A STEP FORWARD IN PROCESS AUTOMATION: RESULTS MAPPING AND AUTOMATED MULTIPLE ABAQUS/STANDARD LOAD-CASES DEFINITION

Dipl.-Ing. M. Tryfonidis [1], MSc S. Vasudeva-Rao [2]

[1] BETA CAE Systems S.A., [2] Chrysler LLC.

Abstract

During the vehicle development process, the body structure undergoes many simulations for the assessment of its durability performance. The evaluation of the body structure for numerous scenarios involves both static and dynamic loads, leading to the generation of a large number of load cases. Following the demonstrated automated process, the results of previous multi-body simulations are mapped to the model in the form of Abaqus boundary conditions. This approach allows the fast identification of the worst case scenario of several loading maxima within the requested time intervals.

The presented model build-up method for these simulation scenarios, has the following characteristics:

- The modelling is performed quickly and easily, without forcing the analyst to use excessive interaction, scripting or text files editing
- The simulation model demonstrates high fidelity and quality, ensuring the confidence in Abaqus/Standard simulations
- Model updates and design changes are applied promptly, while maintaining the association between the updated model and the load case definition.

This paper presents the implementation of the above automated model build-up process, using ANSA Task Manager, in Chrysler LLC..

1. Introduction: Description of the load case

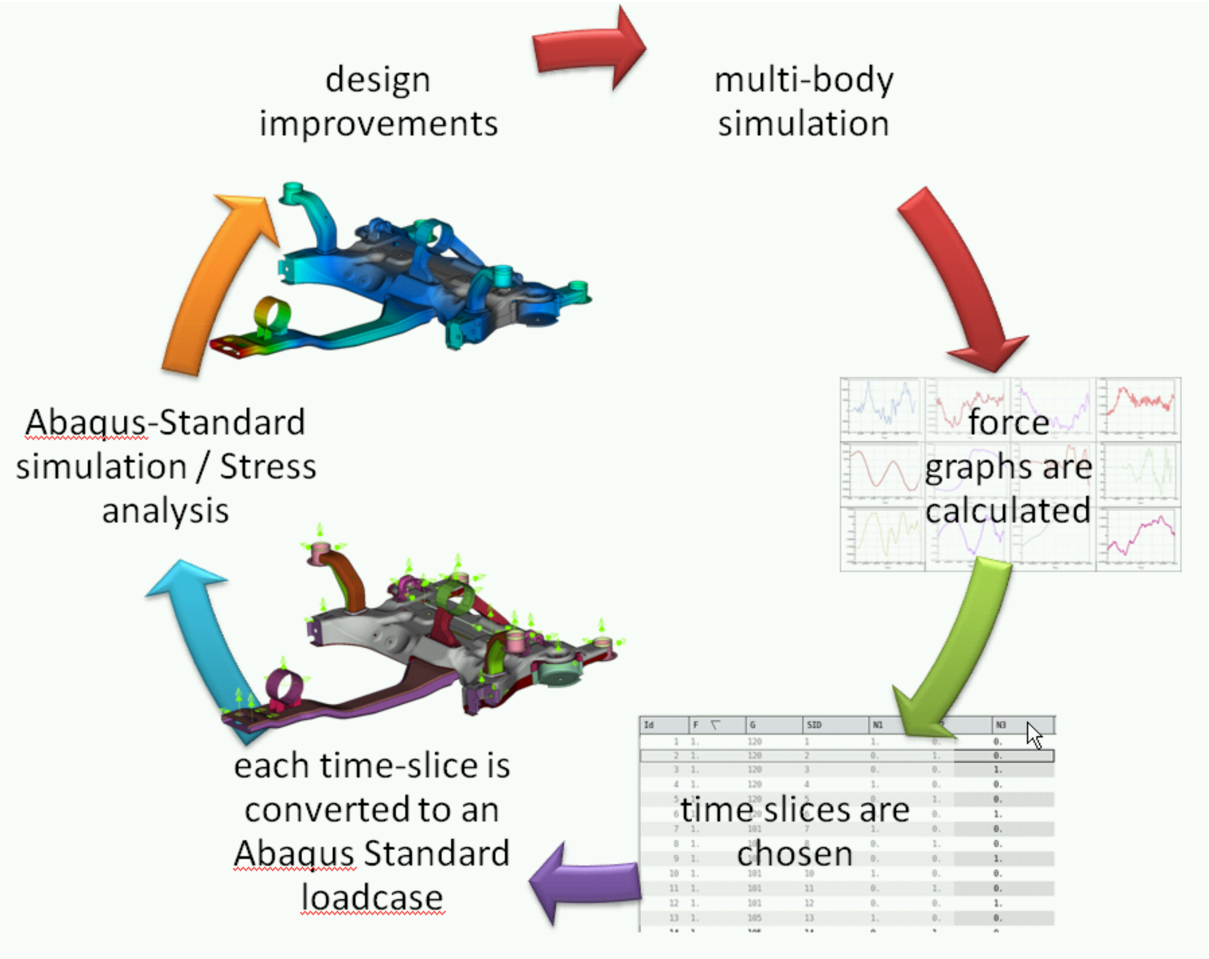
One discipline, in which the durability performance of the vehicle is assessed, is the application of loads calculated by a multi-body dynamics simulation (MBDS) and applied on the mounting locations at subassembly level. The MBDS delivers force graphs against time. Several time slices are chosen out of these graphs and their local maxima are combined to build load cases that will help to identify the worst case scenarios.

