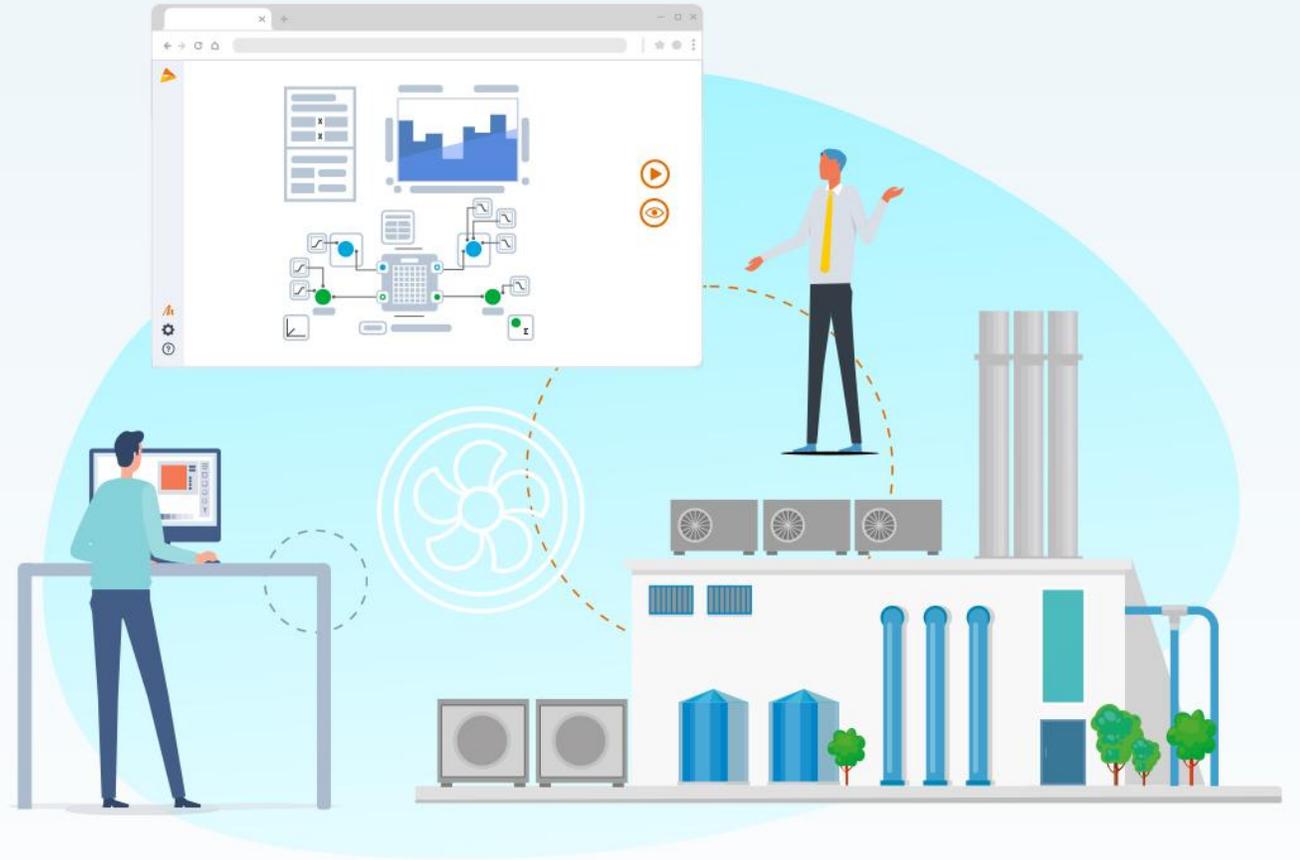


Introduction to Modelon Impact

Mathias Strandberg

Principal Application Engineer



Get to know Modelon Impact

- Core workflow
- Additional apps
- 2024.2 feature highlights

DEMO

Open a shared workspace and follow along!

