ESTECO
INTERNATIONAL
USERS' MEETING

um
2026

Workflow standardization and modeFRONTIER customization

Rémy Bompoint
Optimization Technical Leader

STELLANTIS

Agenda

ESTECO
INTERNATIONAL
USERS' MEETING

SDO/MDO universal workflow

- Key challenges
- Workflow architecture
- Industrial usage

Design Space improvement with pyConsole

- Automated reporting
- RSM user interface

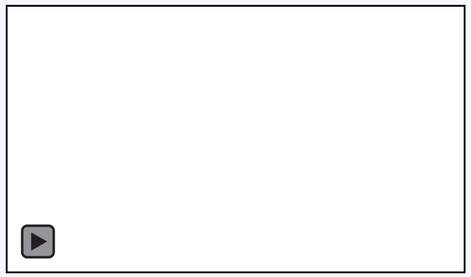
SDO/MDO universal workflow

Purpose of the initiative

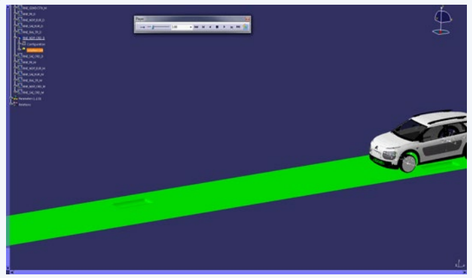
1 modeFrontier optimization workflow for any Stellantis simulation tools.

- ➔ Easily adaptable to any solvers
- ➔ Relatively easy to use

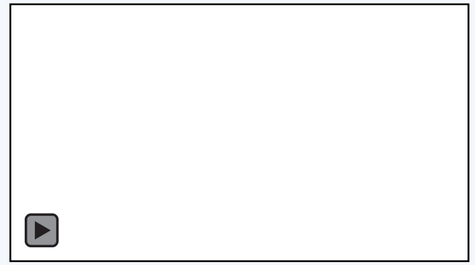
3D mechanics



3D Fluid



Dynamics



01/1D complex systems

TECHNICAL CONSTRAINTS

Diversity of optimization problems

- MDO (safety / NVH / DSM) – 2 to 20 loadcases
- SDO (aerodynamic, 0D/1D, safety)