The important aspects of this procedure follows, which also serves a picture of the amount of data and the complexity of this procedure:

- 6 forces per degree of freedom calculated by the MBDS needs to be mapped to the FE-model in form of Abaqus *CLOAD‘s,
- these *CLOAD’s should be applied on the proper mounting locations, about 25 for the below pictured subassembly
- a *STEP definition plus output requests per time slice needs to be created as well. A usual case requires the application of a set of loads for some tens of time slices.

Once the Abaqus standard simulation is completed, the engineer might proceed to changes on the model design, based upon the simulation results, to improve the durability. To confirm the design changes, the analysis needs to re-run, for which some of the previous steps needs to be reapplied or, at least, checked if the model and the load case are still intact. Even more complicated is the case were a new design version of model’s components are released from the design team. These new components should replace the old ones, but the integrity of the load case needs to be assured. If the kinematic behaviour of the vehicle changes as well, then the MBDS simulations need to get updated. This means that the magnitudes of the forces and maybe even the Abaqus load cases themselves are not up to date any more.

The above process represents the work that needs to be done for one subassembly and one multi-body simulation load case (the process is also shown in Picture 1).


design improvements
multi-body simulation
force graphs are calculated
time slices are chosen
each time-slice is converted to an Abaqus Standard loadcase
Abaqus-Standard simulation / Stress analysis

Picture 1. Schematic representation of the process

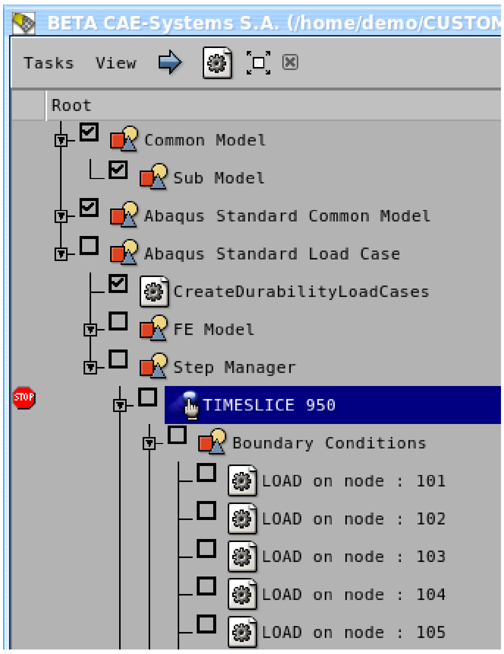
There is no question that, being able to automate this procedure with the lowest possible level of user interaction, leads to high time and cost savings. But it is the updating of the model with design changes and the model integrity checks that will upgrade this process into a more flexible, robust and controllable by the engineer level. Exactly this is the concept of the automation tools that were used and will be presented in the following chapters.

2. The tools in the core of the process automation workflow

For a better illustration of the process workflow, in the core of which stands the Task Manager and the Data Manager of the pre-processing software ANSA, some basic points of these tools follow.

2.1 Task Manager

The Task Manager (TM) is the automation tool that is responsible for the load case setup in this case. All necessary individual actions of the workflow are included in a step-wise manner. The accomplishment of the tasks is performed in the predefined order. In case of model changes and updates, the dependencies between tasks are checked as well, so that they are re-executed for the new model-run.



BETA CAE-Systems S.A. (/home/demo/CUSTOM

Tasks View

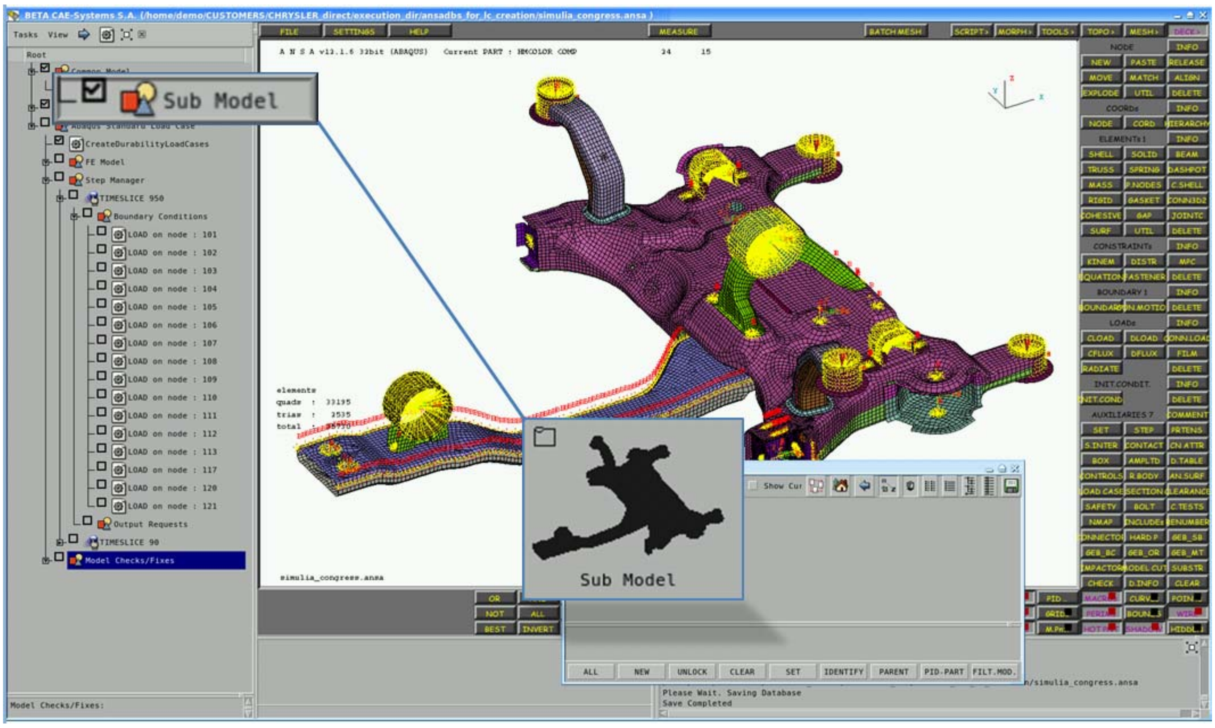
Root
- ☑ Common Model
  - ☑ Sub Model
- ☑ Abaqus Standard Common Model
- □ Abaqus Standard Load Case
  - ☑ CreateDurabilityLoadCases
  - □ FE Model
  - □ Step Manager
    - □ TIMESLICE 950
      - □ Boundary Conditions
        - □ LOAD on node : 101
        - □ LOAD on node : 102
        - □ LOAD on node : 103
        - □ LOAD on node : 104
        - □ LOAD on node : 105

