INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN
ICED 05 MELBOURNE, AUGUST 15-18, 2005

DOCUMENTATION AND ENHANCEMENT OF THE TRACEABILITY OF FINITE-ELEMENT-ANALYSIS

Rüdiger Maus, Christian Weber

Keywords: finite element analysis, documentation of simulations, traceability

1 Introduction

The approach presented in this paper is based on the CAE-environment of the product development process. Simulations and results often are safety-relevant and thus have to be traceable even years after the engineering process is completed, especially if structural stiffness, durability and crash behaviour are concerned. At present, the documentation mostly focuses on the results. Since these are influenced by many more parameters than just the geometry to simulate, the way the simulation model is defined should be documented as well.

The concept intends to enhance the quality of FEA-simulation documentation and to make a comparison of simulation models and results possible. Additionally, it allows the reliable reproduction of simulations. The core components are a database for the storage of simulation models and a so-called processing tool, which reads in new simulation files into and generates the documentation files out of the database. Besides that, another database supports the engineer in building simulation models for new cases.

2 Initial situation

The modern product development process is located in a more and more virtual environment. This surrounding consists of a multitude of Computer Aided systems (CAx-systems) supporting one or more phases of the product development process. The most popular representatives are Computer Aided Design (CAD), Production Planning (CAPP), Engineering (CAE) and Digital Mock-Up (DMU). The first system in the process is CAD, which supports the designer to build up virtual three-dimensional models of the parts and assemblies for the future product. Currently, these models are more and more replacing two-dimensional engineering drawings and thus becoming the so-called “digital master” throughout the engineering process. The following tools mostly base on this “digital master”, which means that they use or re-use the geometry and other information created by the designer.

When the designer changes this “digital master”, the following processes such as CAE and CAPP have to be notified about these modifications in design, material specification or whatever characteristic of the product, to update their own digital models. The information about these versions and variants is mostly accomplished by workflow management and PDM-systems. Since these systems to a large extend do not store the reasons for changes made, it would be very helpful to document this knowledge at least in-phase.

Especially in the FEA-environment, there exists a large variety of tools for analysis and simulation of structures, components and assemblies. These tools help the engineer to ensure that the product will be able to endure the mechanical, thermal, electrical, electromagnetic and all kinds of coupled loads, which it will be exposed to during its lifecycle. Nowadays, the documentation in this area mostly consists of screenshots and numerical values such as maximum stresses and deformations [2]. However, this result is just the output of the simulation model, which itself is built by the engineer who has a large variety of options in doing so. These parameters influence the outcome of a simulation to a large extend, and are essential for the interpretation of results [1].

2.1 Example for influencing parameters in FEA

In the classic FEA-environment, the set up of a simulation model usually starts with the import of the part's geometry from CAD. After that, the geometry is "meshed" to a finite element model consisting of nodes and elements such as – in a three-dimensional simulation – tetrahedrons or hexahedrons. The following list contains only the most important parameters that have to be defined before a simulation can start:

- mesh parameters (e.g. geometry and size of the finite elements, accuracy)
- element types (linear/parabolic interpolation)
- material properties (Young's modulus, Poisson's ratio, density)
- definition of boundary conditions (fixed supports, loads)
- number of steps of the analysis (load steps, increments)

Although none of these parameters is directly coupled to the geometry, the product characteristics or the boundary conditions to consider, they have a great influence on the simulation result. As there is no optimal combination of these parameters for a simulation, the engineer often modifies one or more of them during his work before he/she considers the result trustworthy. Examples for changes are the refinement of the whole finite element mesh or of some particularly interesting areas of the structure, the use of other element types that fit better to specific load conditions (e.g. bending, shearing), or the replacement of a boundary condition by an alternative assumption (e.g. face load by point load(s) or vice-versa). Furthermore, the same geometry/structure is often simulated with different combinations of loads that are predicted for the lifecycle of the product.

The following exemplary case shows the influence of geometry- and load-independent parameters on the simulation result. Therefore, a simple, analytically traceable problem is used. It consists of a straight beam with a fixed support at one end and a bending load at the other. Figure 1 shows a sketch of the problem, the following equations (1) and (2) contain the analytical results. The parameters given are:

- $L = 500mm; W = H = 25mm; A = W \cdot H = 625mm^2$ (length, width, height, area)
- $E = 210{,}000\,N/mm^2; \nu = 0.3$ (Young's modulus, Poisson's ratio)
- $I_y = \frac{W \cdot H^3}{12} = \frac{25mm \cdot (25mm)^3}{12} = 32552.08mm^4$ (geometrical moment of inertia, y-axis)
- $F = 5{,}000N$ (vertical force)