→ Minimize actions to update modeFrontier
Standardize loadcase management

Data storage

- Models size 100Mo → 2Go
- Results size 100Mo → 20Go

→ Minimize storage on user computer

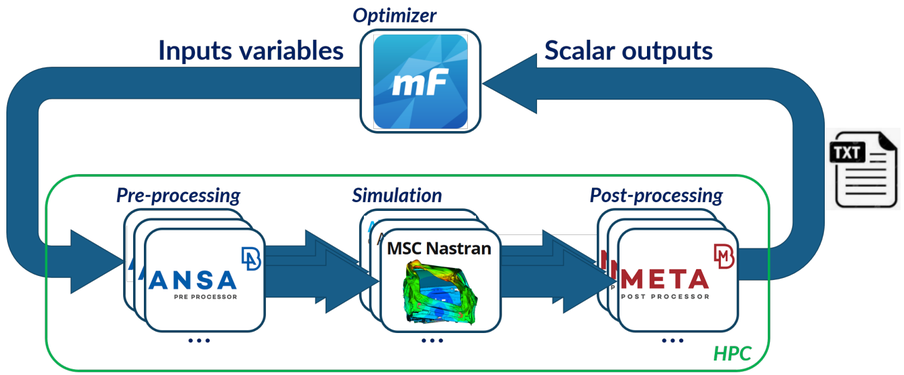
Solution

The brain

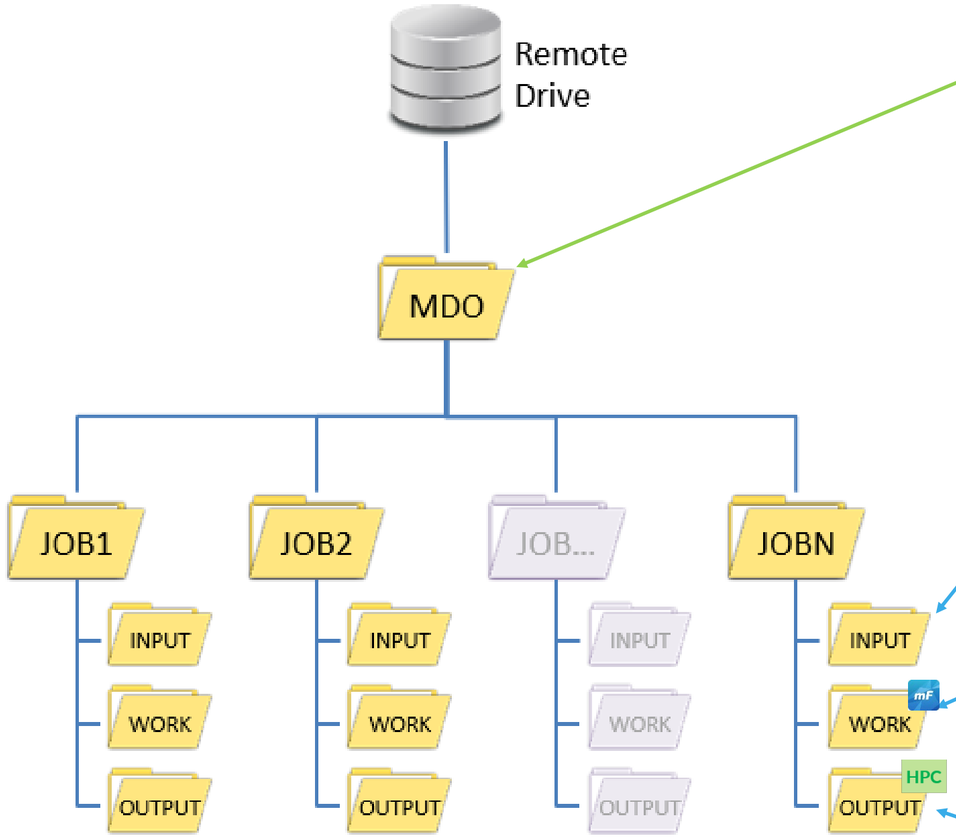
- ModeFrontier Workflow based on Python
- Folder structure and output file standardization (<1Mo)

The muscles

- Workflows on HPC (pre / solver / post chaining)



Folder structure standardization



Architecture example: MDO with N simulation jobs

*.csv file with **folders to use**

Invariant files
- *.csv correspondance file
- parametrized Ansa file
- HPC mandatory files (includes, postprocessing script...)
- HPC job submission template

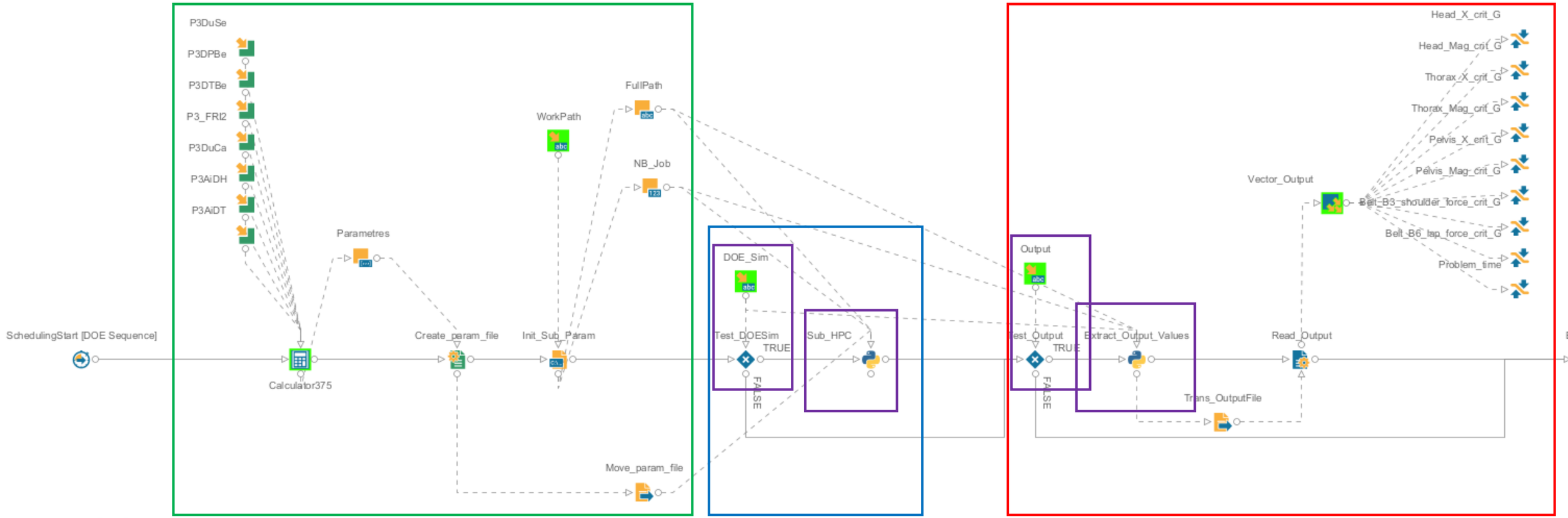
Design files / 1 folder for each Design
- parameters values
- HPC submission file

