

## MULTIDISCIPLINARY DESIGN OPTIMIZATION (MDO) USING ANSA/ $\mu$ ETA POSTPROCESSOR AND ISIGHT

Frederic KRABCHI  
Simulia, France

**Abstract:** Increasingly, industries are using process automation tools in simulation-based design processes to reduce development time and improve product performance and quality. The critical success factor is the ability to capture existing processes involving various design tools in order to help engineers making trade-off studies in the early design cycle. Isight is a simulation process automation framework with design exploration capabilities. It functions like a robot to capture the design process and logic in an easily understood workflow. In this project, we have studied the integration of ANSA and  $\mu$ ETA PostProcessor to automate the model building (ANSA) and result postprocessing ( $\mu$ ETA) in Isight. The intention is twofold—to realize the rapid integration of these tools in the Isight workflow and to apply this process with a number of multidisciplinary studies combined with optimization strategies. To investigate the possibility of integrating these software products, three simulation use cases are established to demonstrate how the processes are captured, automated, and encapsulated in various Isight design exploration strategies (Design of Experiments, Optimizers, Approximations, Quality Methods, etc.).

**Keywords:** multidiscipline, integration, batchmesher, morphing, optimization.

### 1 Introduction

In order to accelerate the virtual product development, iterative computational analysis of various design concepts with geometry changes needs to be automated and used systematically. There are two different approaches:

- ✦ Seamless flow from computer-aided design (CAD) to computer-aided engineering (CAE).
- ✦ The so-called “mesh morphing” approach, a technique where the mesh of a finite element model is changed with a small number of shape-morphing parameters. This process does not involve CAD.

The automation of both approaches provides flexibility and efficiency in different stages of the development process, where different amounts of data are available.

Batch technologies and scripts used within ANSA and  $\mu$ ETA provide tools to automate preprocessing and postprocessing. Isight, on the other hand, connects a variety of applications, automates the execution of multiple simulations, and performs multidisciplinary design exploration and optimization. The objective of this study is to demonstrate the following three application scenarios using ANSA 12.1.6,  $\mu$ ETA 6.2.2 and Isight 3.5.3:

- 1) To drive the batch meshing module of ANSA using a Microsoft Excel spreadsheet, determine the mesh quality of the model, and write back the information to the same Excel spreadsheet. A DOE loop is then used to study the trade-off.
- 2) Methodology for CAD-to-CAE process using BETA tools. A methodology has been developed to drive a CAD-meshing-solve process using a parameterized CATIA V5 model, BETA-CAE pre/post tools, and a CATIA component (interface) within the Isight workflow.

3) Morphing Studies, or using Isight to conduct design studies with the mesh morphing capability. This approach is greatly simplified with ANSA's "optimization task" module. The module exports the design variables and the output FEA files.

## 2 Capture and automate the ANSA batch mesh for a design exploration within Isight

For a better understanding, we first explain how the ANSA batch mesh works in a standalone mode and in batch mode before illustrating the way to rapidly integrate it in an Isight workflow. Obviously, this integration technique remains valid with any simulation code that uses ASCII-based input and output.

### 2.1 Batch mesh module

Building quality meshes for Finite Element (FE) is time-consuming. The resulting mesh has to follow the geometry up to a certain level of accuracy to meet the quality mesh requirements expected by users. The ANSA batch mesh speeds up the meshing process of a CAD model with the objective of respecting the quality criteria provided. This one can be fully controlled by a GUI-driven tool within ANSA or with script files in batch mode.

In a standalone mode, the meshing tasks driven by the ANSA Batch Mesh Manager can execute different sets of meshing rules included in one or several scenarios depending on the number of parts of a model or assemblies. When setting up a batch mesh scenario, two sets of parameters are defined:

- ✓ The first contains the parameters for the mesh generation; the several feature treatment tools, and the quality improvement tasks.
- ✓ The second contains all quality requirements and provides the ability to define threshold ranges. Batch mesh also gives the ability to control a large number of features during mesh generation.

The two sets of parameters (mesh parameters and quality criteria) can be saved in two different files and re-used in other scenarios: the *sample\_parameters.ansa\_mpar* and the *sample\_quality.ansa\_qual* files. All parameters existing in the Batch Mesh Manager are written on these files, and are activated accordingly.

At the end of the batch mesh procedure, a report is generated which presents general statistics regarding the mesh process and the quality achieved, based on the criteria and parameters required.

In batch mode, the batchmesher is controlled with the ANSA *batchmesh.c* script file, available by default in the ANSA library. This manages the input and the output (I/O) files during the process:

- ✦ The CAD model provided by the user: \*.igs, \*.ansa, etc.
- ✦ The mesh parameters and the quality criteria files
- ✦ The error static report file: error\_stats.html.

To be in a pure batch background operation, the option "**-nogui**" has to be specified in the following executable batch command (here from a window prompt):

```
Install_path/ansa64.bat -nogui -i CAD_MODEL -exec "load_script: path\BatchMesh.c" -exec "BatchMesh()"
```

## 2.2 Integration with Isight

### 2.2.1 Overview

Isight is an engineering platform wherein a suite of tools is used to set up an automated workflow able to explore a design space to find the best trade-off. The software's advanced, interactive postprocessing tools let you see the design space from multiple points of view to ensure that you thoroughly understand the trade-offs leading to the best design decision. It can be automatically extended to an enterprise solution through the Fiper Application Control Server (ACS), which allows sharing workflows and components as well as webtop workflow execution in a secure environment.