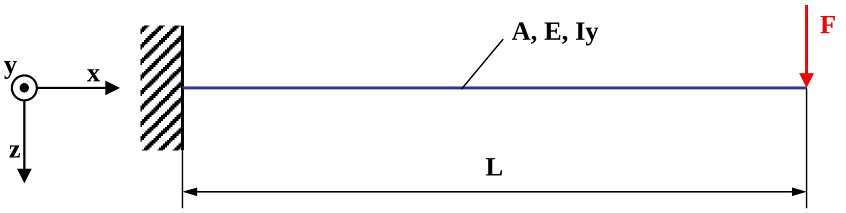


Figure 1. Sketch of the example problem

The results of the analytical calculation are [3]:

Maximum deflection $$d = \frac{F \cdot L^3}{3 \cdot E \cdot I_y} = \frac{5{,}000N \cdot (250mm)^3}{3 \cdot 210{,}000 \,{}^{N}\!/\!{}_{mm^2} \cdot 32{,}552.08mm^4} = 3.81mm \qquad (1)$$

Maximum stress $$\sigma = \frac{F \cdot L \cdot {}^{H}\!/\!{}_{2}}{I_y} = \frac{5{,}000N \cdot 250mm \cdot 12.5mm}{32{,}552.08mm^4} = 480\frac{N}{mm^2} \qquad (2)$$

The FEA-system used for the comparison is MSC.MARC/Mentat from MSC.Software, one of the models simulated with FEA is depicted in Figure 2.

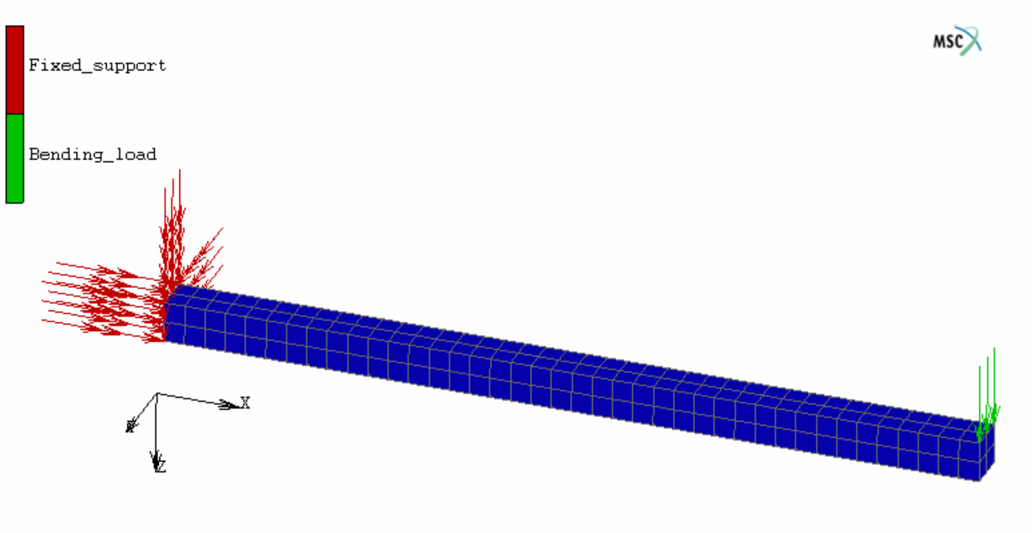


Figure 2. Model of the beam with boundary conditions in MSC.Mentat

The parameters changed in the example are:

- number of elements/nodes
- problem dimension (one-, two-, three-dimensional)
- element geometry (2D: triangles, quadrilaterals; 3D: tetrahedrons, hexahedrons)
- element types
- additional special options often used in simulations

These parameters were modified one by one, and about 200 different models were simulated. The outcome of this study was that when the number of elements becomes high enough, all configurations converge to a value for the maximum deflection at the free end of the beam very close to the analytical result of 3.81 mm.

Some exemplary results, which clarify the differences of the individual simulations, are shown in Table 1.