[Impact workspace: GettingStartedWorkspace](#)

- Browse a workspace
- Investigate a model
- Parameter sweep

Workspace and Project manager

Maintain your work

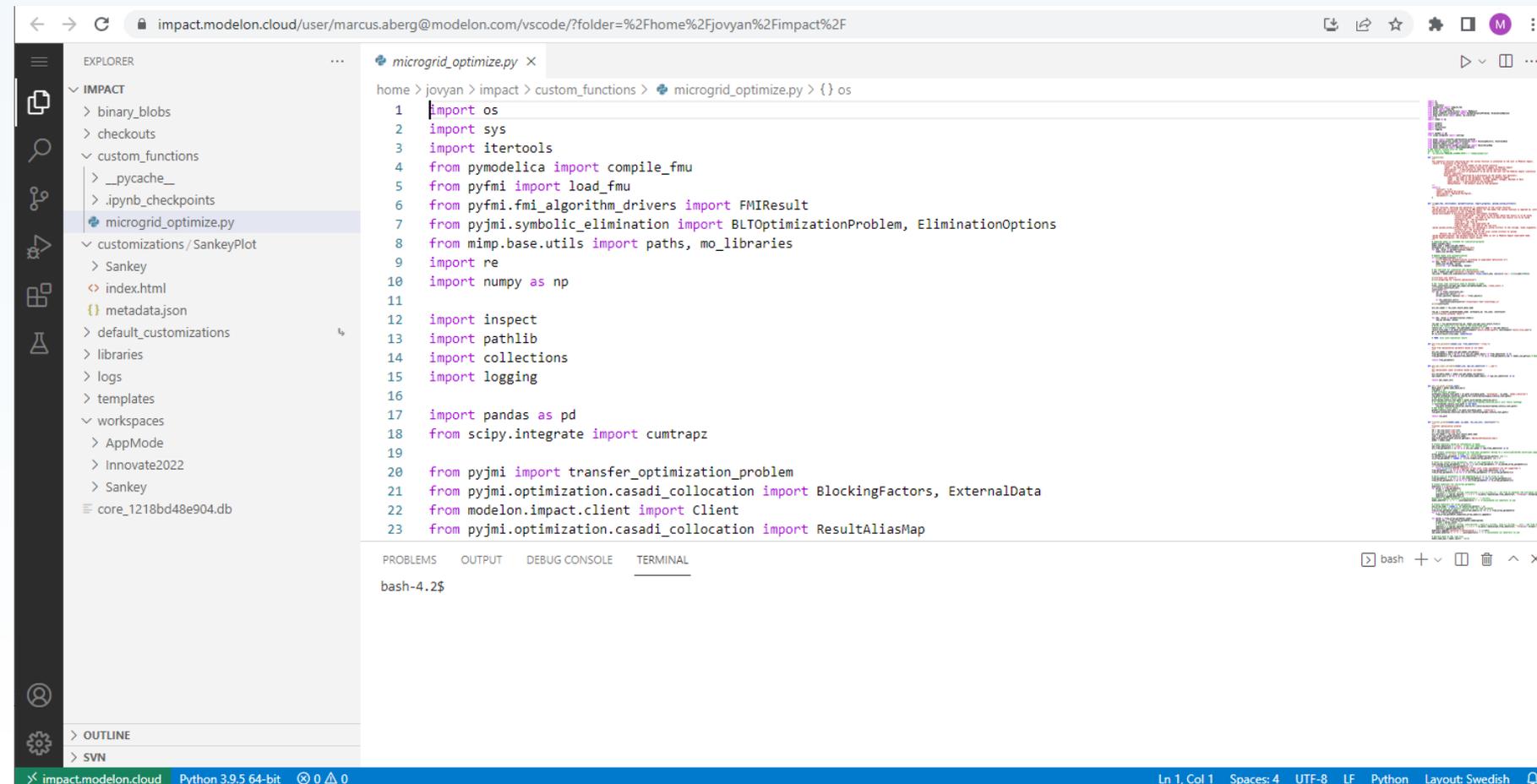
- Workspaces
- Projects
- External Libraries
- Sharing
- GIT credentials locker

The screenshot displays the Modelon workspace and project manager interface. On the left, a sidebar contains navigation options: Workspace (selected), Projects, Libraries, Published, and Credentials. The main area shows the 'GettingStartedWorkspace' with 'SHARE' and 'EDIT' buttons. Below this, it lists project statistics: Total Size: 167 MB, Fmus Size: 7.9 MB, and Experiments Size: 158 MB. The 'Projects' section shows a single project named 'GettingStartedProject' with a 'main' branch. The 'Dependencies' section lists three dependencies: Modelon 4.3.0+build.20, VaporCycle 2.11.0+build.26, and Modelica 4.0.0. At the bottom, there are links to 'Open in Modelon Impact' or 'VS Code'.

