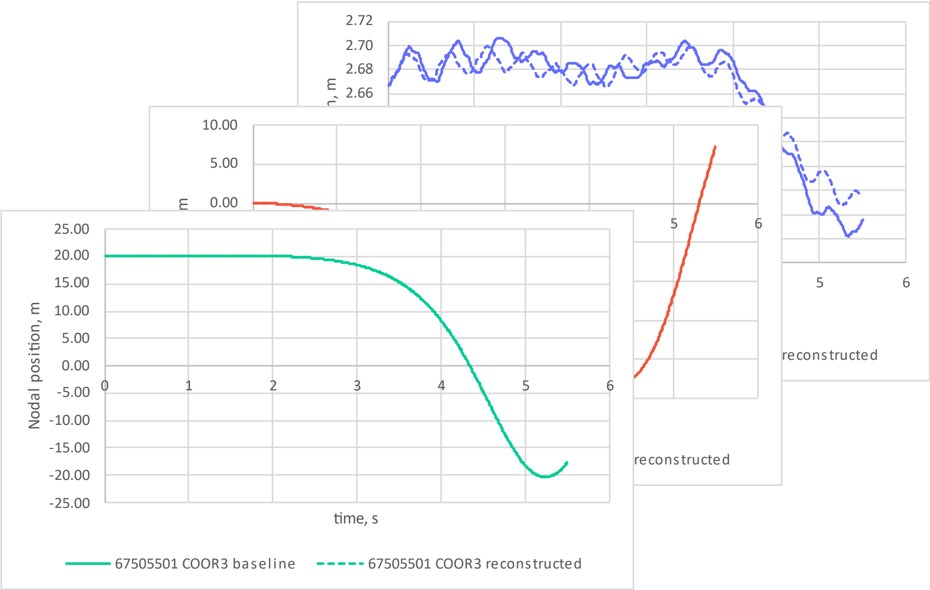
Legend:
Measures
File in non-standard PFAT format
External tool (not part of PFAT)
PFAT processing module
File in standard PFAT format

Structural domain (separation): verification and results

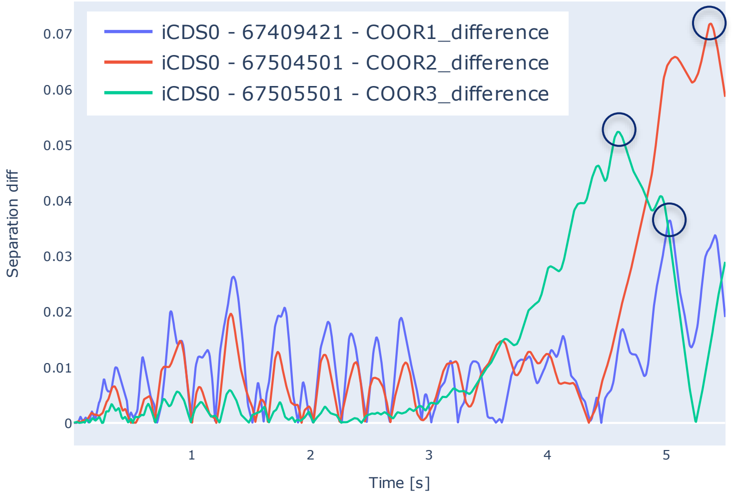
Together ahead. RUAG

- **Measured data:** generated synthetically

➔ **Evaluate:** correct creation of model based on measured data ✔
correct post-processing of model results ✔