Table 1. Results of simulations with modified parameters

Line	**Dimension**	**Elements**	**Nodes**	**El. geometry**[*1]	**Element type**[*2]	**Integration type**	**Deflect. [mm]**	**Deviat. [%]**
1	analytic.						3.81	0
2	1	1	3	line	parabolic	-/-	3.81	0
3	2	41,665	81,920	tria3	linear	full	3.49	8
4	2	640	731	quad4		full	3.45	9
5	2	40,960	41,667	quad4		full	3.49	8
6	3	10	44	hex8	linear	full	2.47	35
7	3	10	44	hex8	linear	reduced	3.79	1
8	3	10	44	hex8	linear	full	3.80	0
9	3	5,120	6,561	hex8	linear	full	3.78	1

Legend:
[*1]: lines (line), triangles (tria), quadrilaterals (quad), hexahedrons (hex), number of nodes
[*2]: linear/parabolic interpolation

Some configurations – especially the one-dimensional example (line 2) – calculate better results with a very small number of elements than other configurations (e.g. 2-dimensional, 3-point-tedrahedrons, line 3) with more than 40,000 elements. Another point is that although the number of elements/nodes (identical mesh) and the element type used are the same, the results vary about more than 50% depending from one single option chosen (full/reduced integration, lines 6 and 7).

This example shows the necessity to store the parameters defined in an analysis to be able to trace and verify the results.

3 Concept description

As the engineer concerned with the analysis should not additionally be loaded with the documentation, the approach presented in this paper shows a possibility for a mostly automated documentation of both the simulated models and some of the results derived. Additionally, the concept supports the engineer to determine the right parameters for the simulation he/she is about to define. Since the complexity of these parameter settings in the FEA environment is very high, it might be useful to not only save data from simulations processed, but also save information about which settings fit best to specific problems. This can, for example, be information about different element types, the definition of boundary conditions, material properties and additionally best-practice methodologies that help engineers without much experience in specific simulation areas to define “good” simulation models more quickly.

The overall concept for the documentation tool is depicted in Figure 3.

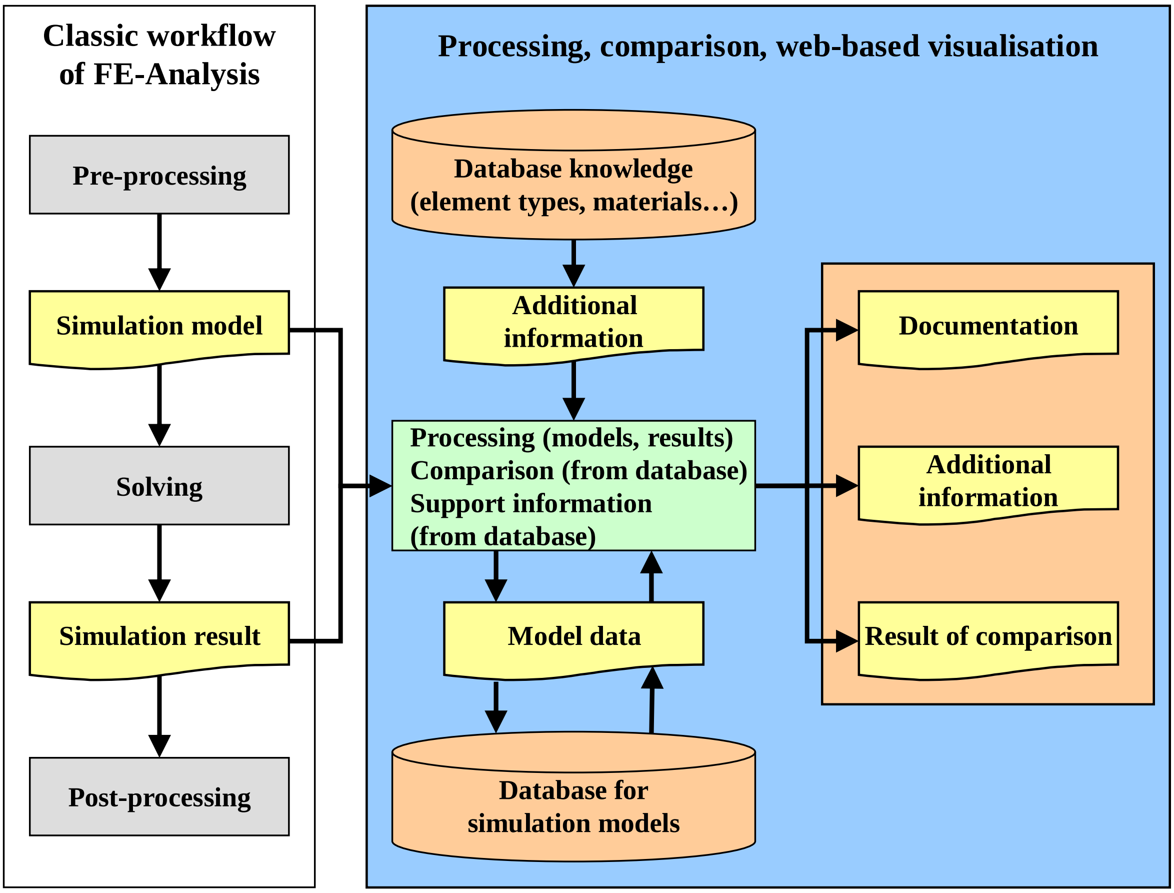

Classic workflow of FE-Analysis
Pre-processing
Simulation model
Solving
Simulation result
Post-processing
Processing, comparison, web-based visualisation
Database knowledge (element types, materials…)
Additional information
Processing (models, results)
Comparison (from database)
Support information (from database)
Model data
Database for simulation models
Documentation
Additional information
Result of comparison