VS Code - Cloud

Web Version of Visual Studio Code

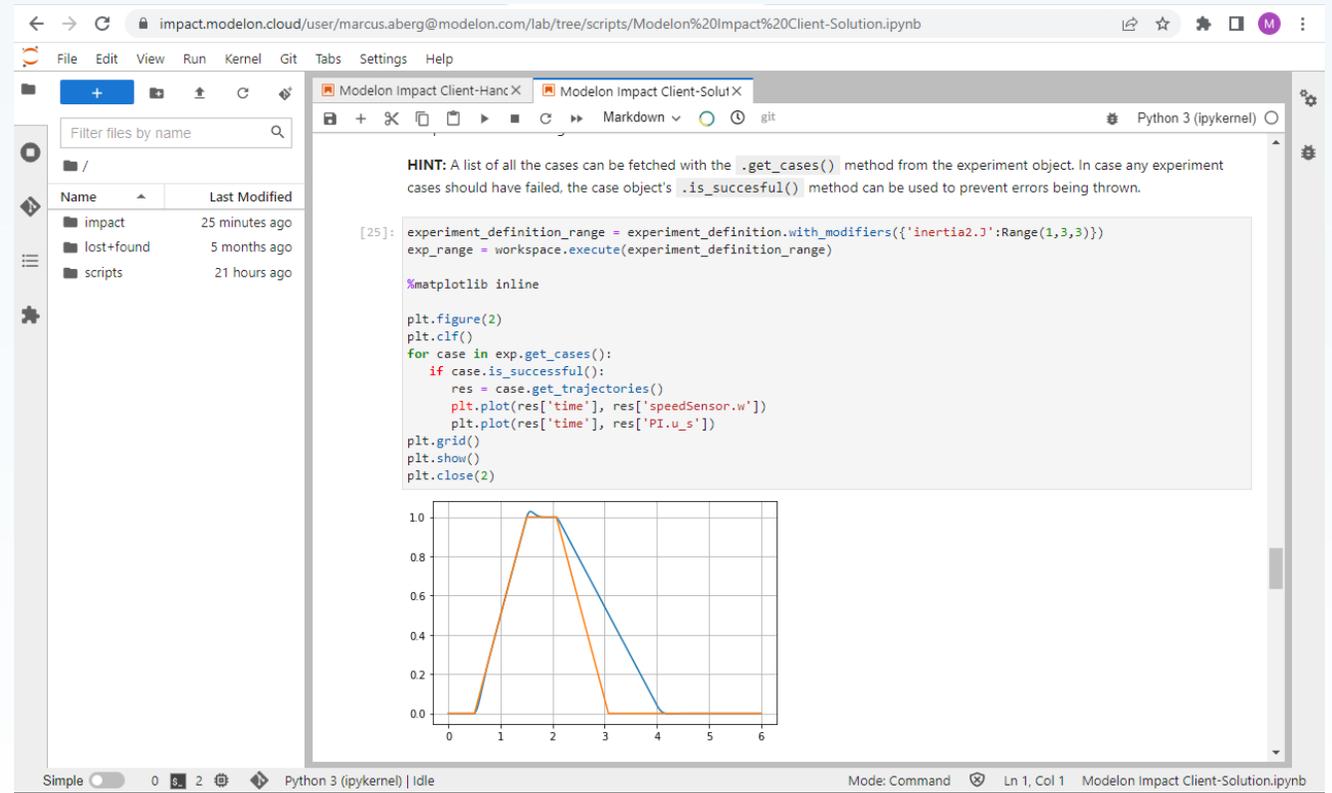
- File Explorer
- Coding IDE
- Version Control
- Debugger
- Terminal