Picture 2. A sample workflow as it is defined in the Task Manager. All activities are shown in a tree-like view. The model and load case setup is done step by step.

The important thing here is that the TM does not only know, which the necessary steps to be executed are, but also checks if they get realized properly. So, whatever activity can be automated is also executed automatically, unless something goes wrong. Then the TM will stop its execution and will highlight the step and the reason that forced him to stop. The engineer is then called to fix the problem, so that the TM can continue with the task executions from that point on.

For the Abaqus standard load cases, that need to be applied, the basic ingredients are the following task items:

- **Sub Model**: In order to link the TM with the proper geometric entities that build the geometry of the subassembly as an FE-model, a Sub Model item is created. This item is a direct link to a Group of parts in the Parts Manager.



Picture 3. The Sub model task item associates the TM with the FE-model of the subassembly

- **Load mapping item:** a user defined function written in ANSA Scripting language, reads the mapping-files and creates the necessary steps in the Task tree by interacting with the ANSA Data Management tool.
- **Step Manager**: this item contains the parameters for the simulation calculation, as well as all the load case definitions
- **Step:** each Step definition represents the Abaqus *STEP keyword. It also stands for the load case of each time slice
- **Boundary condition:** each boundary condition stands for the creation of the Abaqus *CLOAD keywords for the particular Step
- **Output requests** for post processing the simulation results
- **Model integrity checks:** these need to be defined, in order to assure that the FE-model will run with Abaqus without errors
- **Model Export:** finally, the complete Abaqus deck needs to be exported

Some further explanations are given to some of the items mentioned above in the chapters that will follow.

2.2 Data manager

ANSA Data Management (ANSA DM) is a centralized data management system, used to collect and store in a structured and hierarchical form all engineering data that are used during the development process of a vehicle simulation model.

Under ANSA DM, the engineering data are stored in the same physical location. This location is made known to ANSA, so that all communication related to the storage and retrieval of engineering data and their updates is managed automatically. Such data can be:

- **Part and subassembly data**: The geometric representation of parts and subassemblies in ANSA databases
- **Analysis models**: The discipline dependant FE-representations of all parts and subassemblies comprising a simulation model (NVH models, crash analysis models, durability analysis models, etc.)
- **Sub-systems**: Seats and dummies, walls and barriers; auxiliary components necessary for the analysis
- **Task trees**: The procedures to be followed for the build-up of each discipline model
- **Library Items**: Connector entities, boundary conditions, output requests and trim items
- **Post processing data:** Result files, session files, report templates, analysis reports
- **Test results**: Videos, accelerometers

The most important feature of the ANSA DM used during the examined workflow is the storage and usage of library items. In this process libraries for the Task Trees as well as for template boundary condition items were frequently used. Their role will be explained in further detail in the chapters to follow.

3. The load mapping process

The input data for the load mapping process are two ASCII files:

a) **map file**: this one contains the type of the boundary condition (FORCE or MOMENT), the degree of freedom (x, y or z) and the node id, were the force needs to be applied.

b) **time slice file**: this contains the magnitudes of the calculated loadings and that need to be mapped to the sub assembly, for each of the included time slices.

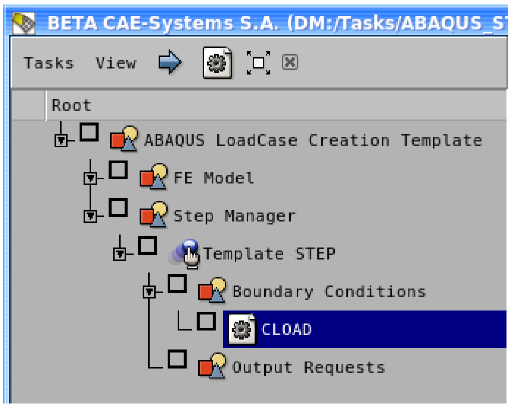
For the purposes of this process, a user defined function was created using the ANSA scripting language feature. It is a C-like language, which besides the basic language features (such as loops, if-statements, maps, structures and others) offers access to the ANSA core functionality, through a library of functions that covers almost the complete software.

The script function acts as a reader: to start with its execution, the user needs to select the map file and the time slice file. Based on the information that was read:

1. One task-tree template per times lice is read from the ANSA DM. These tasks are merged to the existing one and each of them contain a Step item, the parameters for the Abaqus standard simulation, the output requests and a template GEB_BC item for application of the loads.