HPC results files / 1 folder for each Design
- standardized result file
- all other HPC files needed

ModeFrontier workflow

A flexible and generic framework

- ➔ Adapt to any number of loadcases thanks to a structured folder setup
- ➔ Only Inputs & Outputs/Constraints need to be adjusted for each study



Initialization

HPC submission

WORK folders creation

Outputs loading

OUTPUT folders reading

ModeFrontier workflow

A flexible and generic framework

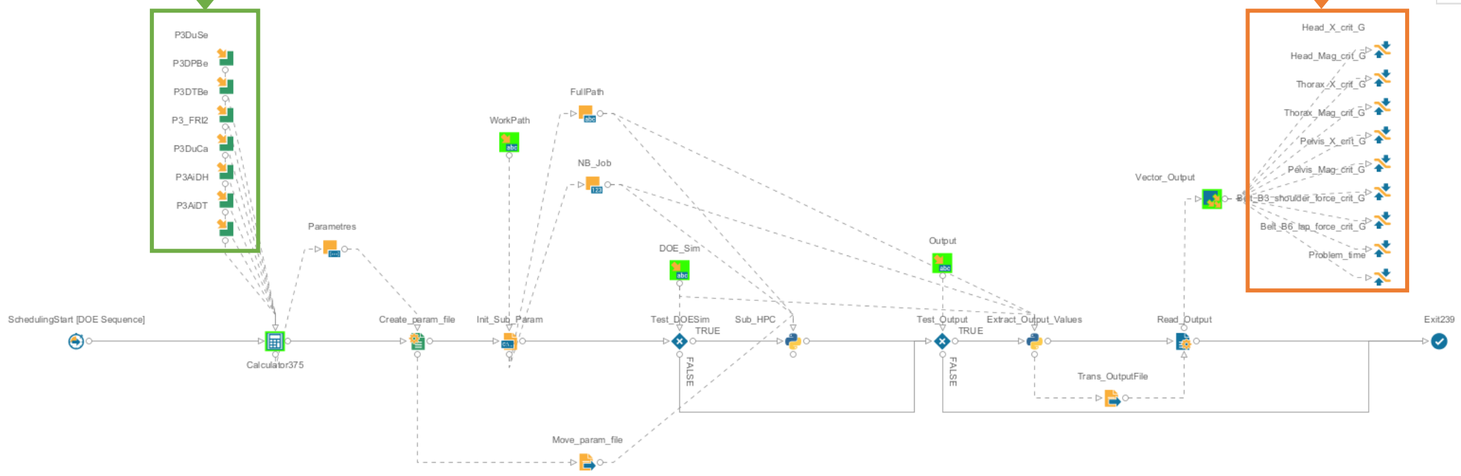
- ➔ Adaptable to any number of load cases through a structured folder setup
- ➔ Only Inputs and Outputs/Constraints need to be adjusted for each study
- ➔ Fast updates via Excel copy-paste
  - Direct definition of optimization studies with clients using Excel sheets
  - Essential for achieving short leadtime (typically 5–50 parameters & 5–500 constraints)

Parameters definition

Name	Default value	Lower bound	Upper bound	Step
P3DuSe	1	0.1	3	0.05
P3DPBe	1	0.1	3	0.05
P3DTBe	1	0.1	3	0.05
P3_FRI2	1	0.1	3	0.05
P3DuCa	1	0.1	3	0.05
P3AiDH	1	0.1	3	0.05
P3AiDT	1	0.1	3	0.05