Figure 3. Concept of the documentation tool

The following subsections explain the major modules of the concept, which are the database for simulation models, the additional database for "knowledge" and the processing tool used to store models in this database for simulation models.

3.1 Database for simulation models

The information about the model files and simulation results are stored in this database. As it is the heart of the documentation concept, its specifications have to be defined at first. The most important ones are listed below:

- clearly defined structure for the storing of model information
- independency from FEA-system
- independency from computer platform and operating system
- accessibility without having FEA-software installed
- avoidance of redundancies as far as possible
- automated generation of documentation files
- feasibility to compare simulation files

The database structure defined in this concept is shown in Figure 4.

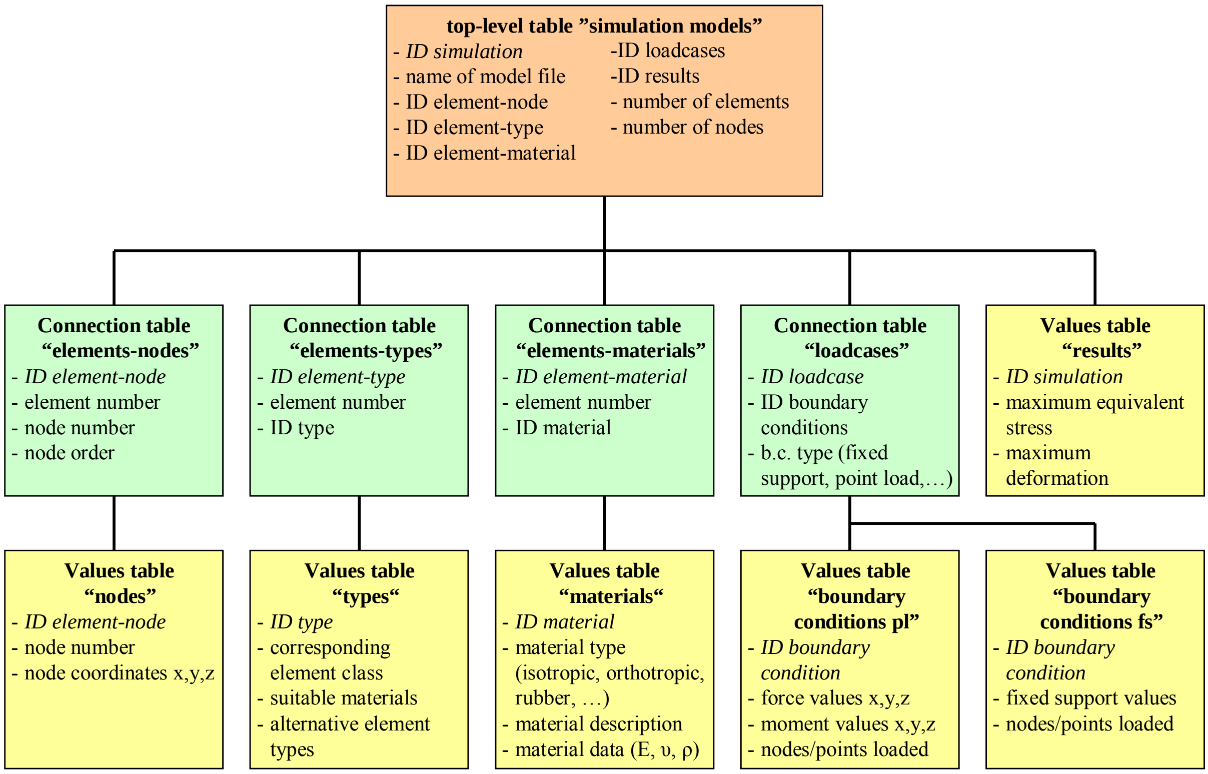


Figure 4. Database structure

In the structure of the database, there exist three classes of tables. They are described in the following paragraphs.

Top-level table "simulation models":

This table with a dataset for each simulation contains a unique identifier for the simulation, the name of the simulation model file, several identifiers for the connection tables and the overall number of elements and nodes of the simulation.