The interoperability between computer programs and Isight can be realized with our Java System Development Kit (SDK), creating dedicated interfaces. These components are called activity components, interacting from Isight with external applications. The way to link the activity components and the external application can be done with APIs, or with the exchanges of ASCII files. In this second category, we have the simcode component, which is a kind of container packaging: two Data Exchanger and one OS Command components. This component can easily demonstrate the ability of Isight to execute external programs and manage their input and output files, particularly adapted with a batch technology application, and completed here with the Excel component for ease of use purposes. The process is therefore encapsulated for design strategy by process components such as DOE loop, Optimization loop, or others.

### 2.2.2 Integration of the ANSA Batch mesh within a simcode component

The integration is made with two different methods for non-expert and people having some basic Java knowledge. In this paper, we will call them static parsing mode and dynamic or generic parsing mode, referring to the way the parsing is done.

First of all, both methods user has to:

- Drag and drop the simcode component on the Isight workflow and open its interface.
- Upload the files to parse (quality criteria, mesh parameter and error report files) respectively on the I/O tabs, which will be re-used as a template.
- Write the ANSA batch command line in the command tab (which can be fully parametrized).
- Add the *batchmesh.c* script file and the CAD model as input files parameters in the Files tab of the simcode components.

In the static parsing mode, and thus in a common use, the user highlights an area in the I/O files to "teach" Isight where to write or read new parameter data (Figure 1).

In the dynamic parsing mode, some Java developments are used to parse automatically the I/O files, based mainly on resizable arrays. The input files are parsed according to the data provided in the input resizable arrays as well as the output files, but it depends mainly on the outputs which must be recovered.

---

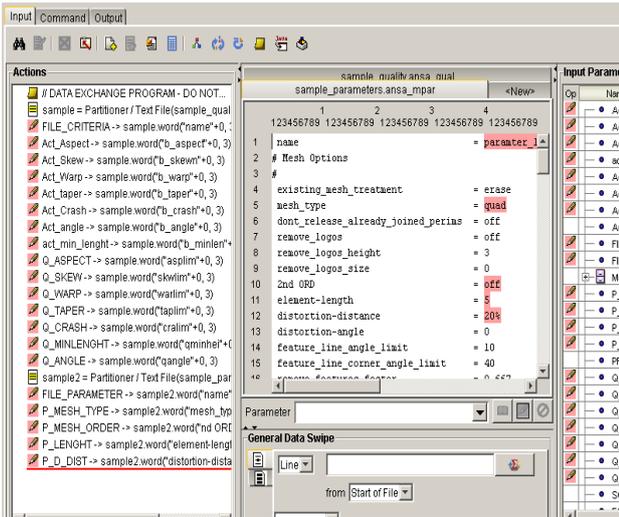


Figure 1: Static parsing on the input files

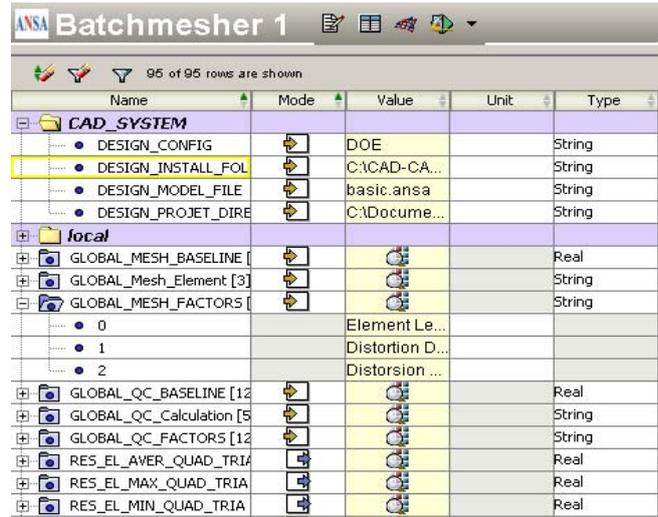
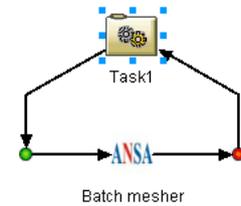


Figure 2: Parameters tab of the generic parsing

Whatever the method, all the parsed parameters are accessible from the Parameters tab of the component to modify their values. Moreover, with the generic parsing mode (Figure 2), a user is able to add or remove parameters by changing the dimension of the arrays, but he has to pay attention to use the exact parameter syntax.

As a first result, the integration of the ANSA batch mesh is done with both methods and the batch mesher simcodes are self-sufficient.



### 2.2.3 Integration of an Excel spreadsheet into an Excel component to drive the process

The Excel component is an activity component that communicates with Microsoft Excel workbooks. This component has two main functions: macro execution and parameter mapping.