Constraints definition

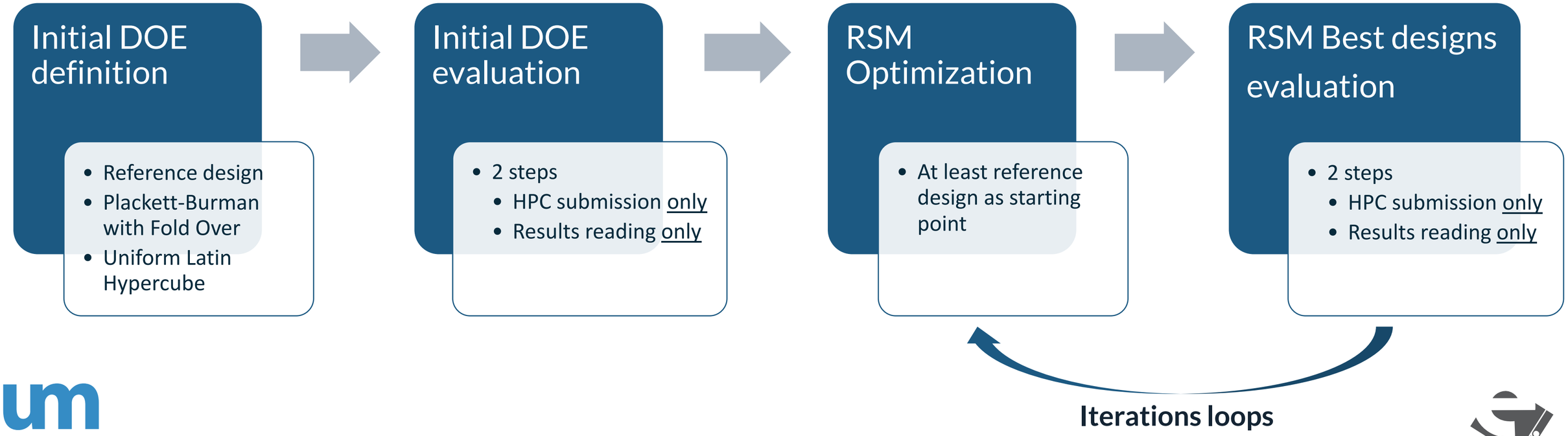
Name	Expression	Type	Limit
Head_X_crit_G	Vector_Output[0]	Greater than	90
Head_Mag_crit_G	Vector_Output[1]	Greater than	90
Thorax_X_crit_G	Vector_Output[2]	Greater than	90
Thorax_Mag_crit_G	Vector_Output[3]	Greater than	90
Pelvis_X_crit_G	Vector_Output[4]	Greater than	90
Pelvis_Mag_crit_G	Vector_Output[5]	Greater than	90
Belt_B3_shoulder_force_crit_G	Vector_Output[6]	Greater than	90
Belt_B6_lap_force_crit_G	Vector_Output[7]	Greater than	90
Problem_time	Vector_Output[8]	Greater than	129



Optimization Strategy

No use of modeFrontier optimization algorithms in autonomous mode from scratch

- ❑ Loadcases with high computation time (crash simulations ~8/36h)
- ❑ Very efficient HPC (possibility to have >100 simulations evaluated simultaneously for some solvers)
- ❑ ModeFrontier concurrency limited (by Windows stations capabilities + algorithms sometimes)



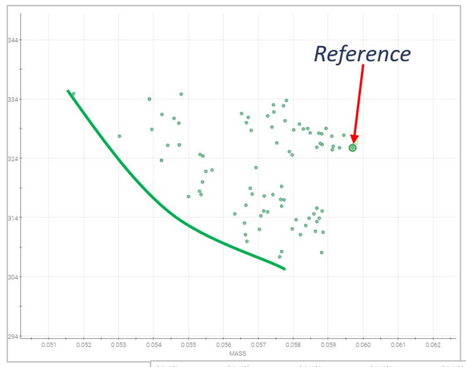
6 years feedback

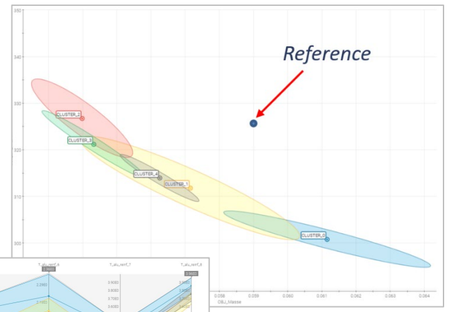
Robust workflow

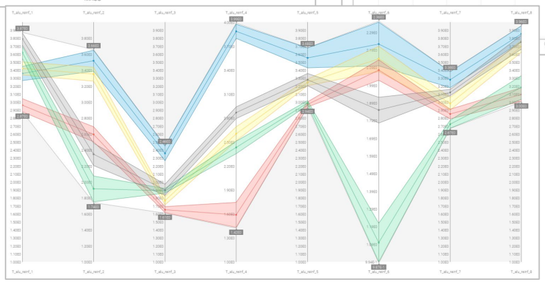
- ❑ No update needed even with solver & HPC evolutions
- ❑ People trained to 1 workflow to face all simulations optimizations