Connection tables:

In these tables, detailed information about the simulation model is stored. Exemplary, the "Connection table elements-nodes" consists of the individual ID linked in the database above, the ID of the elements the IDs of the nodes defining this element, and the order of the nodes, but not the coordinates of the nodes themselves.

Values tables:

In this detail level of the database, detailed information about node coordinates, material properties, element types, boundary conditions (fixed supports, forces, temperatures…) and some picked-out results (maximum of the equivalent stress, maximum deformation…) is stored.

The advantages of this modularisation are at first the avoidance of redundancies, e.g. when the same mesh definition is used with several element types, material properties and/or boundary conditions.

3.2 “Knowledge” database

This part of the concept is designed to support the engineer in choosing the right parameters for the specific problem he has to simulate. The main part of the database consists of information about:

- element types
- material properties
- boundary conditions

For each entry, besides the information given in manuals, the engineer can find hints about the right usage, e.g. the loadings and materials an element type suits for and which other element types are available for the same element geometry and/or material.

3.3 Processing tool

The FEA system used to present the concept for the storing of model information in the database described before is MSC.MARC/Mentat from MSC.Software. After the model is defined in Mentat, the start of the simulation initiates the generation of a so-called “MARC input deck”-file (extension .dat) which can also be delivered from other pre-processor programs. Figure 5 shows the information flow between the pre- and postprocessor Mentat and the solver MARC.

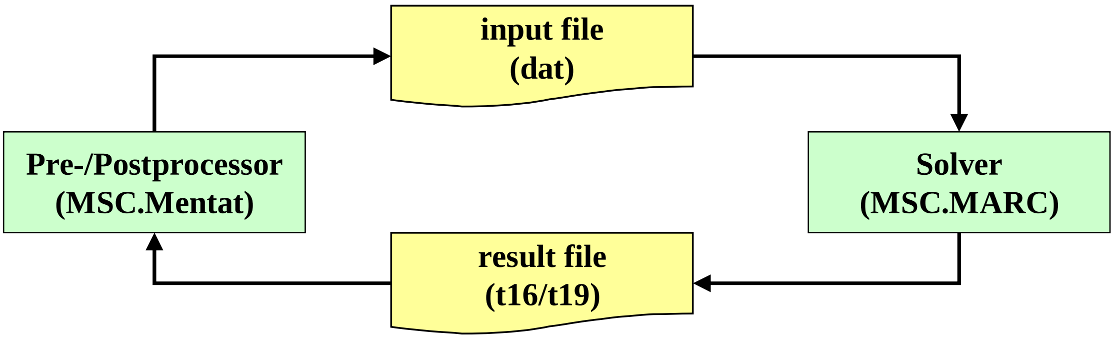


Figure 5. Information flow between MSC.MARC and Mentat without the new tool

The task of the processing tool introduced in Figure 3 is the “check-in” of new simulations into the database. Therefore, the input deck and the result file – which are both ASCII-formatted and thus can be easily accessed – are read and analysed by the application. It splits the data into suitable modules that can be stored in the database structure (see Figure 4). The most important constraints in that procedure are the avoidance of redundancies in the database as far as possible and the traceability of the simulation stored. To avoid redundant data, the processing tool detects similarities between the simulation model being checked-in and models already stored. If the new simulation only differs in certain parts, only these different parts are stored.

The application flow of the “check-in” of a simulation is depicted in Figure 6.