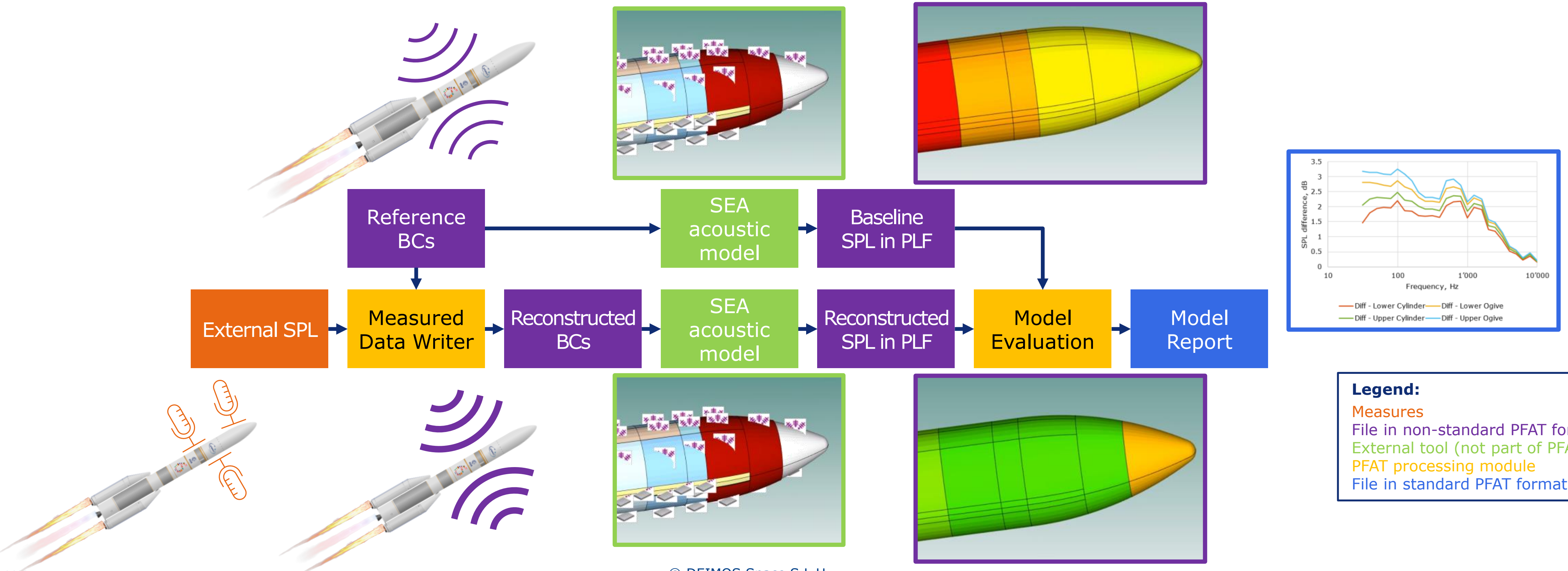


Maximum difference, m			Node with max. difference			Time of max. difference, s		
COOR1	COOR2	COOR3	COOR1	COOR2	COOR3	COOR1	COOR2	COOR3
3.64734E-2	7.18921E-2	5.24135E-2	67409421	67504501	67505501	5.029224	5.374133	4.599224



Acoustic domain: goals and implementation

- **"Direct" values of interest:** acoustics (sound pressure levels) within the PLF
- **Available measurements:** SPL at discrete locations outside the launch vehicle
- **Reconstructed Boundary Conditions:** SPL outside the PLF



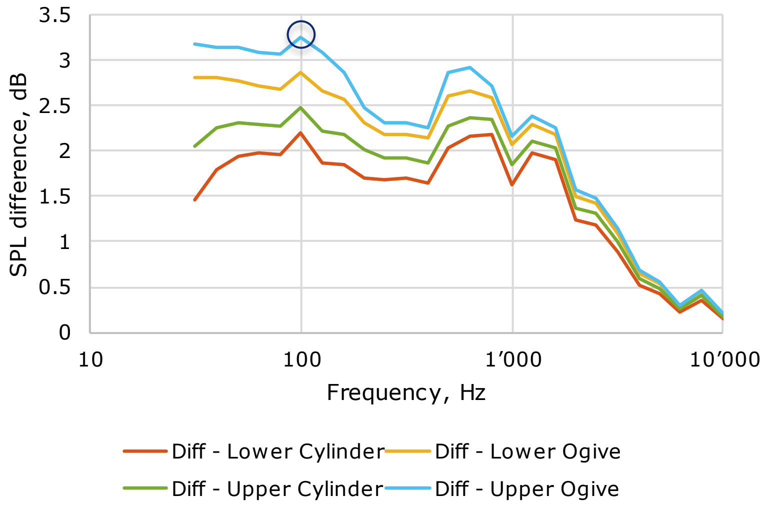
Legend:
Measures
File in non-standard PFAT format
External tool (not part of PFAT)
PFAT processing module
File in standard PFAT format

