

TITLE STORY // PROCESS INTEGRATION

OPTISLANG CONNECTS

optiSLang enables the building of transparent CAE tool chains and combines them with parametric algorithms for a broader usage of CAE techniques.

One of the major challenges of today's CAE engineers is the increasing complexity of processes while results have to be delivered in shorter times. At the same time, within the engineering process, multiple disciplines like NVH, thermal-mechanical-electrical analyses, safety evaluation, tolerance management, cost etc., have to be considered. Improving one discipline could require a compromise in others. Cooperation in multiphysical simulation and multidisciplinary optimization becomes essential for workflows to manage different disciplines and teams.

Simulation Process Management

The daily work of an CAE engineer contains a high percentage of repetitive tasks. Such as reports and result extraction or many parts of model generation. It is also quite common to manually transfer results of one discipline as input into the next step of operation. For example, copying the geometry into a new directory where one starts to mesh it. This also applies for the results of the post processing. Maybe there are Matlab or Python scripts to process the work, sometimes it would mean copy & paste or even retyping into Excel sheets. This procedure is well-known to CAE en-

gineers. Also, customers are aware of how many mistakes might occur during this process. When different variants have to be compared for finding an optimal design, it can lead to an unsorted crowd of data. If simulations are done by different engineers, it can be very hard to compare the results among each other. Thus, many teams or companies started to standardize their working process, resulting in scripting environments with a mixture of Python, VBA, Perl, Bash, Matlab, etc.

optiSLang users run multiple design variants for Robust Design Optimization (RDO) and face the same issues. In the past, engineers had to write scripts to automatize the CAE process. Nowadays, because of multiphysical challenges, CAD and PLM systems, tasks become even more complex. New versions of tools are released every year. All parts of the system need to be connected and maintained. Already several years ago, customers in cooperation with Dynardo realized that a better support of process automation is needed. In order to fulfil this requirement, optiSLang has emerged from an RDO-tool with powerful algorithms to a Process Integration and Design Optimization (PIDO) tool. Thus, the software platform connects tools of customers to automatize their processes of design evaluation. optiSLang