Because Excel is widely used, we have opted to drive the entire process with an Excel spreadsheet. All the required information is merely put on it in line with the parsing methods implemented in the batch mesher simcode components. Named cells are also used to reference a parameter (a cell) or an array of parameters (a range of cells). When uploading the Excel spreadsheet into an Excel component, the label and the type (scalar or array) of the named cells are automatically detected. The user then has to select the parameters to employ and to define their I/O mode (Read or Write) in the list provided by the Excel component editor.

Therefore, two Excel components have been placed at each side of the simcode components.

- ✦ The first consists of reading the data uploaded from the Excel spreadsheet to the workflow as the mesh parameters, the quality criteria parameters, or the CAE system parameters (ANSA install folder, ANSA model name and its path).
- ✦ The second writes back on the same Excel spreadsheet the results parsed from an error static report.

Two different Excel spreadsheets have been implemented in line with the parsing methods.

During the static parsing (Figure 1), Isight handles a constant number of parameters in the parsed files. Obviously the same number of parameters has been written into a simple Excel spreadsheet (Figure 3) that also has the CAE system parameters. In that case, the named cells reference constant values.

BATCHMESHING ANSA		
FOLDER - PROJ-FILES		Ansa command FOLDER: C:\CAD-CAE\ANS\Ansa_v12.1.6
Source Folder	C:\Documents and Settings\XTK.DS\Desktop\DEV_BETA\STATIC	
Project name	basic.ansa	
File name: quality_criteria	quality_1	
File name: parameters	parameter_1	
QUALITY CRITERIA		
CRITERIA	y/n	VALUE
Aspect	y	3
Skew	y	45
Warp	y	5
Taper	y	0.25
Crash	n	1.00E-06
Angle	n	45
Minlength	n	2.5
PARAMETER		
FACTORS		VALUE
Mesh_type	quad/tria	quad
2nd order	off/on	off
Element length		25
Distance_distortion		20%
RESULTS (% beyond the limits)		
TAPER		
SKEW		
ASPECT RATIO		

Figure 3: Simple Excel spreadsheet

With the generic parsing, various numbers of parameters can be parsed because they are included in resizable arrays. That's why named cells used in this other Excel spreadsheet reference a range of cells (array of parameters). It can be noted that removing or adding a parameter in an array could be done between the bounds of the named cells. Furthermore, the user has to know the correct parameter syntax when adding one.

To simplify this procedure, a wizard has been elaborated within the Excel workbook done with Excel VBA code. This wizard mimics the ANSA Batch Mesh Manager, to finally generate automatically an Excel spreadsheet driver for three different designs (Figure 4): to check the quality mesh of a model, and to use the component for a simple loop, a DOE loop, or an Optimization loop.

The figure displays four sequential screenshots of an Excel wizard interface for ANSA Batch Meshing. The first screenshot, titled 'ANSA - BATCHMESHING', shows system information (INSTALL FOLDER: C:\CAD-CAE\ANS\Ansa\_v12.1.6) and model information (PROJECT DIRECTORY: C:\Documents and Settings\XTK.DS\Desktop\DEV\_BETA, MODEL FILE: basic.ansa). The second screenshot, 'Mesh Parameters', shows configuration options for Shell Mesh, including Mesh type (quad), Element Length (5), Distortion Distance (10), and Distortion Angle. The third screenshot, 'SUM UP', shows a table of quality criteria parameters with columns for NAME, BASELINE, LOWER, UPPER, OBJECTIVE, TARGET, SCALE, WEIGHT, and VALUE. The fourth screenshot, 'QUALITY CRITERIA', shows configuration options for Shells and Solids, including STATUT, THRESHOLDS (MIN-MAX), and CALCULATION (NASTRAN, LS-DYNA).

Figure 4: Generic Excel spreadsheet done with a wizard implemented with Excel VBA code

### 2.2.4 Mapping

The components are initially independent. To spread data over the workflow, parameters and parameter files need to be connected from one component to another. Within Isight, it corresponds to map them from one component to another by drawing a line between each of them in the Mapping window (Figure 5).

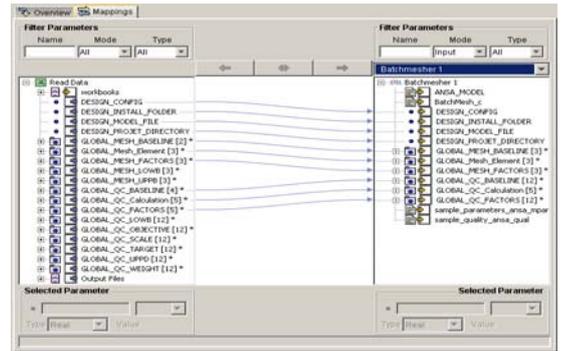


Figure 5: Mapping window

### 2.2.5 Simple loop

To finish the process integration, an OS Command component has been added at the beginning of the workflow to open automatically the Excel spreadsheet.

The steps executed during a simple loop (Figure 6) are the following.

Isight:

- Opens automatically the user-defined Excel spreadsheet to let the user update it.
- Reads the updated data and re-writes the Mesh parameters and the quality criteria files
- Runs ANSA in batch mode with the Batchmesh.c script file.

ANSA:

- Uploads the updated files and the ANSA file.
- Runs the batch mesher and exports the error static report.