Acoustic domain: verification and results

Together ahead. RUAG

- **Measured data:** generated synthetically

➔ **Evaluate:** correct creation of model based on measured data ✔
correct post-processing of model results ✔

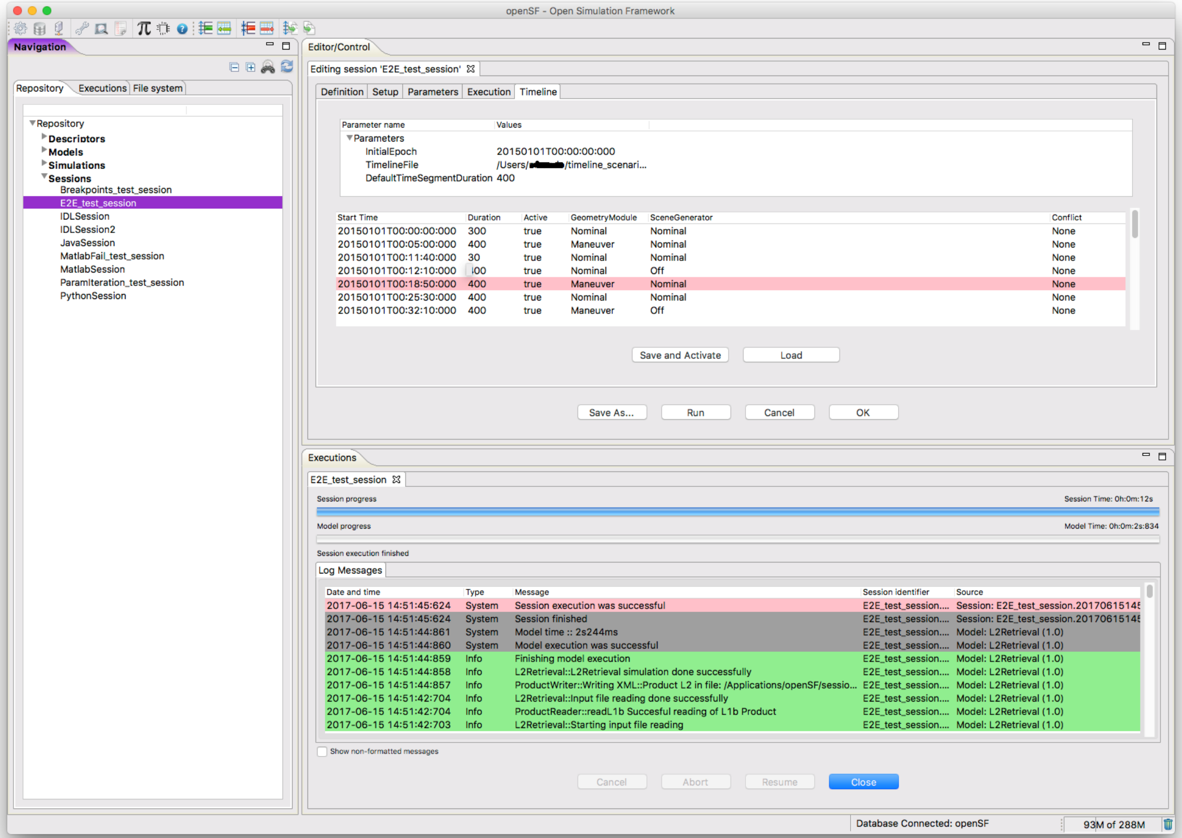


Maximum SPL difference, dB	Interior Volume with maximum SPL difference	Frequency with maximum SPL difference, Hz
3.26228e+00	Upper_ogive	100.0