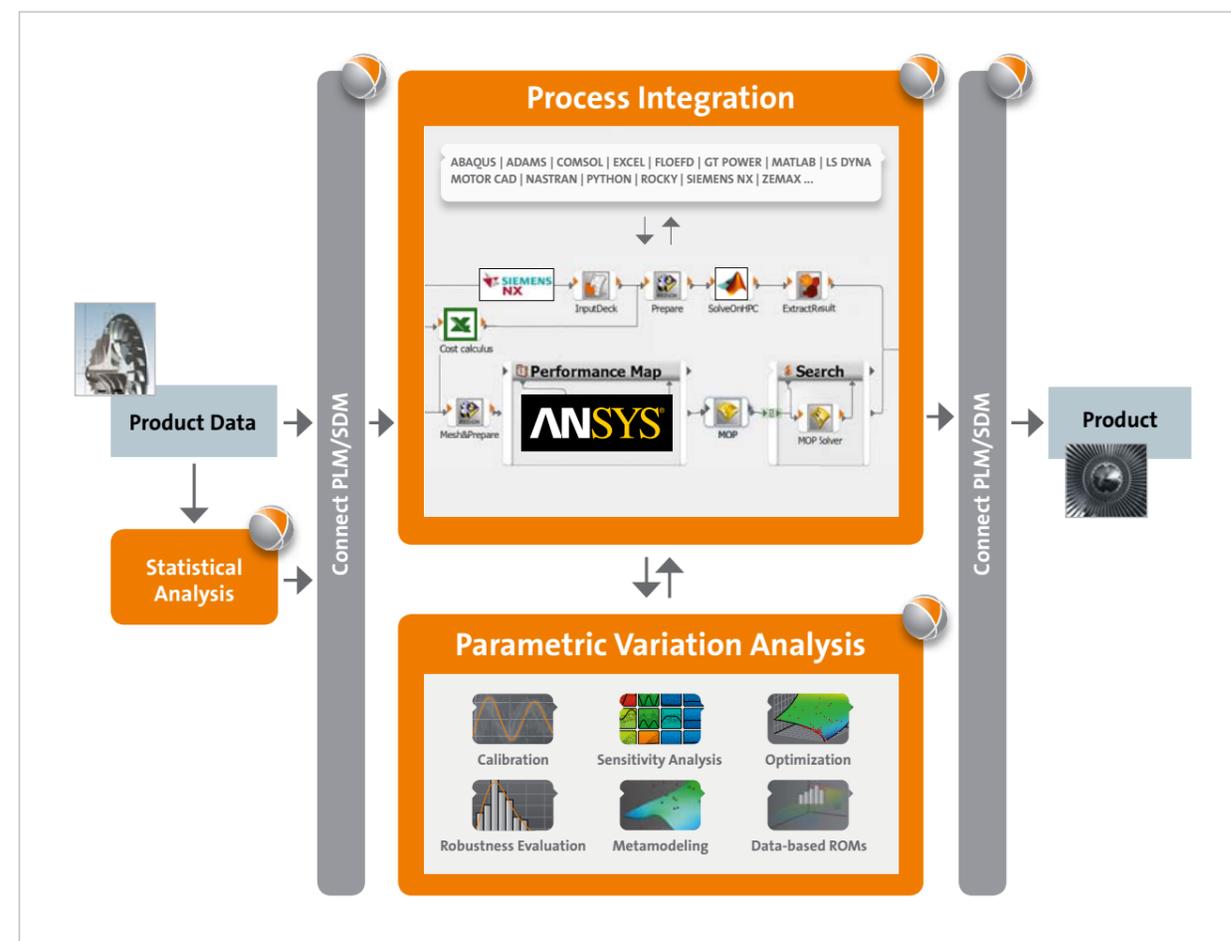


Fig. 1: Simulation Workflow Management with optiSLang

includes interfaces to all major software tools used in the virtual product development. For example, there are integrations of CAD parametric modeling with CATIA, Siemens NX and ProE. CAE parametric modeling is supported by linking ANSYS, ABAQUS, ADAMS, COMSOL, FLOEFD, GT POWER, LS-DYNA, MOTOR CAD, NASTRAN, ROCKY, ZEMAX and many others. Programming environments like EXCEL, MATLAB, PYTHON can also be applied (see Fig. 1). Dynardo established partnerships with many CAE/CAD/PLM vendors to develop and secure the support of these tools.

The coupling with optiSLang can be automated, either in a single solver process chain or in very complex multidisciplinary and multi-domain workflows. The workflow management allows users to combine several tools in sequences and iteration loops. Conditions, branches or nested loops can be set up graphically. Workflows can be stored as a template project and used again for sub-workflows in a collaborative project. Multiple disciplines can be handled in modular ways to be available in broader systems. This helps connecting different experts and teams.

With optiSLang the overall process is getting much more transparent compared to scripting solutions. As a tool for Simulation Process Management (SPM), it can build and

maintain complex tool chains. The modular process integration approach of building programmable nodes provides a very economical way to standardize and automatize CAE workflows. There are many examples of successful projects with Dynardo's customers like Bosch, Daimler, et al.

Example of connecting multiple disciplines

With optiSLang's GUI, a simulation workflow can be built graphically. Therein, any tool can be connected into sequences or put into loops. Figure 2 (see next page) shows such an optimization workflow. The following provides a more detailed insight.

Cost calculation via Excel sheet

In most applications, cost has to be minimized while functional requirements have to be optimized. Therefore, the two disciplines have to be kept in mind and evaluated. Often the cost calculation allows simulations only to be performed if designs are economically valid and if the cost of the design variant is not too high. Thus, expenses and benefits of simulation have to be figured in a balance.

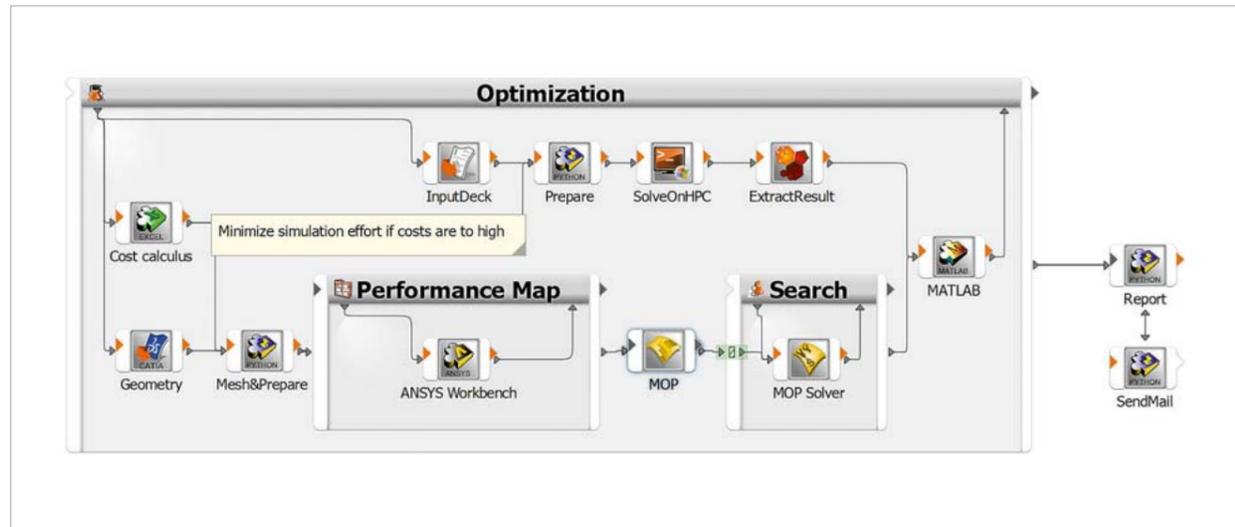


Fig. 2: Optimization workflow in optiSlang considering structural costs and metric of performance map, running several solvers and using HPC