Isight parses the quality report and writes back the results in the same Excel spreadsheet.

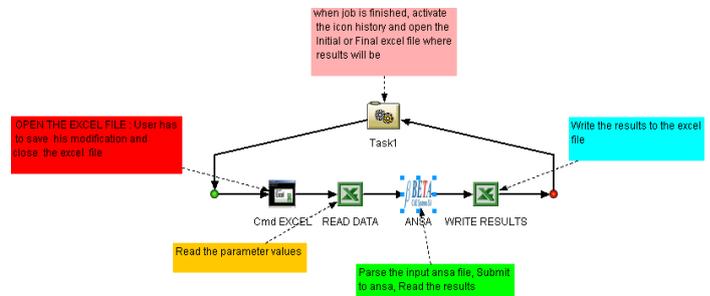


Figure 6: Batch mesh during a simple loop

### 2.2.6 DOE loop

The batch mesher component is encapsulated by a process component whose function is to be executed a number of times. The I/O parameters are automatically detected by the DOE component.

When opening the DOE editor (Figure 7), the user has to define:

- 1) *The DOE technique among a dozens of algorithm*
- 2) *The number of points if necessary*
- 3) *The design variables and their bounds*
- 4) *The objectives functions for the postprocessing*

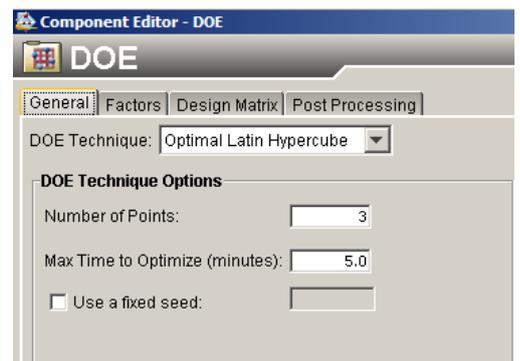


Figure 7: DOE Component Editor

To change the setup of the DOE after a first run, the user has two possibilities, depending on how the batch mesher component was developed:

- ⇒ With static parsing, the DOE needs to be re-configured in its editor
  - ⇒ With generic parsing, the modification can be applied directly in the Excel spreadsheet
- Indeed, the entire process is directly driven by this Excel spreadsheet.

2.2.7 Scenarios driven by the generic Excel spreadsheet

As shown previously, the ANSA batch mesher coupled with Isight can be used in different ways:

⇒ Check the quality mesh of an FE model

The “Freeze Existing Mesh” option is activated on the mesh parameter file. Because the model provided is already meshed, in that configuration the current meshing is therefore frozen and the error static report will be focused on the Existing Mesh.

⇒ Batch meshing with a Man in the Loop

To be in a **Man-In-the Loop** simulation program, the option “-nogui” in the command line has to be removed. The meshing sequence is therefore applied in real time with the ANSA window displayed. Moreover, the user can check the quality mesh and apply manual improvement if necessary. In that case, the modified ANSA model must be saved in the default folder as well as its new quality report in order to take into account these modifications in the Excel spreadsheet. The process will continue as soon as the ANSA application is closed.

⇒ Use the batch mesher on a DOE loop

In the following example, the use case is a strut tower (Figure 8). An Optimal Latin Hypercube technique with ten points covers the design space based on eight design variables on input. The objectives functions were to minimize the response of six quality criteria.

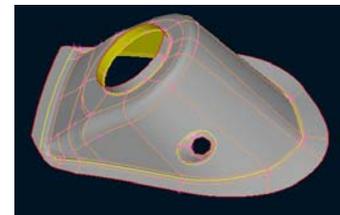


Figure 8: Strut tower (courtesy of Beta-CAE)

SUM UP								
DESIGN STRATEGY		DOE						
INPUT PARAMETERS								
NAME	BASELINE	LOWER	UPPER					
Element Length	5	4	8					
Distortion Distance	10	5	10					
QUALITY CRITERIA								
STATIC								
PARAMETERS	BASELINE	LOWER	UPPER	OBJECTIVE	TARGET	SCALE	WEIGHT	RESULTS
aspect ratio	3	2,7	4	minimize		1	1	0
skewness	45	30	50	minimize		1	1	0
warping	10	2	10	minimize		1	1	0
taper	0,25	0,2	0,4	minimize		1	1	0
element length	5	3	9	minimize		1	1	0
triangles %	5	1	5	minimize		1	1	0

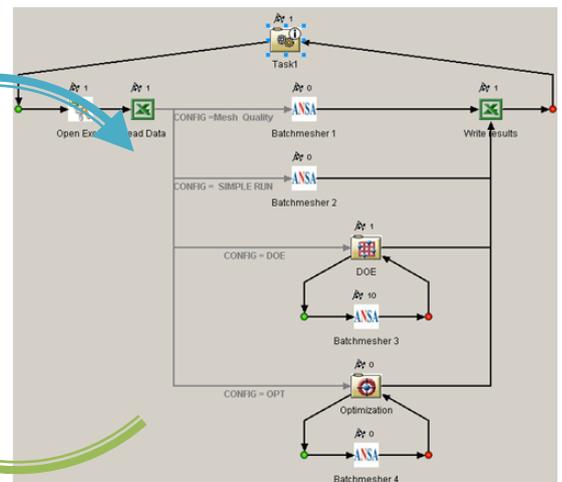


Figure 9: DOE loop example

At the end of the sequence, the best trade-off of the DOE loop is written back in the Excel spreadsheet (Figure 9). The objectives were achieved because the resulting percentage of off elements of each criterion is null. Afterwards, we have done an approximation study with the previously collected DOE data set, allowing us insight into I/O dependencies (Figure 10).