JupyterLab - Cloud

Interactive development environment

- File Explorer
- Notebook Editor
- File Editor
- Terminal
- Python Shell



The screenshot displays the JupyterLab Cloud interface. On the left, a file explorer shows a directory structure with folders named 'impact', 'lost+found', and 'scripts'. The main area is a notebook editor with a Python kernel. The code in the notebook includes a hint about using `.get_cases()` and `.is_successful()` methods, followed by a code cell that defines an experiment range, executes it, and plots trajectories. The plot shows two trajectories (one orange, one blue) over time, with the orange trajectory peaking at time 2 and the blue trajectory peaking at time 4.

```
[25]: experiment_definition_range = experiment_definition.with_modifiers({'inertia2.J': Range(1,3,3)})
exp_range = workspace.execute(experiment_definition_range)

%matplotlib inline

plt.figure(2)
plt.clf()
for case in exp.get_cases():
    if case.is_successful():
        res = case.get_trajectories()
        plt.plot(res['time'], res['speedSensor.w'])
        plt.plot(res['time'], res['PI.u.s'])
plt.grid()
plt.show()
plt.close(2)
```

DEMO

- Open Jupyter Notebook
- Connect to workspace
- Run parameter sweep
- Plot the results
- Connect back to Impact UI



Build



2024.2 Modelon Library Suite

Talk to our experts to find out more!



Support for licensing of 3rd party commercial Modelica libraries



Analyze



Solver – Calibration against steady-state data (BETA)

Calibration: Reference file loader

1. Setup 2. Upload 3. Export

Choose a Result for Calibration

Select a result from the model that you want to calibrate.

Important Considerations

- The result should be from the experiment that is to be calibrated to ensure a
- Next step to identify parameters and variables

Calibration, Calibrate

xtol: 0.01 ftol: 0.01

max_iter: 100 max_fail: 0

output: calibration

baseline_result: Baseline Case 1

reference_result: CoupledClutchesMeasurement

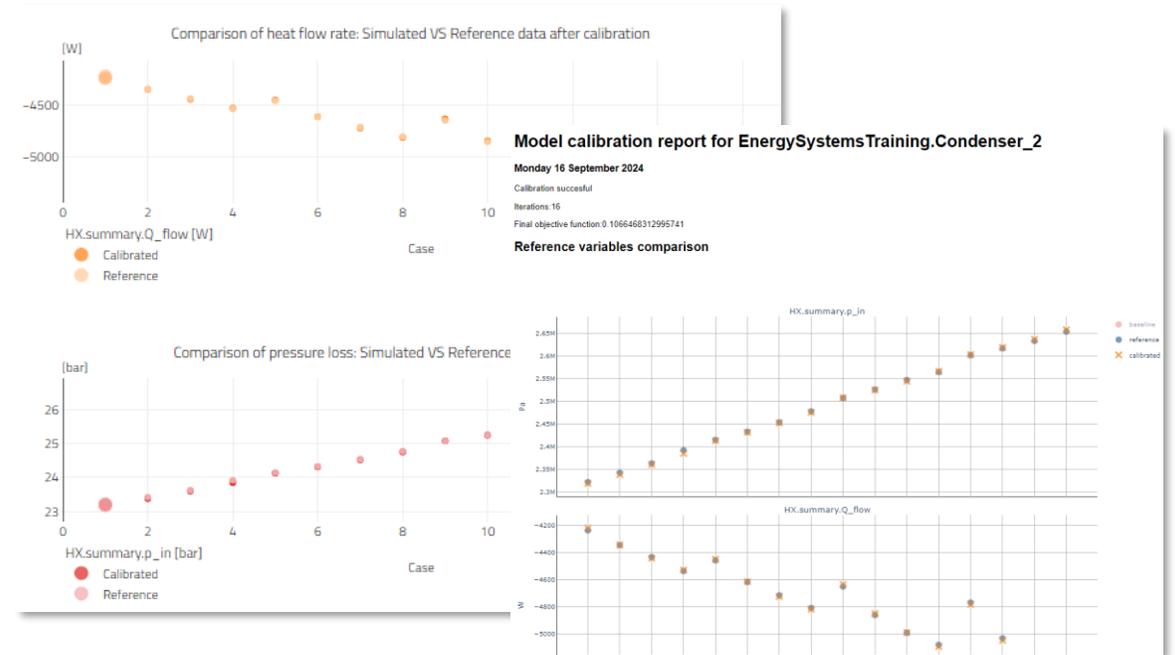
reference_variables: j1.w, j2.w, j3.w, j4.w