Inhouse code evaluation

This part of the workflow (see upper part Fig.2) starts also with the same parameters like in the excel sheet but, in this case, the calculation takes longer to solve – so it is submitted to a Linux-cluster and runs multiple design points simultaneously. The results are automatically extracted and forwarded to a successive Matlab node.

Load case calculation for given geometry variation

This part (see lower part of Fig. 2) consists of a nested chain, which briefly builds a performance grid for each geometry variation proposed by the optimizer. Therefore, a custom Design of Experiment (DOE) is used. The design points are then forwarded to optiSlang’s MOP algorithm for metamodeling. Based on meta-models, the worst loading case for the geometry is searched by optiSlang’s optimizer. The worst case scenario is forwarded to the Matlab node.

Post processing

The Matlab node takes the results of 3 prior parts of the workflow and combines them for the optimization task. Additionally, some graphics are automatically produced. All the results are forwarded to the optimizer which can now evaluate them and generate the next iterations until they converge.

If the optimizer finds optimal designs, a report is generated and send via mail. The whole process runs automatically. In this way, the Excel sheet and the CATIA model is updated for each variation. Subsequently, ANSYS Workbench will be started. As additional benefits, optiSlang is capable of handling failed designs and economically controls the maximum runtime of a solver.

Supporting collaborative, flexible and standardized workflows, optiSlang is the platform for efficient, future-oriented CAE-teamwork.

How can CAE expertise be forwarded to other team members?

In the past, there was a relatively small group of CAE engineers with such a high level of expertise to perform key analysis and useful data extraction with tools which were difficult and highly manual to handle. Democratization in CAE-based product development should mean to make analysis tools and results accessible, not only to analysts and advanced simulation experts but also to all engineers involved in the development process. Thus, collaborative work within CAD/CAE teams, which are responsible for different “physics” or disciplines of the product, can share results and processes. optiSlang provides powerful interfaces to publish created RDO and CAE workflows. The power of optiSlang’s RDO and Simulation Process Management can be easily integrated into customized platforms.

Example of a collaborative workflow

How such an integration works can be explained using the example of a digital twin. A machine has to be analyzed to figure out when to maintain or repair it. For this reason, sensors are installed. However, this measured data does not provide information how the performance of the machine is effected in the future. Therefore, sensor data has to be combined with 3D-simulation for a continuously updated calibration of important machine parameters. As a result, necessary information can be extracted to optimize maintenance cycles. Fig. 3 shows such a published workflow. The workflow receives the sensor data and calibrates the machine condition. Once this data is available, a condition check can be performed when maintenance will be required. With such a workflow any engineer just has to:

- download the sensor signal
- start the workflow
- forward the results to other team member