PFAT Graphical User Interface

Based on ESA openSF

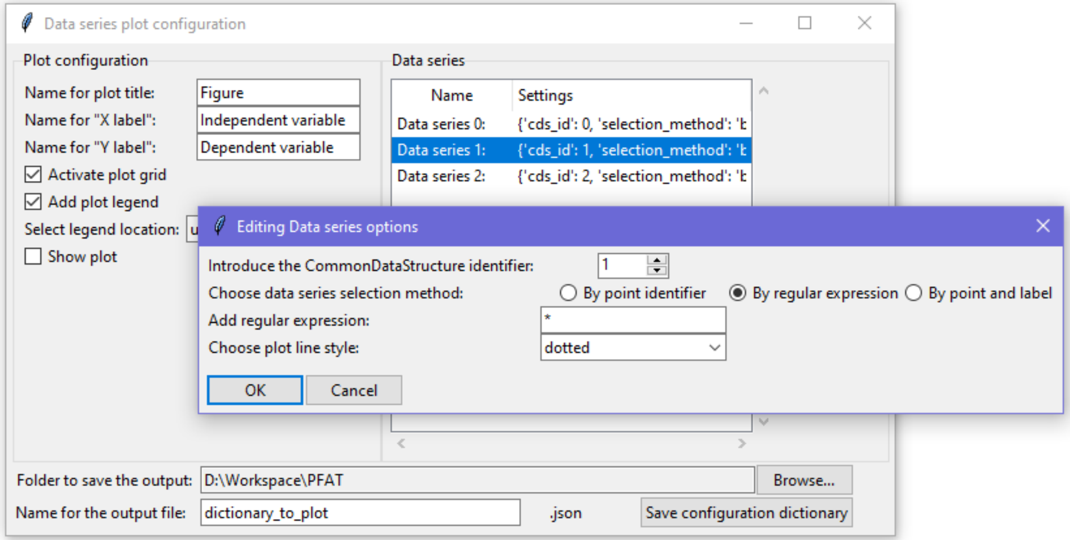
- All the openSF functionalities are made available