calibration_parameters: j2.j, j3.j, j4.j

Calibration Library

- Examples
- Custom functions
- Custom Web App

Note, requires Pro license

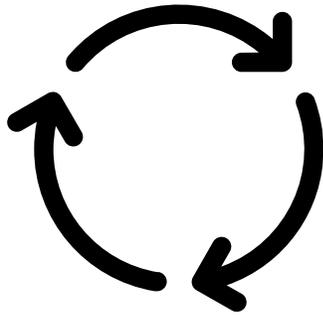


Calibration Capabilities

- Fast (steady-state) or Robust (dynamic simulation)
- Faster with Productivity licenses (parallel execution)
- Reference data: Single or multi-point
- Reference data: Measurement or simulation (high fidelity model)
- Analyze result: UI or calibration report



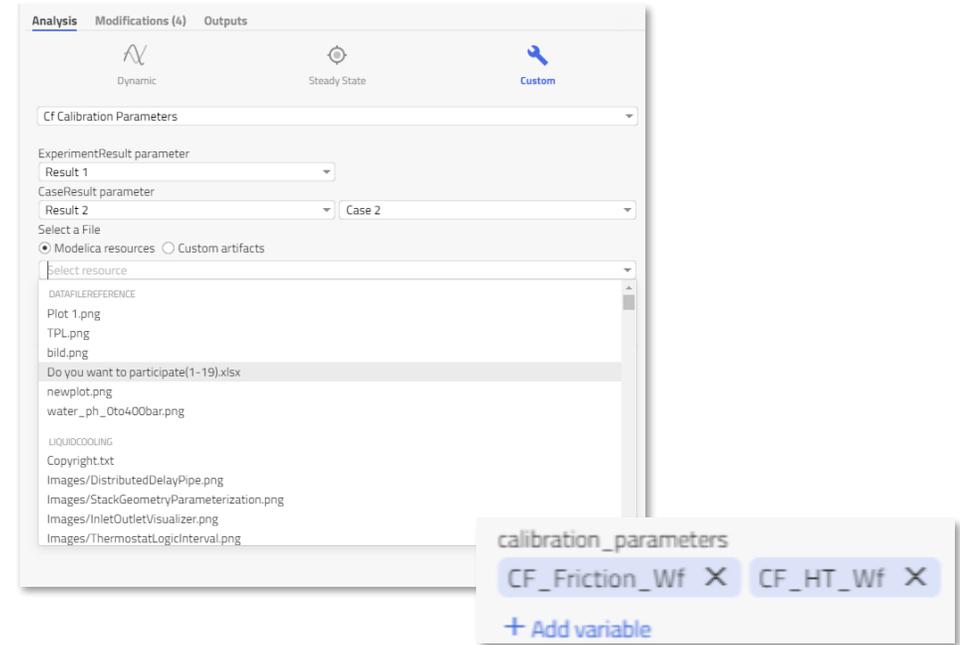
Solver – More powerful custom functions



Productivity add-on
enables parallel
execution

Major custom function API extensions

- Support of algorithms that iteratively schedule and execute cases (Orchestrator custom function)
- Access to data and results as input arguments



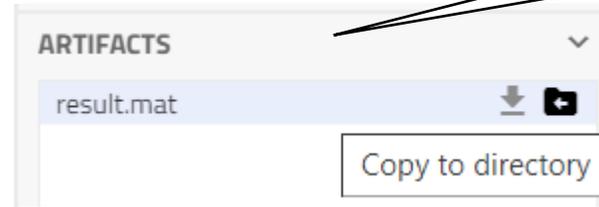
Custom function UI extensions

- Support to select results, cases, and files
- Support to select parameters & variables from a model