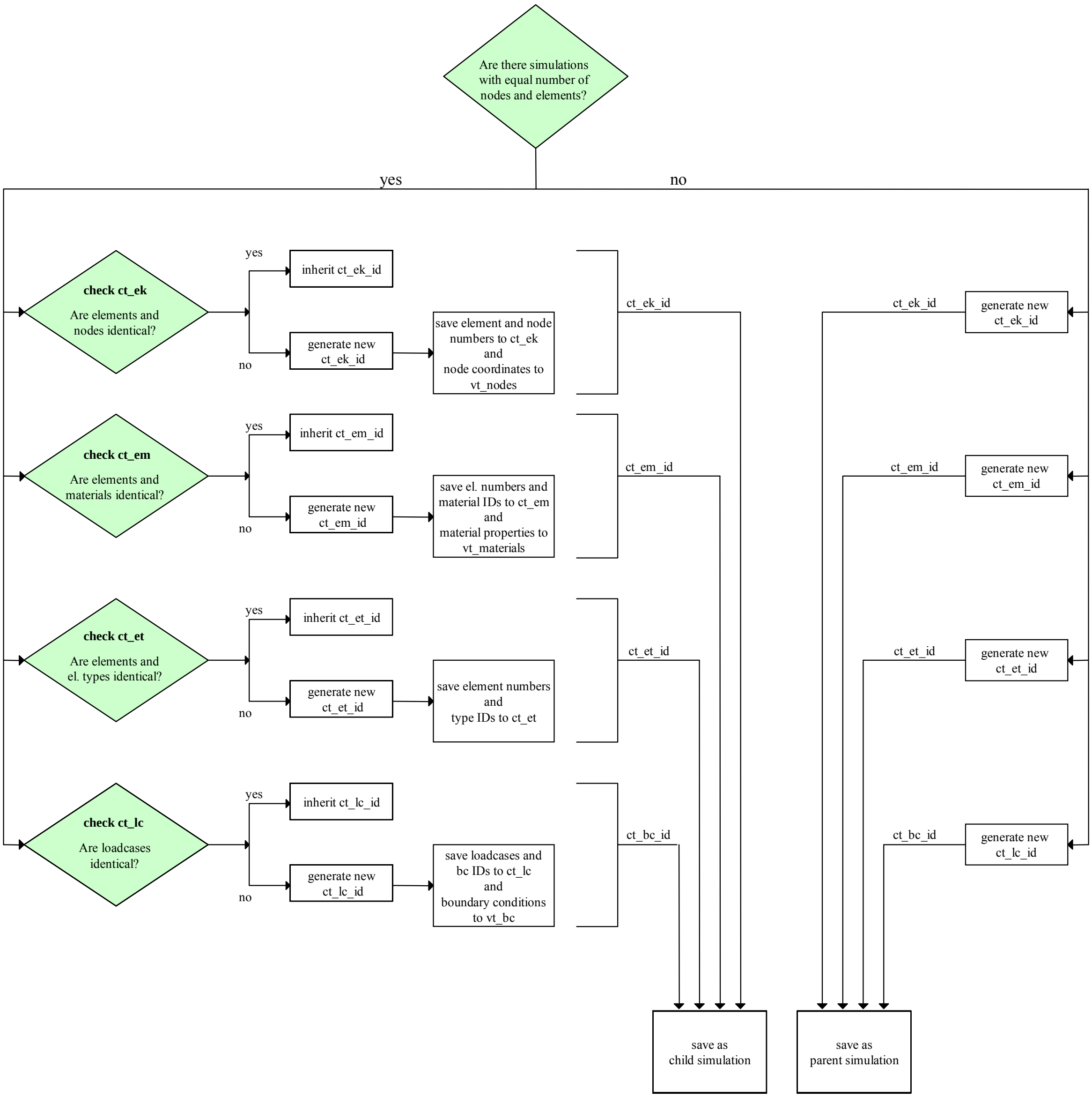

Are there simulations with equal number of nodes and elements?
yes
no
check ct_ek
Are elements and nodes identical?
yes
no
inherit ct_ek_id
generate new ct_ek_id
save element and node numbers to ct_ek and node coordinates to vt_nodes
ct_ek_id
check ct_em
Are elements and materials identical?
inherit ct_em_id
generate new ct_em_id
save el. numbers and material IDs to ct_em and material properties to vt_materials
ct_em_id
check ct_et
Are elements and el. types identical?
inherit ct_et_id
generate new ct_et_id
save element numbers and type IDs to ct_et
ct_et_id
check ct_lc
Are loadcases identical?
inherit ct_lc_id
generate new ct_lc_id
save loadcases and bc IDs to ct_lc and boundary conditions to vt_bc
ct_bc_id
generate new ct_ek_id
generate new ct_em_id
generate new ct_et_id
generate new ct_lc_id
save as child simulation
save as parent simulation

Figure 6. Application flow during the “check-in” of a simulation model

The first check is the overall number of nodes and elements in the simulation to store. To avoid the comparison of parts of models, that cannot have an identical mesh because these numbers differ, stored simulations with equal element and node-counts are identified. If no such simulation is found in the database, the one “checking-in” is completely stored to the database. Otherwise, the next step is the comparison of the connection tables, starting with “elements-nodes”-table. If all datasets comply with each other, the ID of the corresponding “elements-nodes” variant is written into the top-level table. As soon as a difference is recognized by the documentation tool, a new connection table “elements-nodes” ID is generated and the data is written into the connection table “elements nodes” and the values table “nodes”. If the “elements-nodes” table already exists, the node coordinates in the value table “nodes” are compared and if necessary, a new set is created. As a next step, the connection tables “elements-types” and the “elements-materials” are processed in the same way. If e.g. a material with identical properties already exists, only the connection table “elements-materials” is stored.