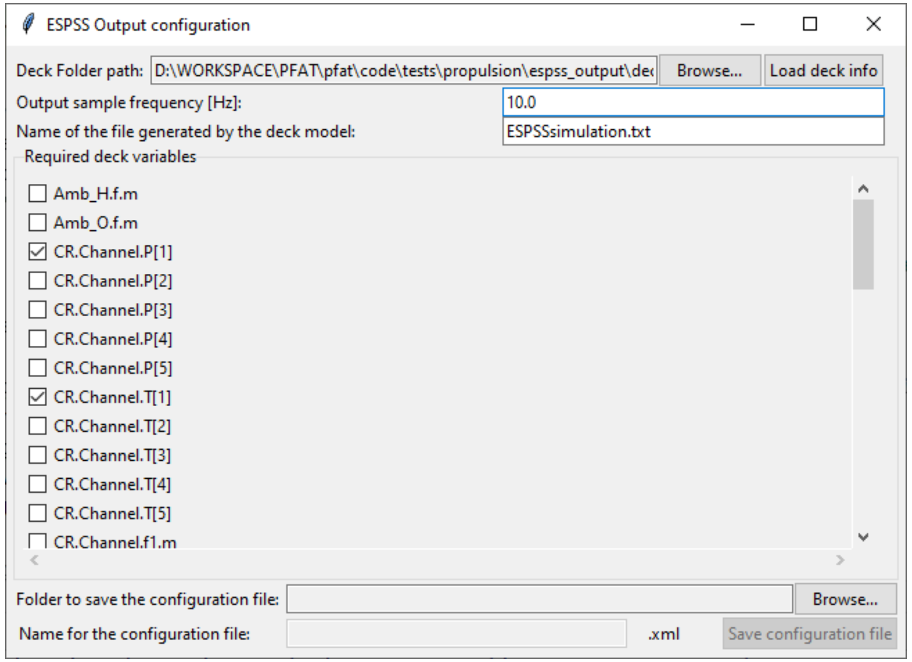


Dedicated Extensions

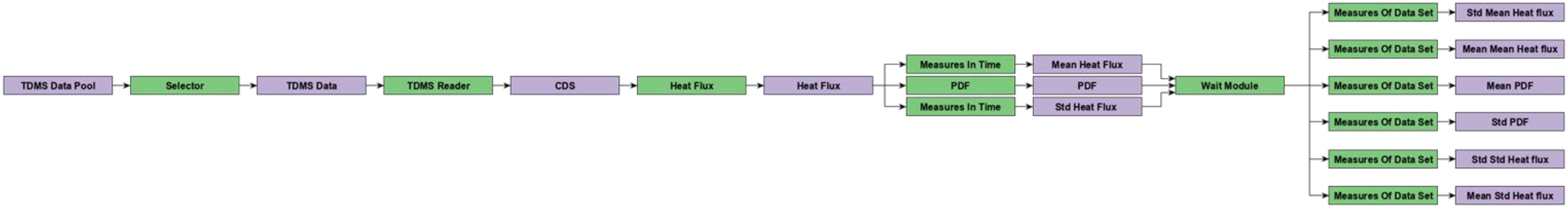
Data series plot configuration

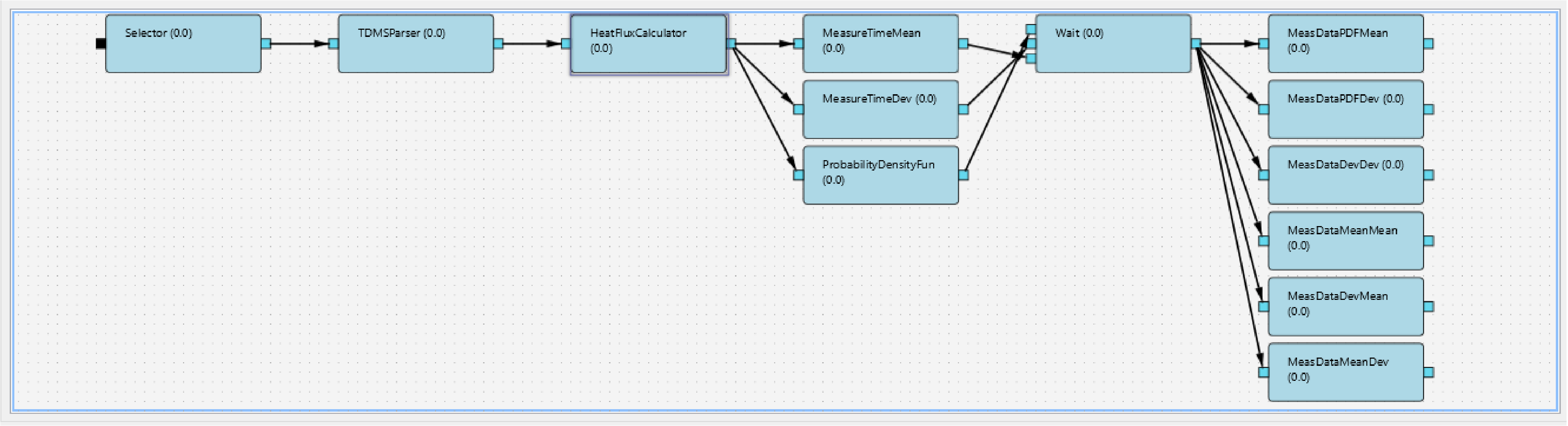
Deck model parameters and ESPSS Output configuration