Visualization Tools – UI-support to display generated artifacts

Artifacts viewer is visible only when there are artifacts to display



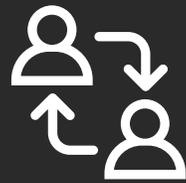
Files can be created from

- Model (e.g. DataAccess)
- Custom function (e.g. Linearization, Calibration)



Artifacts viewer

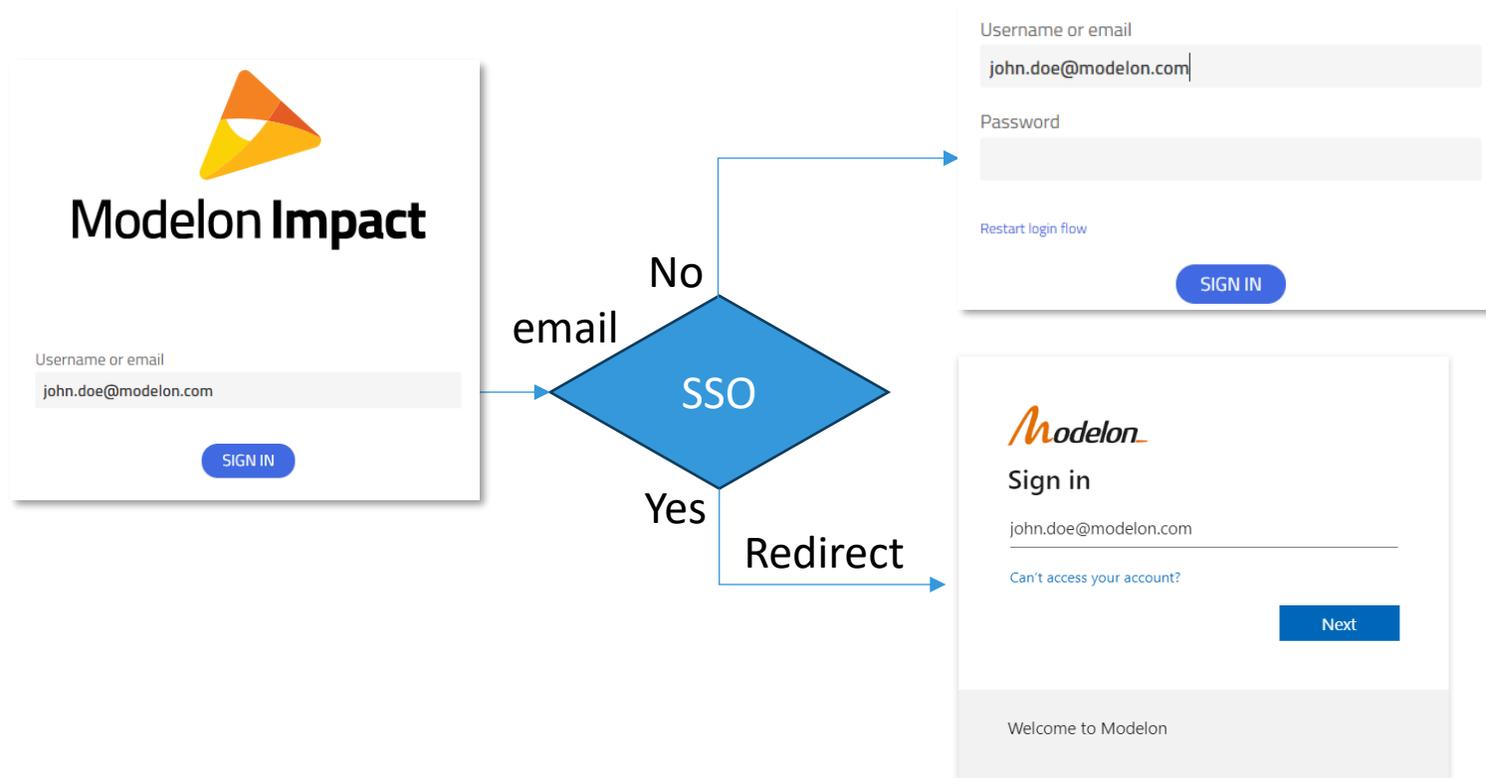
- Overview of all generated artifacts
- Preview artifact in a new browser tab (e.g. view a html-report)
- Copy to directory (e.g. data file to Resource folder)
- Download