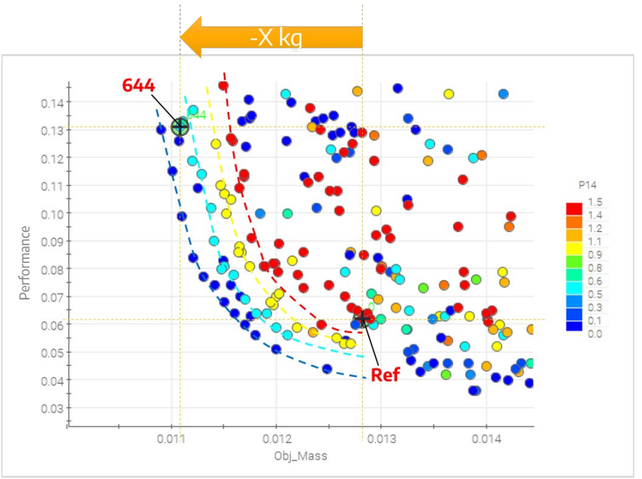
Allow to provide various insight to projects

- ❑ 2 types of studies: decrease mass or improve performance
- ❑ Parameters/Outputs knowledge: correlation matrix, sensitivity, scatter plot (pareto front), parallel coordinates, clustering
- ❑ Decision support: evaluate trade-offs









Focus on feasible designs

Design Space improvement with pyConsole

Purpose of the initiative

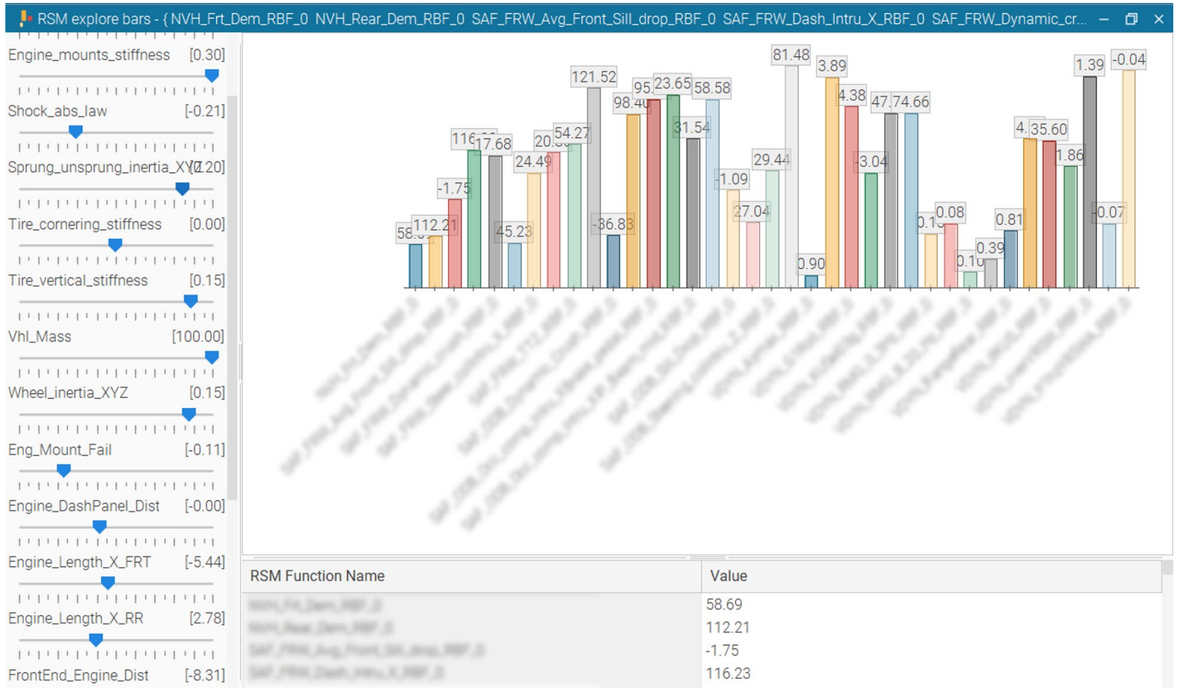
Limitations of the Design Space tool for some of our use case