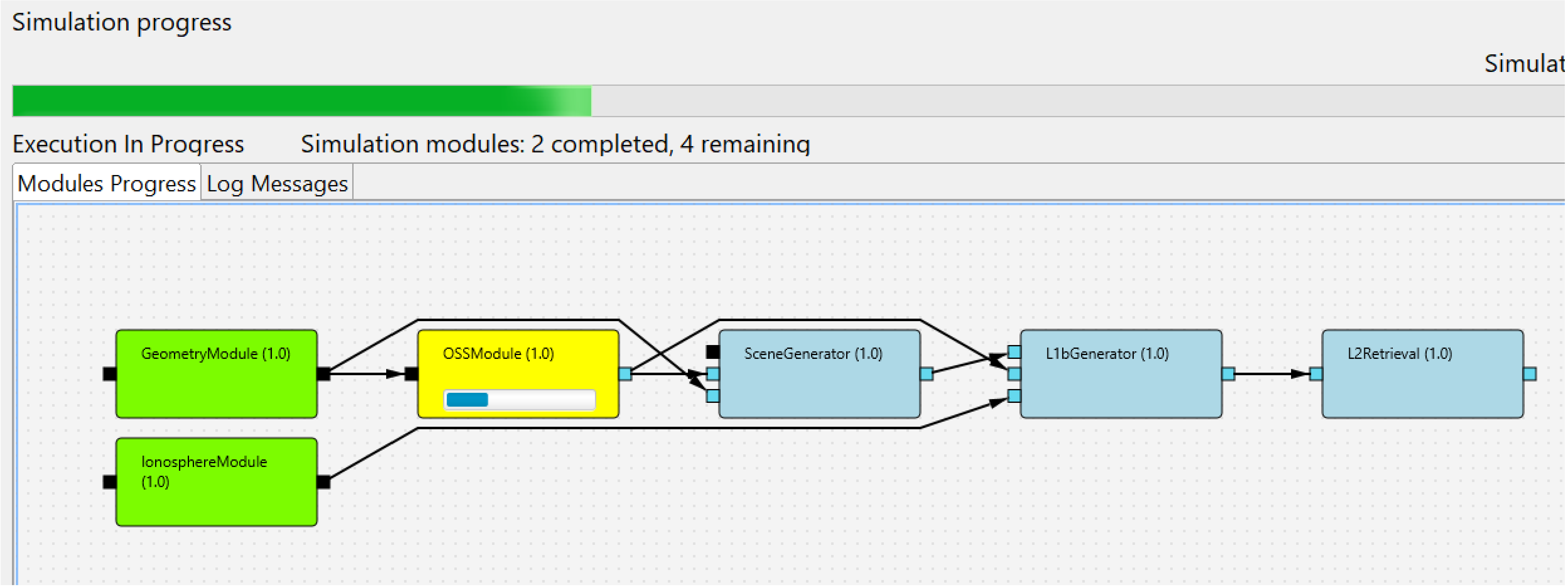


Graphical Processing Chains





Graphical Processing Chains



PFAT Command Line Interface

PFAT has been designed to be used via Command Line Interface, to exploit its capabilities

```python
def measured_launch_vehicle_acceleration_reader(file_path: str, data_labels: List[str]) -> CommonDataStructure:
    """ Function that reads .txt files and assigns values to the data labels in the launch vehicle acceleration type of
    file

    Function expects a an input file with data that can be converted into float type and is arranged in columns. It does
    not check the file, and in case any value is given incorrectly, the data will raise problems when introduced in
    numpy arrays, in other functions. The pandas read function does support "NaN" inputs and these inputs will be
    treated as such by numpy.

    Args:
        file_path: path to the file that will be read
        data_labels: list of data labels that will be used to store the data in the common data structure.
                     They are given in the same order as the columns of the file

    Returns:
        Out: Common data structure with the data loaded and assigned to the default point
    """
    # Only checks inputs are floats. Does not filter NaNs.
    data = pd.read_csv(file_path, sep="\t", dtype=float).T
    out = CommonDataStructure.from_array(np.array(data.iloc[0, :]).reshape(1, len(data.iloc[0, :])),
                                         data_labels[0], data_labels[1:], np.array(data.iloc[1:, :]))
    return out
```