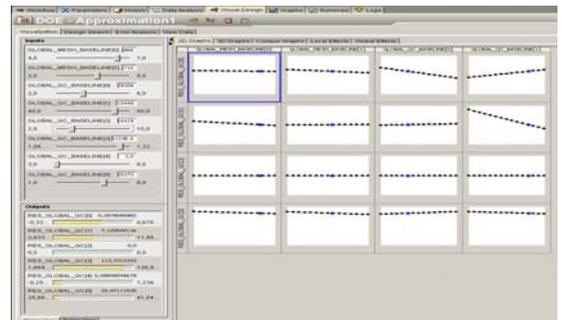


Figure 10: I/O dependencies

### 3 CAD to CAE with BETA-CAE product

Despite the wide use of CAD and CAE systems, the two processes are still not completely connected. We proposed to tackle the problem with the integration of BETA-CAE products for CAE preprocessing and postprocessing within the Isight platform and its components. A methodology was created around a use case to illustrate such a process. The model comprises a hierarchy of CAD-CAE features such as parts, meshing, loads, boundary conditions, and postprocessing. Information relating to both design and analysis can be specified and modified. The specified information from the design process is used to activate relevant CAE analysis tasks, thus supporting integration from the CAD to CAE process. If any of the specified design constraints is not satisfied from the CAE results, the initial model can be modified and the CAE analysis executed again. To couple CAD and CAE systems, CAD and CAE components within Isight have been used.

#### CAD components

The purpose of a CAD component is to expose and modify automatically the native parametric values of a CAD system as geometric parameters and mass properties, and to save the updated model in a native or neutral CAD format compatible with the CAE system. These parameters and the updated files can generate downstream CAE.

Our CAD components are developed by our partner Translation Technology in Spokane, WA on the basis of their Acc-u-trans CAD product, providing seamless access to several leading platforms, supporting CATIA V5, Unigraphics NX, and Pro/ENGINEER software. Communicating through the APIs of the programs, scripts are not necessary, and thus CAD components can rapidly manage large models even by a non-CAD expert.

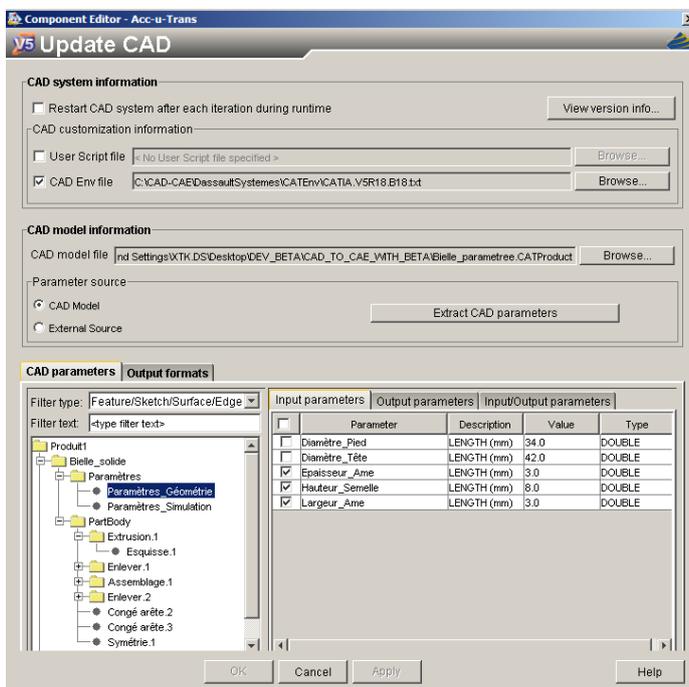


Figure 11: CATIA V5 component interface

For example, the CATIA V5 component (Figure 11) can easily dialogue with CATIA V5 to pilot parameters of CAD models or assemblies. It also has the functionality to support CATAnalysis files, exchange with embedded CATIA applications (extracting and submitting FEA parameters to drivers), and send a VBA script file to be executed into CATIA (automation of a complex, time-consuming task such as the generation of an FE spot weld, etc.).

## CAE components

SIMULIA has various CAE components, such as the NASTRAN, LSDYNA, ANSYS, MADYMO, ABAQUS, etc., that work in the same way as the CAD components.

On the other side, the preprocessor ANSA and the postprocessor μETA bring together advanced capabilities with specialized batch technology modules for process automation, adapted for a multidisciplinary purpose. These tools are captured with simcode components as already shown previously to mimic our CAE components, which merge CAD and CAE information to generate input decks and also extract outputs from the results files.