2. The GEB_BC item is repeatedly created, until there exists one item per mounting location.

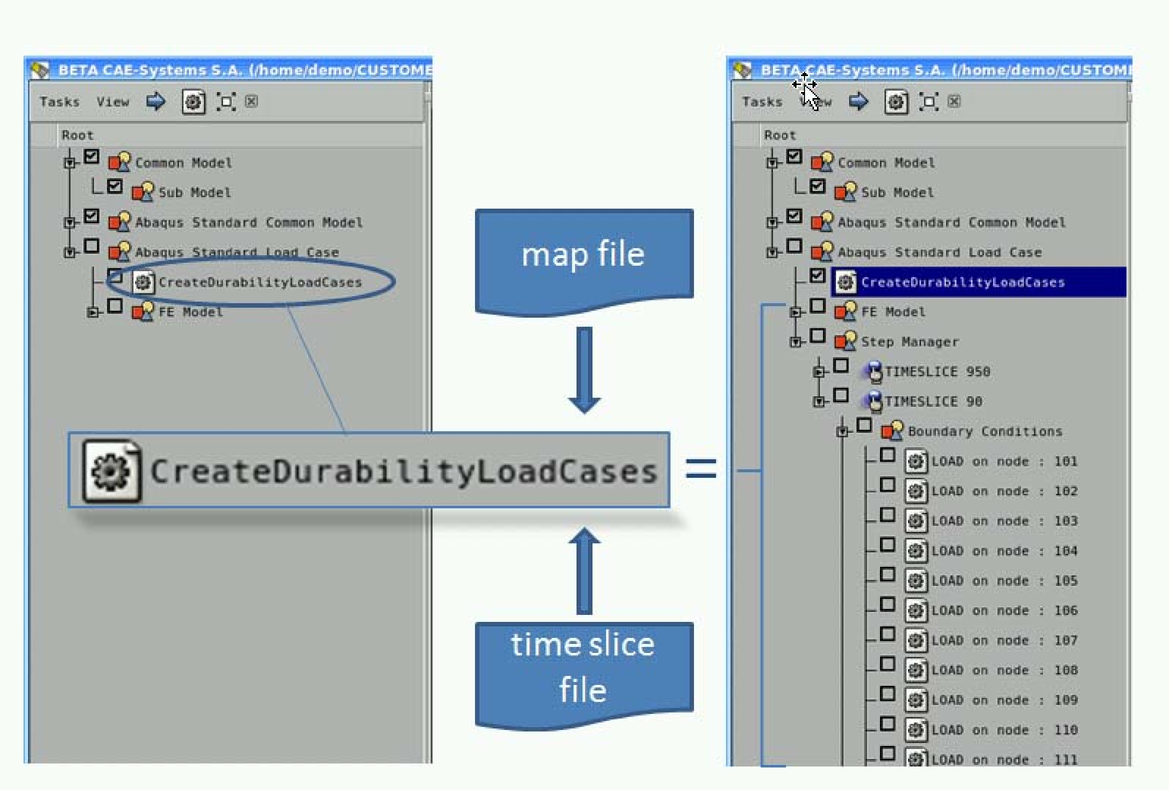
3. The magnitudes read from the time slice file in addition with the degrees of freedoms identified in the map file, are mapped inside the GEB_BC item in form of parameters. These parameters will be used on a later step, in conjunction with the CLOAD template file, to apply the proper magnitudes to each of the load.



```
BETA CAE-Systems S.A. (DM:/Tasks/ABAQUS_S
Tasks  View
Root
□ ABAQUS LoadCase Creation Template
  □ FE Model
  □ Step Manager
    □ Template STEP
      □ Boundary Conditions
        □ CLOAD
      □ Output Requests
```

Picture 4. The Abaqus Standard load case creation template. This task tree is stored in the ANSA DM and is repeatedly merged in the model, for each of the time slices.

To say it in simple words: a user script item, part of the TM itself, generates load cases by merging task templates from the ANSA DM, always according to the input data. Of course, the exit codes of the scripts are indicating to the TM its proper or improper execution.