- ❑ Best suited for advanced users, especially when dealing with large datasets
- ❑ Several charts needed to analyze specific outputs, not so easy to navigate

➔ Automate analysis report

- ❑ RSM explore bars useful but not enough customizable

➔ Create dedicated GUI



PYCONSOLE BENEFITS

Purpose of the initiative

Limitations of the Design Space tool for some of our use case

- ❑ Best suited for advanced users, especially when dealing with large datasets
- ❑ Several charts needed to analyze specific outputs, not so easy to navigate

→ Automate analysis report

- ❑ RSM explore bars useful but not enough customizable

→ Create dedicated GUI

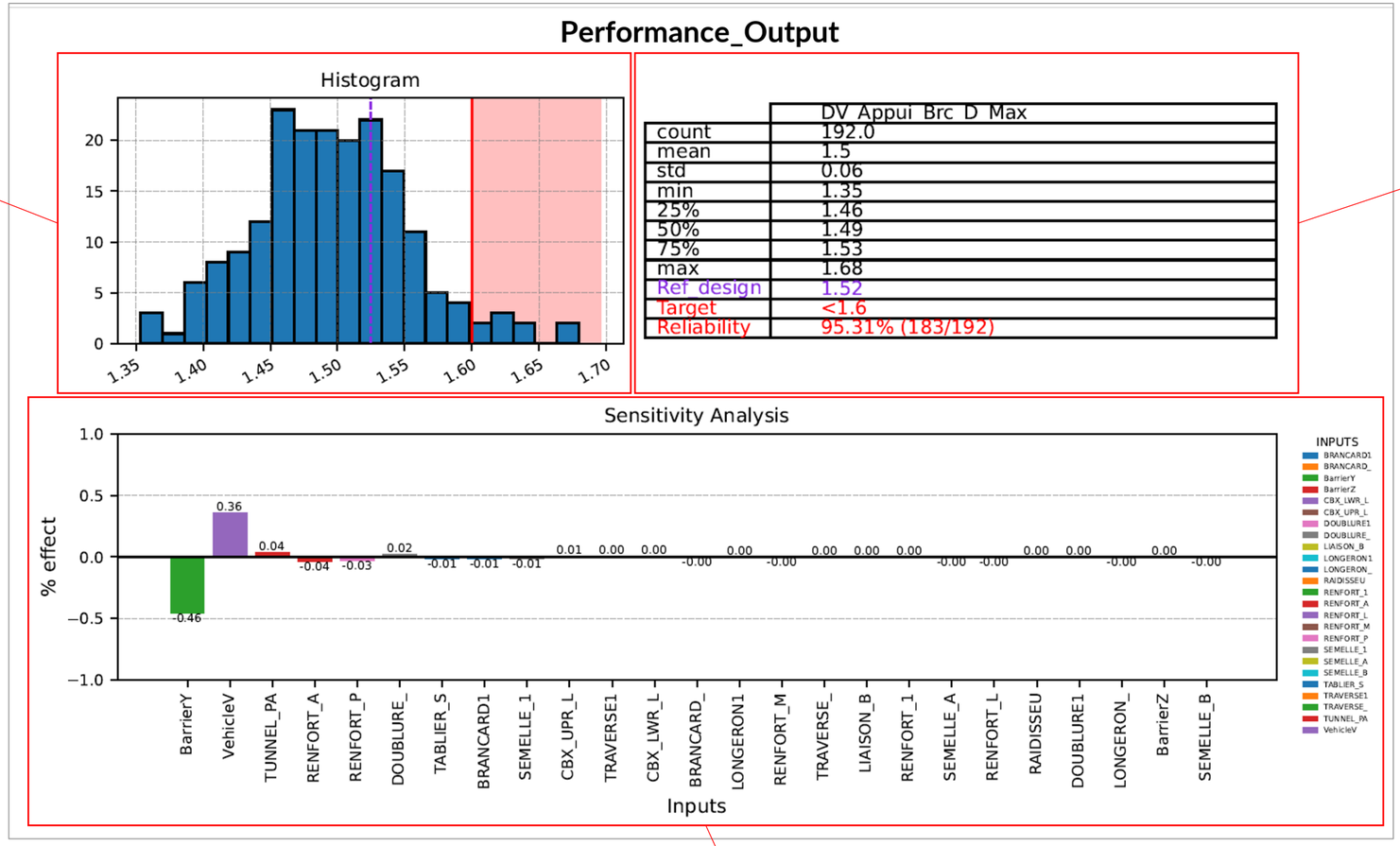
Benefits of pyConsole

- ❑ Leverage Python's flexibility while accessing data from the modeFRONTIER environment
- ❑ ModeFrontier features currently used with pyConsole
  - Table management
    - Data retrieval
    - Column type identification (input/output/constraint/...)
  - Trained RSM
  - Sensitivity Analysis table results