Coupling the Accu-trans CAD component with Isight creates an advanced tool, which facilitates:

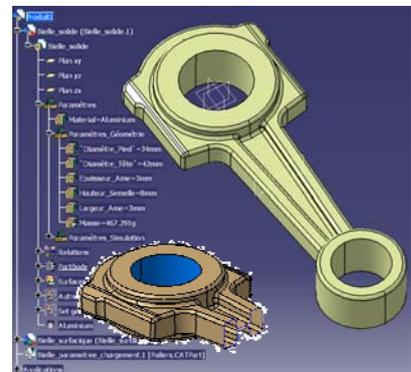
- ✓ Interoperation between CAD modeling and computing codes,
- ✓ Implementation of coupling between CAD & CAE codes,
- ✓ Providing generic user interfaces that are user-friendly and efficient,
- ✓ Re-use of generated process,
- ✓ Postprocessing, reporting, and visualization.

### 3.1 Description of a test case

The CAD model is a 3D CATIA V5 parameterized piston rod created by Digital Product Simulation, specialized in the development of CAD and CAE tools within the CATIA V5 platform. The case study is a tensile strength simulation wherein the tensile forces are applied on the large end and the small end is clamped through rigid elements. The aim of this part was to develop a methodology according to the working hypotheses and the problem definition.

#### **Working hypothesis:**

- ⇒ Boundaries and loads are applied respectively on the large and small ends (inner diameter of the holes), and must not be modified by the parameterization.
- ⇒ Preprocessing with ANSA (batch mesh module, loads, BCs).
- ⇒ Include specific cards after the export of the FEA from ANSA (e.g., materials).
- ⇒ Computation with Abaqus
- ⇒ Postprocessing of the result file within μETA



#### **Problem definition:**

The design problem is to understand the influence of three design variables of a connecting rod section (Figure 12) to outputs as mass and Von Mises stress. Furthermore, during the process the quality mesh has to be preserved.

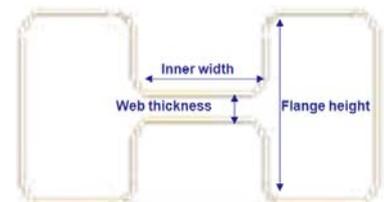


Figure 12: Conrod cross-section

### 3.2 Methodology

The CAD model has been built with two files: a CATProduct and a CATPart. The CATProduct contains the 3D model and the parameters to drive, whereas in the CATPart we automate the extraction of the skin of the model without the constraints surfaces where the boundaries and loads are applied. These surfaces are placed and constrained in ANSA as a pre-existing file. When extracting the parameters of the CAD model into a first CATIA component, the three geometrical parameters have been selected on input (Figure 11) and the mass parameter on output. The updated model is first saved in a native format, therefore the CATPart is transformed into a second CATIA component on an .igs format file before sending it to ANSA (Figure 13 - point 1).

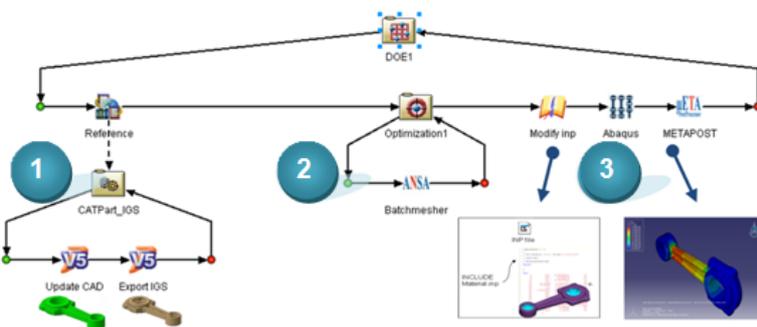


Figure 13: CAD-to-CAE workflow of a piston rod

In the meshing part (Figure 13 - point 2), it three main scenarios have been created within the ANSA Batch Mesh Task Manager:

- 1: The pre-existing ANSA file with the constraints surfaces;
- 2: The envelope extracted;
- 3: The envelope and the constraints surfaces are reconnected, meshed with shell elements (but respectively with their own element length), and afterward fulfilled with volume elements.

The “batchmesh.c” script file has been adapted to execute in batch mode the above scenarios. At end of the meshing, an Abaqus job and a quality report are exported. Therefore, a new batch mesher component has been created and encapsulated by an optimization loop to minimize the number of the violating elements by handling the mesh element lengths of the surfaces.

In Figure 13-point 3, a data exchanger component rewrites two values of material parameters as the young modulus and Poisson’s ratio. The modified Abaqus job is then submitted to be calculated with an OS Command.  $\mu$ ETA gets back the .odb result file, and with a file session it extracts the Von Mises stress values into an ASCII file where Isight reads the response.

When the simple loop is finished, the DOE loop becomes easy to put in place.

## 4 Capture and automate morphing studies with Isight for an optimization loop

### 4.1 Morphing module and Parameters

The ANSA morphing module allows the production of alternative models based on a unique FE master model for automation, and thus for optimization. This tool is mainly based on morphing boxes surrounding the FE model. When changing the shape of these boxes, the enclosed elements follow accordingly. During the process, smooth or reconstruction functions can be also applied to preserve the quality and the integrity of the morphed model.