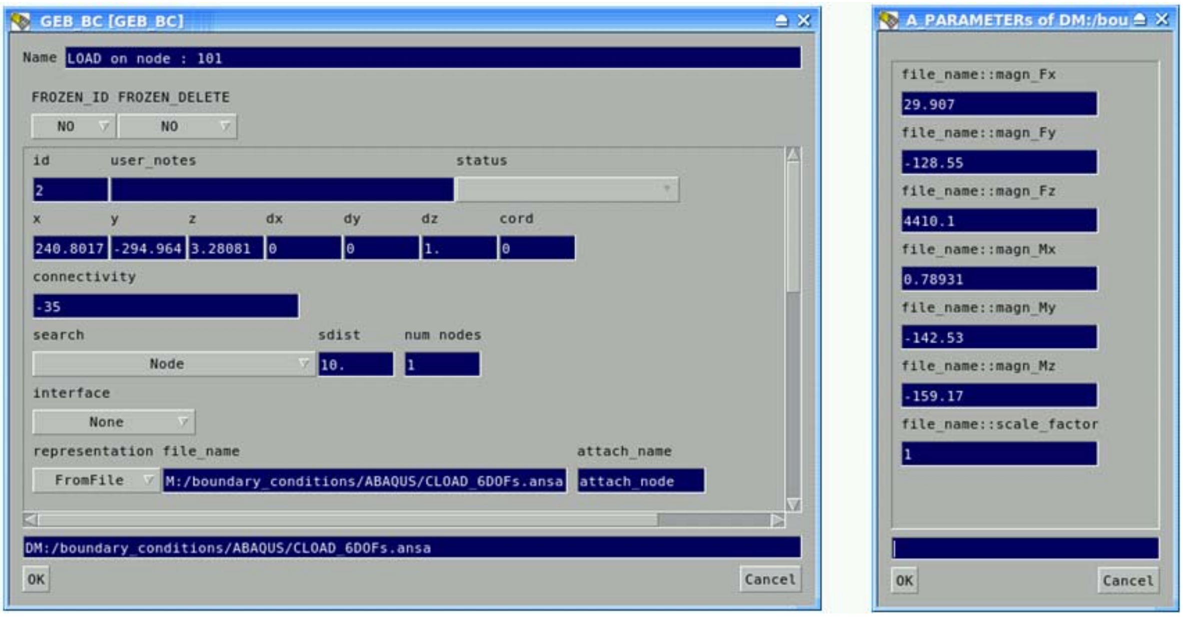
```
BETA CAE-Systems S.A. (/home/demo/CUSTOME
Tasks  View
Root
☑ Common Model
  ☑ Sub Model
☑ Abaqus Standard Common Model
□ Abaqus Standard Load Case
  ☑ CreateDurabilityLoadCases
  □ FE Model
```

```
BETA CAE-Systems S.A. (/home/demo/CUSTOME
Tasks  View
Root
☑ Common Model
  ☑ Sub Model
☑ Abaqus Standard Common Model
□ Abaqus Standard Load Case
  ☑ CreateDurabilityLoadCases
  □ FE Model
  □ Step Manager
    □ TIMESLICE 950
    □ TIMESLICE 90
      □ Boundary Conditions
        □ LOAD on node : 101
        □ LOAD on node : 102
        □ LOAD on node : 103
        □ LOAD on node : 104
        □ LOAD on node : 105
        □ LOAD on node : 106
        □ LOAD on node : 107
        □ LOAD on node : 108
        □ LOAD on node : 109
        □ LOAD on node : 110
        □ LOAD on node : 111
```

Picture 5. The load case creation step. The user script function, based on the information served by the map and time slice files, creates the load case definitions.

4. The boundary condition items

The boundary condition (GEB_BC) items are ANSA generic entities that create solver dependent FE-representations. Their definition knows what kind of a boundary condition needs to be applied on which location and the parts to attach on. Different kind of search patterns can be specified, in order for example to attach a boundary condition to a node, to holes or to a certain amount of shell elements.



Name: LOAD on node : 101
FROZEN_ID: NO
FROZEN_DELETE: NO
id: 2
user_notes:
status:
x: 240.8017
y: -294.964
z: 3.28081
dx: 0
dy: 0
dz: 1.
cord: 0
connectivity: -35
search: Node
sdist: 10.
num nodes: 1
interface: None
representation: FromFile
file_name: M:/boundary_conditions/ABAQUS/CLOAD_6DOFs.ansa
attach_name: attach_node
DM:/boundary_conditions/ABAQUS/CLOAD_6DOFs.ansa

file_name::magn_Fx: 29.907
file_name::magn_Fy: -128.55
file_name::magn_Fz: 4410.1
file_name::magn_Mx: 0.78931
file_name::magn_My: -142.53
file_name::magn_Mz: -159.17
file_name::scale_factor: 1

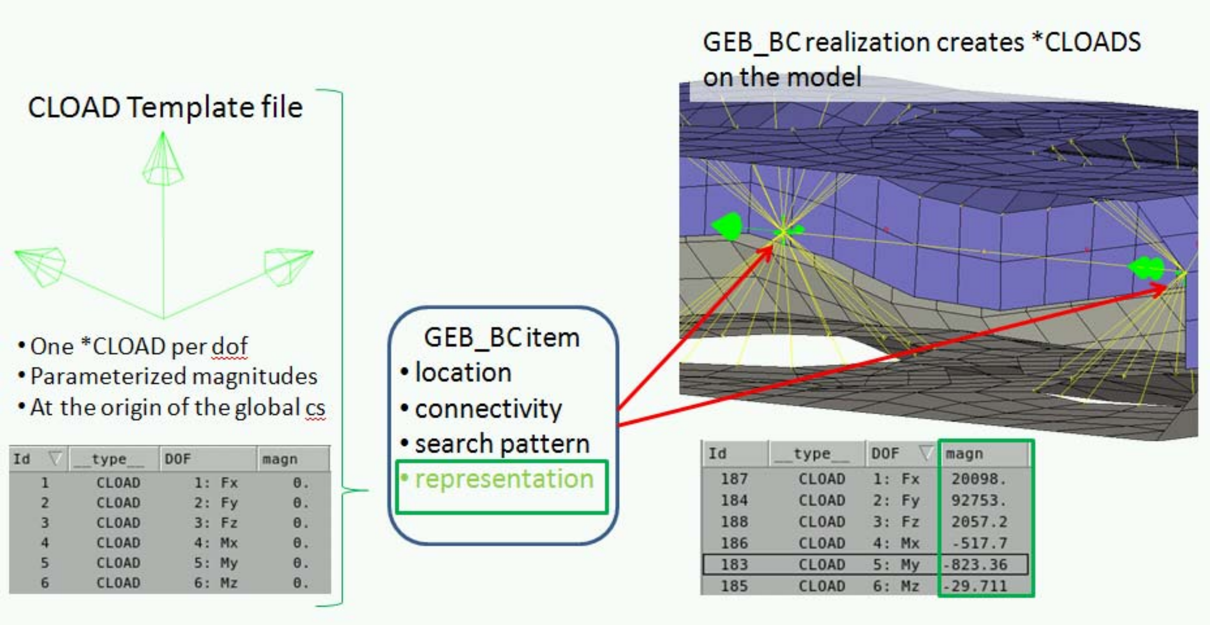
Picture 6. Boundary condition item (GEB_BC) with its complete definition