4 Validation Campaign

Validation Approach

Comprehensive traceability from use cases down to unit testing

34 Use Cases

↕

43 Functional Requirements (37 Testing + 6 Inspection)

↕

34 Test Cases Specifications

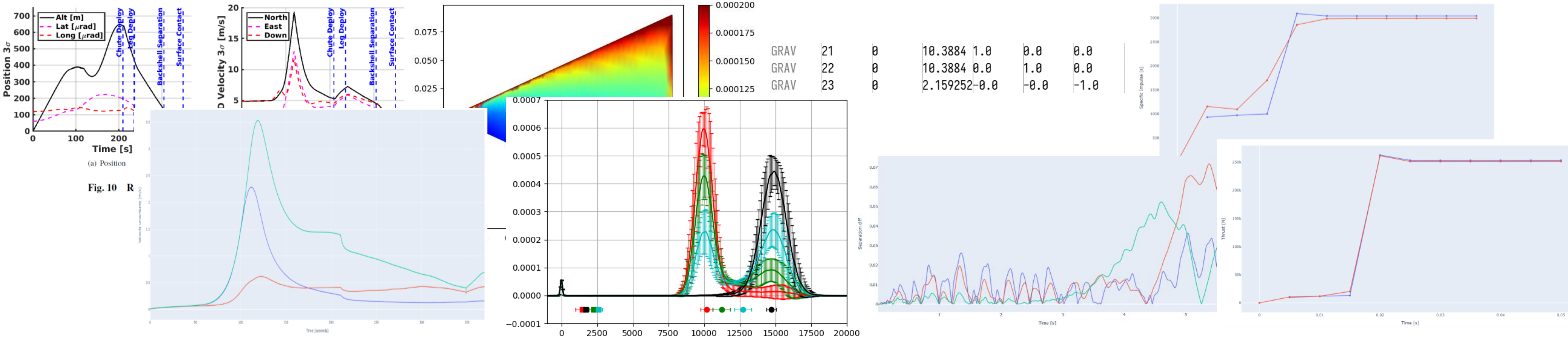
↕

944 Unit Tests + 17 Acceptance Tests Specifications

Validation Approach

17 Acceptance Test Cases for 4 Engineering Domains

- To guarantee that the toolkit is able to perform End-to-End post-flight analysis
- Acceptance Tests reproduce post-flight analysis against real flight data (whenever available) or against synthetic data mimicking real flight data



5 Achievements and Future Works

Achievements

During the PFAT activity, the following specific objectives have been fully achieved:

- A flexible and powerful Post-Flight Analysis Tool has been developed, having in mind its use in post-flight analysis of ESA missions, i.e. the interoperability with other engineering software tools commonly used by ESA (e.g. TAU, NASTRAN, ASTOS, ESPSS);
- Robust and generic post-flight algorithms and analysis tools have been implemented, supporting the following engineering domains: propulsion, aerothermodynamics, structures, acoustic and trajectories;
- An exchange format has been designed and implemented: the Common Data Structure;
- A powerful Graphical User Interface is provided along with a flexible Command Line Interface, providing a dual approach to PFAT use;
- Automatic generation capabilities of post-flight analysis reports have been implemented, with user friendly mechanisms to set them up;
- The toolkit capabilities have been validated against reference real post-flight data, when possible. When flight data were not available, realistic synthetic data have been used

PFAT has been an **ambitious** project, whose objectives have been **fully reached**.

Its use with real data could quantify the objectives achieved and help in identifying the gaps to be filled.

The future work should be oriented in two main directions:

- Reinforcing the capabilities already present (e.g. adding new filters, mathematical functionalities, sophisticated methodologies), and
- Extending them towards other engineering domains and/or data formats (e.g. providing access to further sensors/flight data)

Thank you

www.elecnor-deimos.com