The shape of these boxes can be associated with morphing parameters—handled, for example, by control points for geometrical transformations, or by their edges or faces (for, respectively, lengths and angles measurements or offset measurements). In the same design, values available on ANSA cards can be controlled by the ANSA parameters (special entities able to reassign new values to ANSA cards). Hence, these different kinds of parameters can modify the shape of the model and FE cards. Accessible through a user-defined script or a session file, automation becomes achievable, but this way is not really adapted for an optimization loop, whereas the ANSA Optimization Task module perfectly fulfills this need.

### 4.2 Optimization Task module

Indeed, the Optimization Task (Figure 14), included in the ANSA Task Manager, is an efficient tool able to execute preprocessing tasks for an optimization loop. During the initialization of the sequence, Design variables can be defined and connected to Morphing parameters and ANSA Parameters entities, which are saved in an input DV file (Figure 15). User scripts or session functions (commands) can be also added to perform the morphed model, as well as a command to export the model in an FE format at end of the sequence.

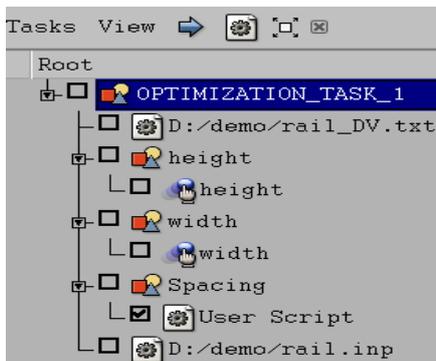


Figure 14: Optimization Task module

```

# Output from:
# d:/demo/rail_morph.ansa
#
# DESIGN VARIABLES
#-----#
# ID | DESIGN VARIABLE NAME | TYPE | RANGE | CURRENT VALUE | MIN VALUE --> MAX VALUE | STEP
#-----#
1, Emboss_Depth, REAL, BOUNDS, 0., -6., 0.
2, Emboss_Width, REAL, STEP, 0., -2., 2., 1.
3, Emboss_Dist, REAL, BOUNDS, 0., -5., 10.
4, Thickness, REAL, LIST, 1.2, 0.8,0.9,1.,1.2,1.4,1.7
#-----#
    
```

Annotations in the table:

- Min value**: points to the 'MIN VALUE' column.
- Max value**: points to the 'MAX VALUE' column.
- Step**: points to the 'STEP' column.
- Current values**: points to the 'CURRENT VALUE' column.
- Discrete values**: points to the 'LIST' range for the 'Thickness' variable.

Figure 15 : DV file

During the Execution mode, the Optimization module performs the following tasks:

- ✦ Replaces the current values of the Design Variables defined in the ANSA model with those contained in the DV file.
- ✦ Modifies the FE model accordingly.
- ✦ Executes every session or script function defined in the Optimization task.
- ✦ Exports the updated FE model.

4.3 Optimization loop coupling Isight with ANSA and μETA.

A typical design simulation process dealing with morphing studies with the coupling of Isight with ANSA Optimization Task and μETA is summarized in the Figure 14.