For the purposes of the examined case, the GEB_BC items used in the template have the following characteristics:

- the location of the boundary condition is read-in from the map file
- the sub assembly of interest is referenced as the connectivity
- the search pattern used for the boundary condition application is always the closest node of the sub assembly to the location of the GEB_BC item
- the CLOADS are directly attached to the structure, with no interfacing
- the FE-representation, that is applied during the GEB_BC realization, is a library item stored in the ANSA DM

So, the only things that are different for each GEB_BC are their location and the magnitudes of the forces. During the realization phase of the item, the library file is merged into the main model and is getting attached at the closest node that was found. At the same time, the magnitudes for each degree of freedom are applied as well.

During the creation of the GEB_BC's, the node id on which the loads are applied is set in to a numbering rule. This means that even if that particular component gets replaced by an updated one, the nodes were the forces get applied will still have the same id. This is mainly done for organizational as well as for post processing reasons.



CLOAD Template file

- One *CLOAD per dof
- Parameterized magnitudes
- At the origin of the global cs

Id	__type__	DOF	magn
1	CLOAD	1: Fx	0.
2	CLOAD	2: Fy	0.
3	CLOAD	3: Fz	0.
4	CLOAD	4: Mx	0.
5	CLOAD	5: My	0.
6	CLOAD	6: Mz	0.

GEB_BC item

- location
- connectivity
- search pattern
- representation

GEB_BC realization creates *CLOADS on the model

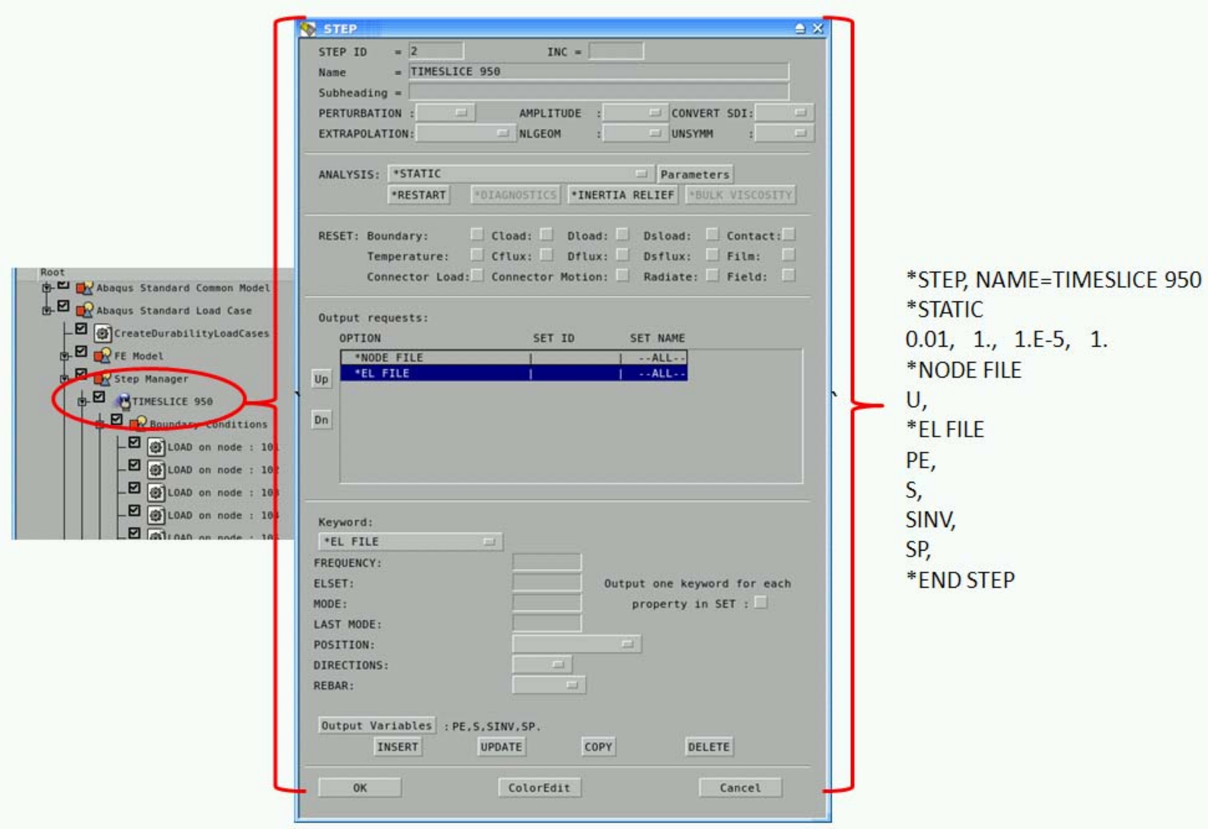
Id	__type__	DOF	magn
187	CLOAD	1: Fx	20098.
184	CLOAD	2: Fy	92753.
188	CLOAD	3: Fz	2057.2
186	CLOAD	4: Mx	-517.7
183	CLOAD	5: My	-823.36
185	CLOAD	6: Mz	-29.711

Picture 7. The concept of the GEB_BC: the template file contains the *CLOADS, one for each dof. During the realization of the GEB_BC, the loads are attached to the structure were the items are located and the magnitudes are applied at the same time.