To further automate the process, the user defines the workflow

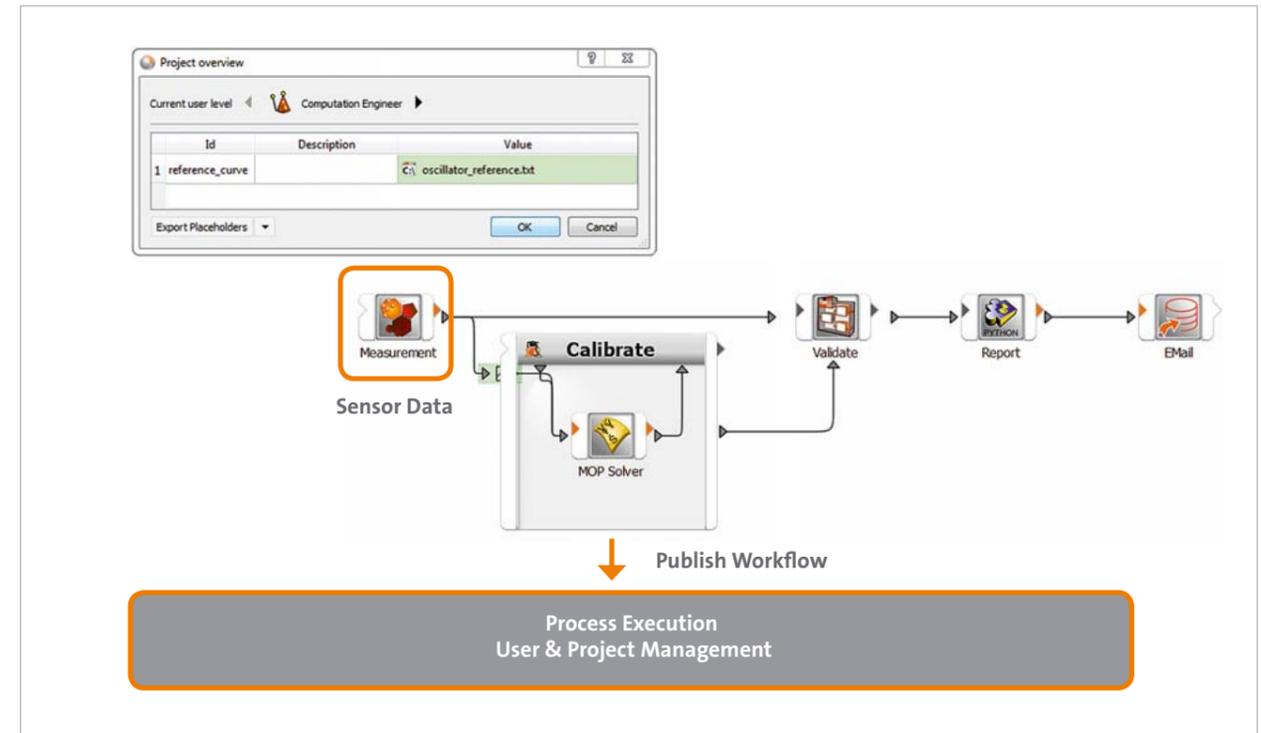


Fig. 3: Digital Twin - Calibration workflow

Start in web browser (no local installation needed) an upload measured curve

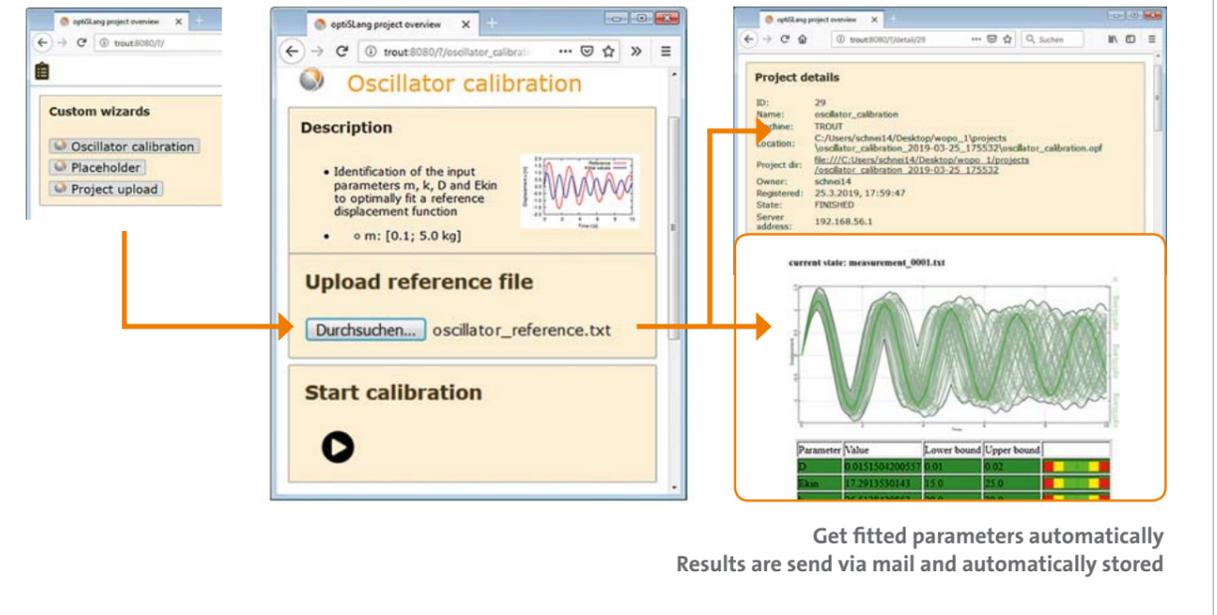


Fig. 4: In-field engineer use calibration workflow

as a template with only the sensor data as a placeholder. Then, the data is uploaded to a Process Execution & Data Management System (see Fig. 3). Every team member can use this procedure in a Process Execution System. In-field engineers just need a browser to have access to all data published in the workflow.

Authors //

David Schneider, Henning Schwarz (Dynardo GmbH)