If one ore more tables are identical, the simulation is stored as a “child simulation”, otherwise as a “parent simulation”. Table 2 gives an example for the top-level table.

Table 2. Example for top level table with different simulations saved

ID simulation	ID el-nodes	ID el-type	ID el-material	ID loadcase	ID results
1	1	1	1	1	1
2	1	2	1	1	2
3	1	1	2	1	3
4	1	2	2	1	4
5	1	2	2	2	5
6	2	1	1	2	6

In this case, the data for the definition of six different simulations can be reduced to a maximum of two datasets per table.

3.4 Documentation files

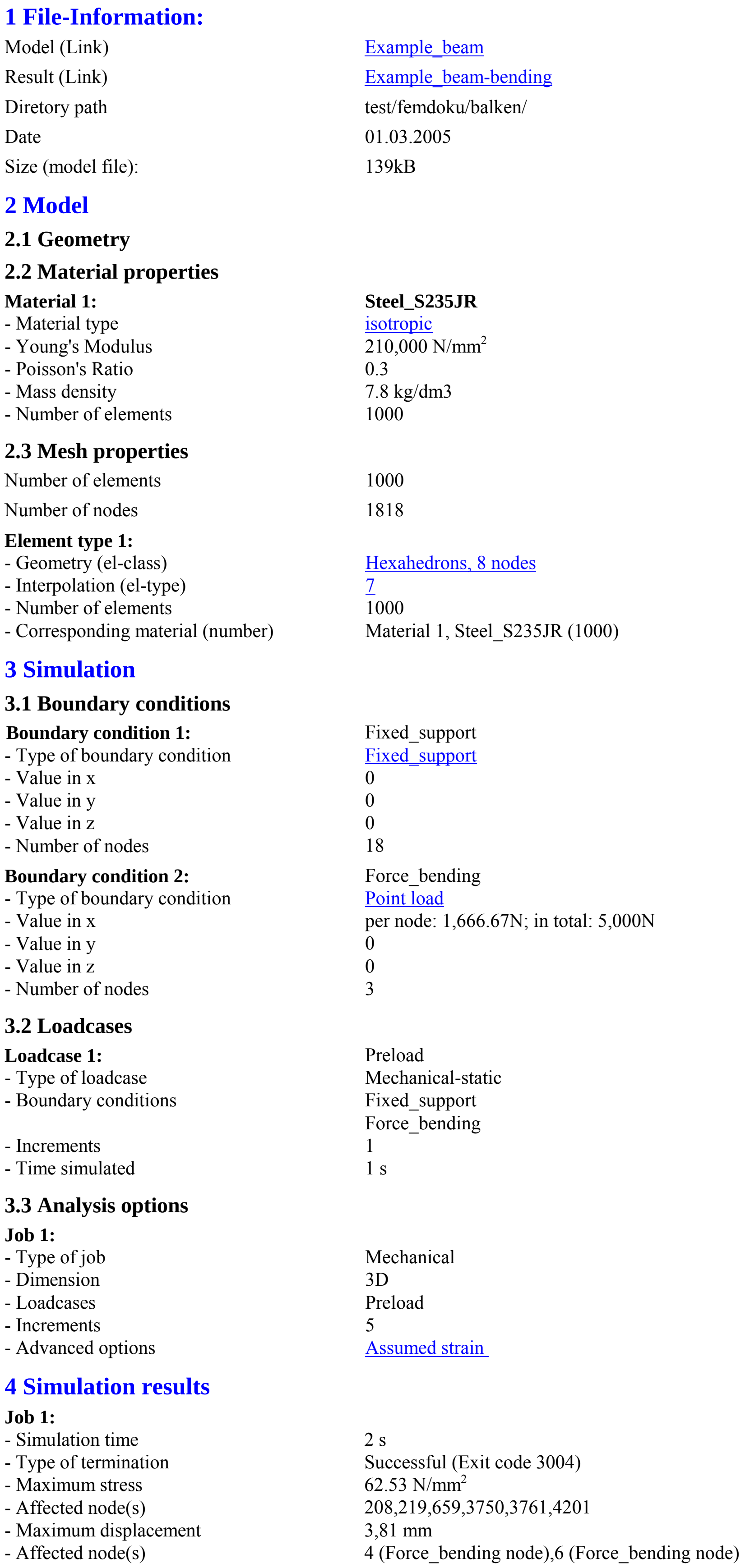
One important outcome of the database is the documentation of each simulation model and its results. In this approach, the prerequisites identified for the documentation files are:

- platform-independent format
- accessibility via intra- and/or internet
- security (access control, authentication)
- support for different “views” (FEA-specialist, person in charge for product development)
- embedding of pictures, tables,…
- most recent data as a basis
- freezing of specific status
- support for different languages

One format suiting very well to the prerequisites listed above is the HTML-format. Once the database is filled with information of simulation models and results, a documentation file can be generated automatically (e.g. using PHP-techniques). As mentioned before, the detail level can be customized to provide only the data relevant for the actual viewer. To make it possible to freeze a documentation file, either a facsimile can be created, or the dataset can be “write-protected”.