5. The Step definition

The task tree template that contains the step definition, has already predefined the basic parameters that are needed for that specific kind of analysis. That is:

- the type of the analysis (*STATIC in this case), as well as all the necessary parameters (such as time increment settings)
- name and subheading, according to the time slice
- output requests for the post processing of the analysis results



STEP ID = 2 INC =
Name = TIMESLICE 950
Subheading =
PERTURBATION : AMPLITUDE : CONVERT SDI:
EXTRAPOLATION: NLGEOM : UNSYMM :
ANALYSIS: *STATIC

RESET: Boundary: □ Cload: □ Dload: □ Dsload: □ Contact: □
Temperature: □ Cflux: □ Dflux: □ Dsflux: □ Film: □
Connector Load: □ Connector Motion: □ Radiate: □ Field: □

Output requests:

OPTION	SET ID	SET NAME
*NODE FILE		--ALL--
*EL FILE		--ALL--

Keyword: *EL FILE
FREQUENCY:
ELSET:
MODE:
LAST MODE:
POSITION:
DIRECTIONS:
REBAR:
Output one keyword for each property in SET : □

Output Variables : PE,S,SINV,SP.

```
*STEP, NAME=TIMESLICE 950
*STATIC
0.01,   1.,   1.E-5,   1.
*NODE FILE
U,
*EL FILE
PE,
S,
SINV,
SP,
*END STEP
```

Picture 8. The Step definition as part of the task tree. All parameters as well as output requests are included.

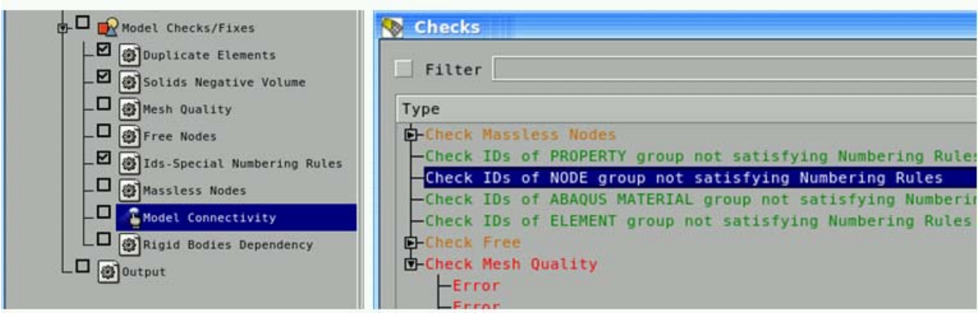
6. Model integrity checks

Before continuing with the deck export into the Abaqus format, it is necessary to check the validity of the FE-model definition, according to the solver's guidelines. So, the last sort of items that are added, this time by the user, are the model checks and fixes, by reading the respective task tree template from the ANSA DM.

Some of these checks run with predefined parameters, in order to identify problematic keyword definitions, such as rigid bodies' dependencies or negative volume elements. But an error is not only something that would prevent Abaqus from concluding the solving phase. Errors are also id numbers that do not follow the required numbering rules, or the existence of duplicated elements of any type has no meaning at all in this (but of course accepted by Abaqus).

Also, since the work is done at sub assembly level, there should be everything connected. No loose entities should exist in the model. This connectivity check is done with user interaction. During its execution, the user is requested to pick an element, from which the search algorithm will start looking for all connected entities to this one. Whatever is not connected is prompted to the user, in order for him to understand what went wrong with the model. This check is of more importance after part replacement activities took place.

Finally, if all the checks are performed a list appears containing all possible errors plus description. The entries of this list are associated with the model, so that visibility and identification controlling is given to the user. For some the checks, auto- and semiautomatic fixing algorithms are available as well.



```
☐ Model Checks/Fixes
  ☑ Duplicate Elements
  ☑ Solids Negative Volume
  ☐ Mesh Quality
  ☐ Free Nodes
  ☑ Ids-Special Numbering Rules
  ☐ Massless Nodes
  ☐ Model Connectivity
  ☐ Rigid Bodies Dependency
☐ Output
```