Collaborate



Access - Single Sign-On for enterprise accounts

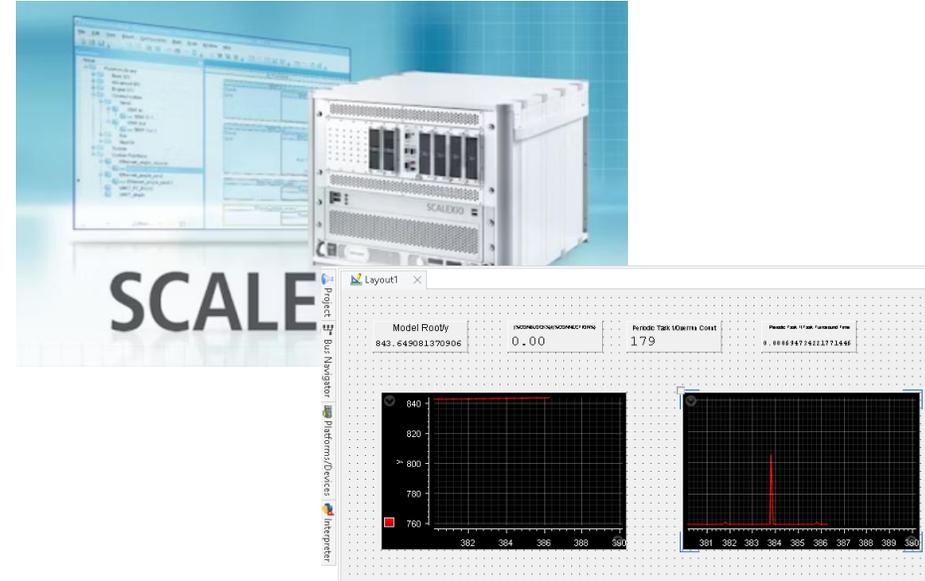
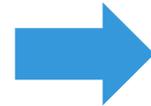
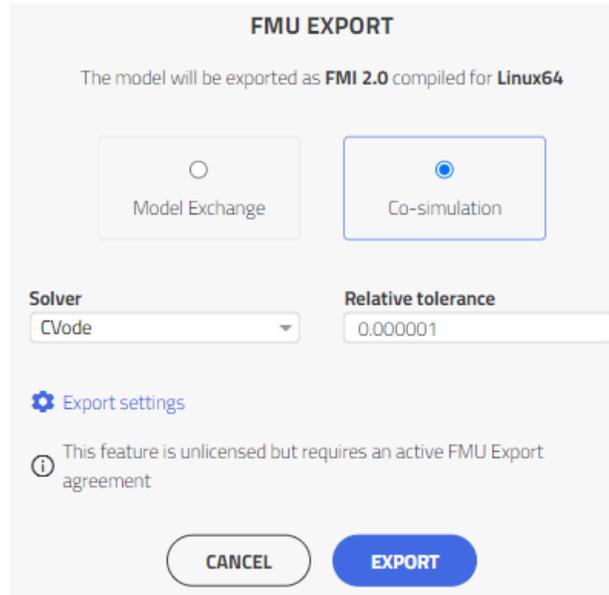


Single Sign-On @ Modelon Cloud

- Enable customers to configure and use existing authentication system incl security settings such as multi-factor authentication



Sharing Tools - FMU export to dSPACE SCALEXIO



Directly export FMUs into one of the most popular real-time platforms

- Export FMU binary into dSPACE SCALEXIO
- dSPACE has support for FMU binary import

Thank you!

Questions?

Modelon



Accurate Simulations. Better Decisions.