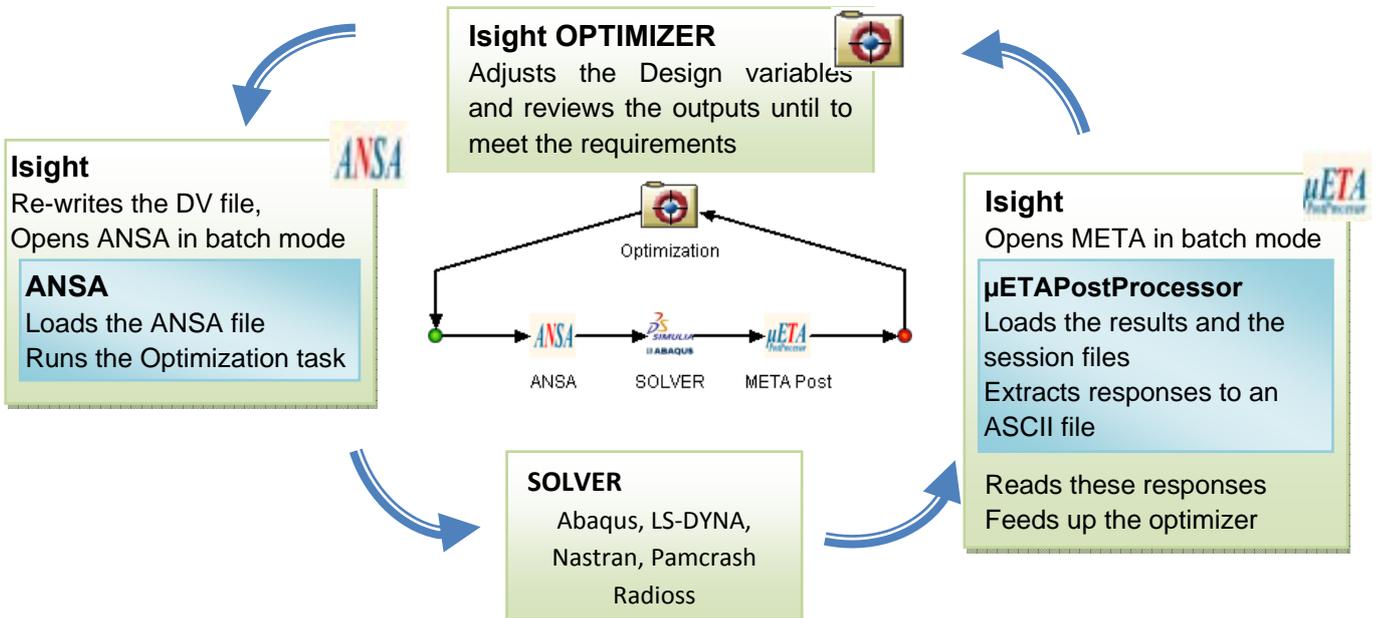


Figure 16: a typical design simulation process integrating ANSA and μETA within Isight driven here by its optimization loop component

Whatever the complexity of the morphed model or the analysis, the process integration and the way to drive the workflow remains totally identical. Therefore, the use cases provided in the ANSA tutorial were done easily since the morphing tools and Optimization task module became familiar. Indeed, the static load applied to a hook with Nastran, or the crash analysis of a front rail realized with LS-DYNA, have followed the same rules, except that we have used respectively the Abaqus implicit solver and the dynamic explicit solver (Figures 17 and 18).

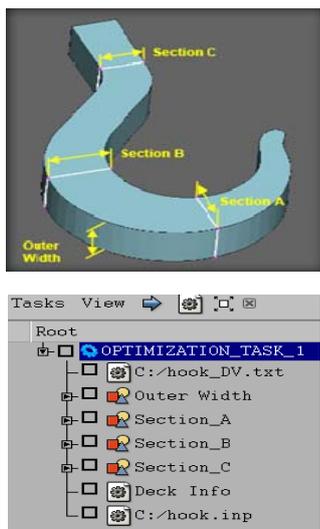


Figure 17: A hook in a static load with the Abaqus implicit solver

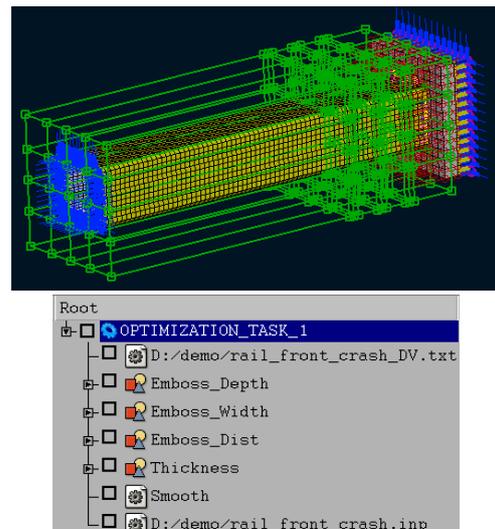


Figure 18: A rail arranged for a front crash with the Abaqus explicit solver

## 5 NEXT STEPS

The automation scenarios described above have illustrated several areas of improvements in meshing processes. The combinations of them will surely lead to better processes and productivity gains. However, in order to realize such benefit at the enterprise level, a proper deployment infrastructure is required. Isight is part of SIMULIA's vision for Simulation Lifecycle Management (SLM). SLM is designed specifically to manage simulation processes and data, to automate where possible routine and repetitive tasks, and to provide traceability between Products and Simulations. Under SIMULIA SLM's vision and development roadmap, we see the following as potential next steps:

- To couple the meshing process with the Product data through a dedicated Product Structure for Simulation which will provide the input information and store the optimized meshes in a meaningful structure for downstream processes;
- To expand the use of such process on a large number of parts and assemblies, and to develop monitoring facilities in an extended enterprise configuration;
- To improve the reuse of existing meshes (part and assemblies) for the morphing scenario (number 3) by providing tools to find the most suitable data from previous studies.

## 6 CONCLUSION

We have demonstrated in this paper:

- ✦ A fast and versatile approach to couple of the preprocessor ANSA and the postprocessor μETA within the Isight integration framework. The process does not require any scripting.
- ✦ An advanced CAD interface (component) in the context of a rapid CAD-to-CAE process.
- ✦ The resulting workflow enables multidiscipline design exploration studies.
- ✦ It is possible to develop a customized interface for in-house codes using the System Development Kit (SDK).
- ✦ Development of a generic workflow (problem-independent) that can be driven from an Excel spreadsheet is achievable.

## REFERENCES

- (1) ANSA version 12.1.5 User's Guide, BETA CAE Systems S.A., July 2008
  - (2) ANSA version 12.1.5 tutorials (6, 8 and 9), BETA CAE Systems S.A., July 2008
  - (3) μETA PostProcessor version 6.2.0. User's Guide, BETA CAE Systems S.A., June 2008
  - (4) Isight version 3.5.3 User's Guide, SIMULIA, April 2009
  - (5) Isight version 3.5.3 Component Guide, SIMULIA, April 2009
-

## **Acknowledgments**

The author greatly acknowledges particularly Dr. Mike Sheh and Olivier Hardy from SIMULIA, George Korbetis from BETA CAE S.A., and Xavier Dugros from Digital Product Simulation for their valuable contributions.

---