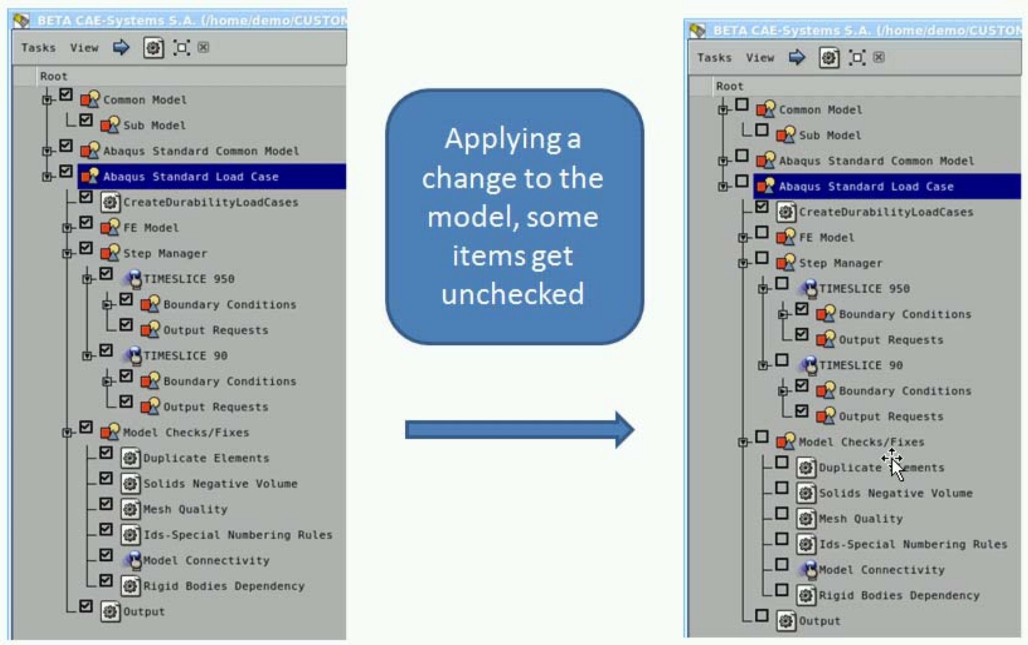
```
Checks
☐ Filter
Type
Check Massless Nodes
Check IDs of PROPERTY group not satisfying Numbering Rule
Check IDs of NODE group not satisfying Numbering Rules
Check IDs of ABAQUS MATERIAL group not satisfying Numberi
Check IDs of ELEMENT group not satisfying Numbering Rules
Check Free
Check Mesh Quality
  Error
  Error
```

Picture 9. The model integrity checks are performed prior to the deck export.

7. Updating the model with new design versions

In case, new versions of the components of the model are released, containing design improvements, the analysis needs to run again. The replacement activity is done by the user at part or sub assembly level, using the Part Replace functionality. In order to keep the association with the sub assembly as well as with the load cases, this functionality tries to identify the activities that need to be reapplied (connections, mass trimmings, set definitions, boundary conditions etc.), in order to satisfy both, regardless if their creation was done by task items or not.

Right after the part replacement action, the TM identified the actions that need to be re-executed or, at least, checked if they are still intact.



```
BETA CAE-Systems S.A. (/home/demo/CUSTOM
Tasks  View
Root
☑ Common Model
  ☑ Sub Model
☑ Abaqus Standard Common Model
☑ Abaqus Standard Load Case
  ☑ CreateDurabilityLoadCases
  ☑ FE Model
  ☑ Step Manager
    ☑ TIMESLICE 950
      ☑ Boundary Conditions
      ☑ Output Requests
    ☑ TIMESLICE 90
      ☑ Boundary Conditions
      ☑ Output Requests
  ☑ Model Checks/Fixes
    ☑ Duplicate Elements
    ☑ Solids Negative Volume
    ☑ Mesh Quality
    ☑ Ids-Special Numbering Rules
    ☑ Model Connectivity
    ☑ Rigid Bodies Dependency
  ☑ Output
```

Applying a change to the model, some items get unchecked

```
BETA CAE-Systems S.A. (/home/demo/CUSTOM
Tasks  View
Root
☐ Common Model
  ☐ Sub Model
☐ Abaqus Standard Common Model
☐ Abaqus Standard Load Case
  ☑ CreateDurabilityLoadCases
  ☐ FE Model
  ☐ Step Manager
    ☐ TIMESLICE 950
      ☑ Boundary Conditions
      ☑ Output Requests
    ☐ TIMESLICE 90
      ☑ Boundary Conditions
      ☑ Output Requests
  ☐ Model Checks/Fixes
    ☐ Duplicate Elements
    ☐ Solids Negative Volume
    ☐ Mesh Quality
    ☐ Ids-Special Numbering Rules
    ☐ Model Connectivity
    ☐ Rigid Bodies Dependency
  ☐ Output
```

Picture 10. The TM checks the dependencies of activities and requires re-execution of the involved steps.

In order to be sure for the integrity of the model, all checks need to re-run after a part replacement activity. This way, if the part information is improper or something went wrong during the replacement activity, during the “model checks and fixes”-stage of the task execution, all such possible errors are traced, depicted and served on a list in one shot for the complete deck.

8. Conclusions

The data organization and process automation necessary for the creation of Abaqus standard load cases can be achieved with the usage of the Task Manager as the core tool. An analytical description of the load mapping process and the load case preparation has been presented. The input data, loadings calculated from a previous simulation are mapped at sub assembly level. This is accomplished by a user script function that reads in the input files that contain the mapping information, as well as the force magnitudes that has to be applied to the mounting location of the model.

In close cooperation with library files stored in the ANSA Data Management pool, task trees, boundary conditions and load case definition in form of Abaqus *STEP are created and applied in an automatic way. The robustness is safeguarded through the usage of the model integrity checks.

An integrated process workflow is presented that covers even the updating of the model with new versions of the components of the model, which contain design variations. The identification of the worst case scenarios can be done this way far quicker. The engineer now has the freedom to choose as many time slices needed for the respective simulation scenario and be more flexible and effective in improving the durability performance of the body structures he is in charge.

References

1. “Abaqus v6.8 Keywords Reference Manual”, Dassault Systemes Simulia Corp., Providence, RI, USA 2008
2. “Abaqus v6.8 Analysis User’s Manual”, Dassault Systemes Simulia Corp., Providence, RI, USA 2008
3. “ANSA v12.1.5 User’s Guide”, BETA CAE Systems S.A., Thessaloniki, Greece 2008
4. “CAE Data and Process Management with ANSA”, BETA CAE Systems S.A., Thessaloniki, Greece 2009
5. “Scripting Language: A guide to ANSA automation”, BETA CAE Systems S.A., Thessaloniki 2008