3.5 Documentation example

Figure 7 shows an excerpt from a documentation file of one variant of the example beam also used for the demonstration of influencing parameters in Chapter 2.1.



1 File-Information:

Model (Link)	Example_beam
Result (Link)	Example_beam-bending
Diretory path	test/femdoku/balken/
Date	01.03.2005
Size (model file):	139kB

2 Model

2.1 Geometry

2.2 Material properties

Material 1:	**Steel_S235JR**
- Material type	isotropic
- Young's Modulus	210,000 N/mm^2
- Poisson's Ratio	0.3
- Mass density	7.8 kg/dm3
- Number of elements	1000

2.3 Mesh properties

Number of elements	1000
Number of nodes	1818
Element type 1:	
- Geometry (el-class)	Hexahedrons, 8 nodes
- Interpolation (el-type)	7
- Number of elements	1000
- Corresponding material (number)	Material 1, Steel_S235JR (1000)

3 Simulation

3.1 Boundary conditions

Boundary condition 1:	Fixed_support
- Type of boundary condition	Fixed_support
- Value in x	0
- Value in y	0
- Value in z	0
- Number of nodes	18
Boundary condition 2:	Force_bending
- Type of boundary condition	Point load
- Value in x	per node: 1,666.67N; in total: 5,000N
- Value in y	0
- Value in z	0
- Number of nodes	3

3.2 Loadcases

Loadcase 1:	Preload
- Type of loadcase	Mechanical-static
- Boundary conditions	Fixed_support Force_bending
- Increments	1
- Time simulated	1 s

3.3 Analysis options

Job 1:	
- Type of job	Mechanical
- Dimension	3D
- Loadcases	Preload
- Increments	5
- Advanced options	Assumed strain

4 Simulation results

Job 1:	
- Simulation time	2 s
- Type of termination	Successful (Exit code 3004)
- Maximum stress	62.53 N/mm^2
- Affected node(s)	208,219,659,3750,3761,4201
- Maximum displacement	3,81 mm
- Affected node(s)	4 (Force_bending node),6 (Force_bending node)

Figure 7. Example of an automatically generated documentation file (without screenshots)

3.6 Comparison of simulation models

Besides the documentation of one single simulation file, the concept presented here is also able to compare simulations with each other. This is done right out of the database, where similarities and differences are easy-to-find and thus can be analysed. Depending on the user's preferences, either he/she can see the complete documentation files side by side or only the differences found in simulation model definitions and analysis results. This helps to find differences (e.g. a changed boundary condition) very quickly.

4 Conclusion

The concept presented in this paper can contribute to a better understanding and traceability of FEA-simulations, even if they are done a long time ago. The database approach allows accessing the documentation at runtime without any special software via intra- or intranet anytime. Additionally, the documentation can be suited to the specific needs of the viewer to give him all the information of interest for him.

Ideas for future work are that more detailed result information should be available from the database, e.g. stresses and deformations at points of specific interest. After that, the concept could be adapted to other pre-/postprocessors and solvers (e.g. Abaqus, Nastran/Patran, STEP-format) and the possibility to generate input files for solvers right out of the database could be implemented. This would allow modifying a finite element model very quickly (e.g. element type, material properties) without having to use the pre-processor.

References

[1] Cook, Robert D.: "Finite element modeling for stress analysis", John Wiley & Sons, New York et al, 1995

[2] Fröhlich, Peter: "FEM-Leitfaden", Springer, Berlin et al, 1995

[3] Beitz, W. and Grote, K.-H.: "DUBBEL – Taschenbuch für den Maschinenbau", Springer, Berlin et al, 2001

Dipl.-Ing. Rüdiger Maus
Prof. Dr.-Ing. Christian Weber

Saarland University
Institute of Engineering Design/CAD
Building 8.2
PO Box 151150
D-66041 Saarbrücken

Phone: +49 (0) 681 302 3075
Fax: +49 (0) 681 302 4858

E-mail: maus@cad.uni-saarland.de
weber@cad.uni-saarland.de
URL: www.cad.uni-saarland.de