Automatic PDF generation: 2 pages per output

- Page 1: Output analysis overview

Histogram chart with:
- Reference design value
- Unfeasible area

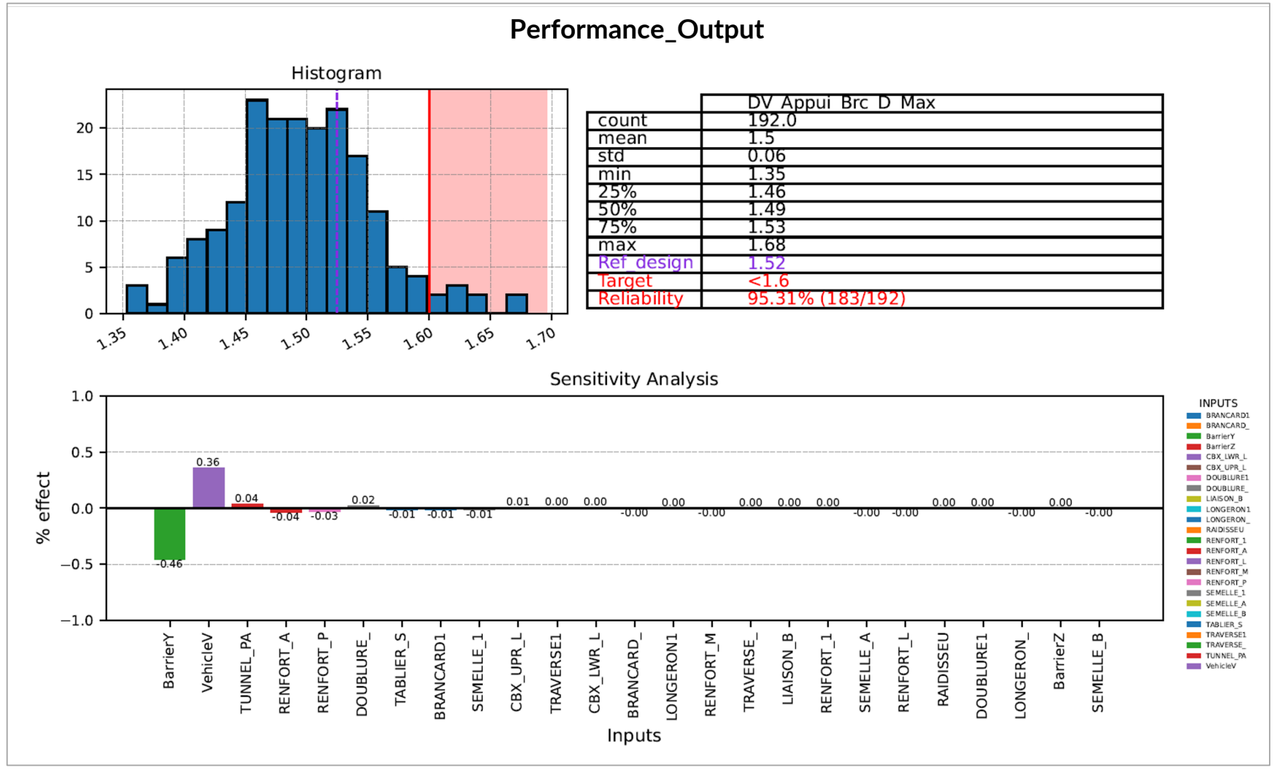


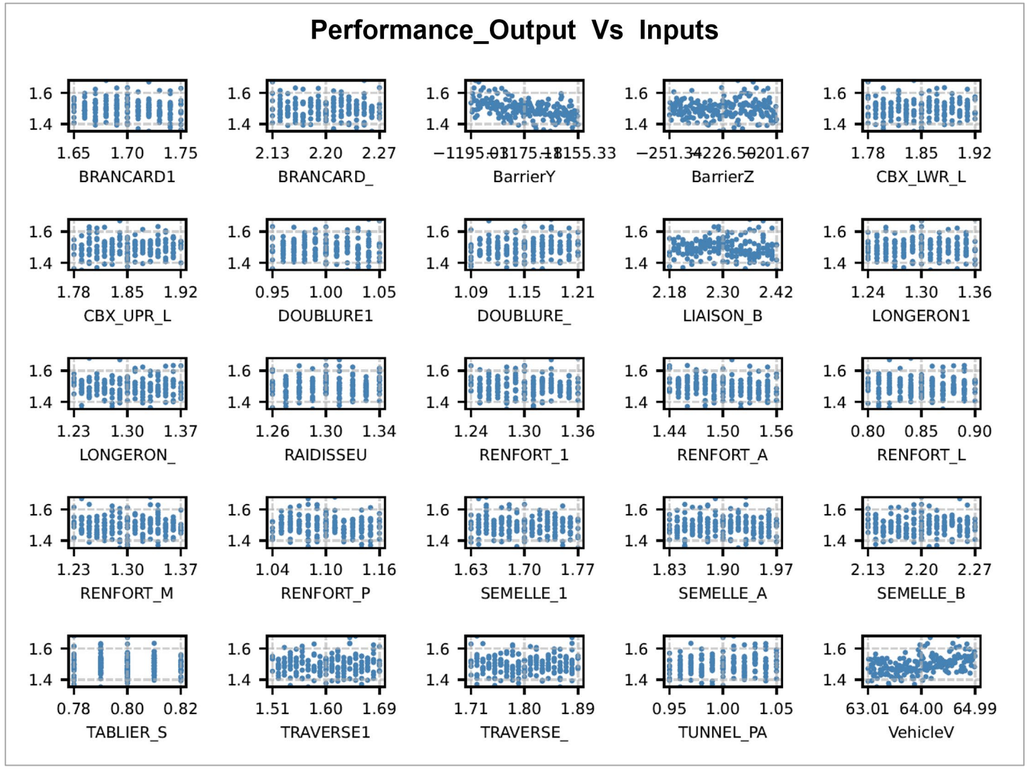
Output statistics:
- Min/max/quartiles/...
- Reliability

Sensitivity:
- Signed SSANOVA from modeFrontier

Automatic PDF generation: 2 pages per output

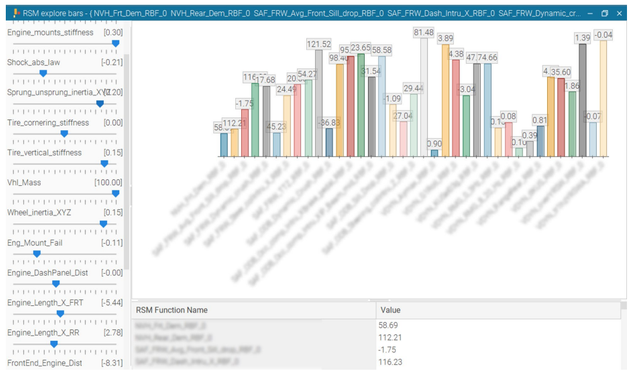
- ❑ Page 1: Output analysis overview
- ❑ Page 2: Scatter plots - Output vs. Inputs





GUI INTERFACE FOR RSM EVALUATION ON MULTIPLE OUTPUTS

From RSM explore bars …



…to dedicated GUI interface

Discipline reduced KPI

Sliders to set parameters values

Outputs value + KPI

Conclusion

SUMMARY & NEXT STEPS

Summary

Harmonization	Uniform setup for any MDO application
Combined expertise	Leverage the combined power of modeFrontier (the brain) and HPC (the muscles)
Robustness	Proven over 6 years, adapting to evolving HPC and software environments
Customization	Leverage the combined power of modeFrontier and Python to meet user needs
Automation	Generate synthesis reports for any number of inputs and outputs

Next steps

- Extend generic workflow usage to new loadcases/simulation fields
- Extend automated reports
- Integrate user RSM models and optimization strategies

um 2026

Thank you

esteco.com