Facing Future Challenges in Crash Simulation Engineering – Model Organization, Quality and Management at Porsche

Marcel Koch[1], Steffen Mattern[2], Robert D. Bitsche[3]
[1] Dr. Ing. h.c. F. Porsche AG, [2] DYNAmore GmbH, [3] SCALE GmbH

Abstract

Numerical simulation has become an indispensable tool in car crash analysis, reducing the number of physical experiments and driving the engineering process. While the predictive capabilities of the simulation models have greatly increased in recent years, so has the level of detail, the complexity and the number of load cases analyzed. Dealing with these complexities can become a burden distracting engineers from their main task: the vehicle development process. It is therefore essential to provide automated, standardized and robust simulation processes in order to support the engineers throughout all steps of pre-processing, simulation and post-processing.

In order to meet these requirements, Porsche has adopted a modularized model organization strategy where every structural member of the car is modelled as a self-contained component. This strategy allows for the independent modification and verification of components, and enables sharing of common components between different disciplines and projects. The large number of resulting components is managed by a simulation data management system called LoCo, which provides features for model assembly, version control and the distribution of data. The model assembly process to build a complete input deck for LS-DYNA® is based on an approach that promotes the re-use and sharing of modules.

After every modification of a component an automatic quality check is performed for the individual component. This "model check" is a multistage process and assures the compliance of every component with Porsche's model quality requirements. While checking of individual components cannot guarantee an error-free model, the chance of errors in the assembled model is greatly reduced.

The configuration and final mass of the car, the number and type of occupants as well as the position of the barrier and correct boundary conditions are chosen in accordance to the respective load case. The process allows, for instance, assembling a specific car configuration in multiple load cases simultaneously or running different derivatives in a specific load case. Also the modularized structure of the database builds a basis for the quick and error-free set up of robustness studies or multidisciplinary optimizations. This enables the engineers to quickly react to the questions that arise in the development process in their daily work with high quality and hence reliable simulation data.

The paper presents the modularized model organization strategy together with the automatic quality check of each module (i. e. input file for LS-DYNA) and describes the benefit of the process. It will be shown how the established process supports Porsche's engineers in facing today's and future challenges of crash simulation analysis leading to a further increase of the number of required simulations. It allows keeping the engineer's focus on the actual development process without struggling with the complexity of the simulation method.

Introduction

The significance of crashworthiness simulation, as well as the necessity of reliable and predictive simulation models in the context of vehicle development has been increasing continuously over the last years. Maximizing efficiency and performance while providing optimal safety and reducing the project period challenges today’s virtual vehicle development. At the same time, the minimization of the number of physical tests and the permanent need for integration of new technologies have to be in accordance to the existing and also to new legal and consumer regulations.

Facing these challenges requires an extremely detailed and precise representation of each of the car’s parts regarding geometry and material properties in order to provide high predictive quality with respect to kinematical and structural behavior in different loading situations. This complexity consequently increases the number of engineers working together at a car development project for functional and crashworthiness design.

To fulfill the need for fast and reliable predictions through simulation, efficient communication and accurately formulated standards within the group are essential. The latter is also important in order to enable an interdisciplinary exchange of models, including knowledge and experience – another part of a project engineer's daily business. In this context, properly revising and adapting the several single steps of the process chain involved in modern crash simulation is crucial. In order to support this process, standardization and automation of repeating tasks is therefore desired.

Deficiencies in structural design are supposed to be detected in an early stage by means of CAE supported virtual vehicle development for instance to avoid expensive modifications of the manufacturing tools. Furthermore, the effects of specific measures have to be evaluated at short notice in order not to compromise the project's schedule. Therefore, during the development process of a component, constructional stages available in CAD have to be converted thoroughly into high quality finite element models in a fast and error-free manner.

The complete crash model, that is, the simulation model representing each structural part used within the project, is subdivided into functional devices. In the following, we will refer to these logical sub-structures as *components*. The subdivision applied in the crash project mostly follows the organization in the CAD database. A simulation deck of a complete car for a crash load case – each simulation model using a specific set of boundary conditions will be entitled as *load case* in the following – is assembled by using 40 to 50 of these components. The entire project, with all its variants, configurations and derivatives, may require up to 120 components. Each of them has to be set up based on the current CAD construction model, leading to a simulation model for LS-DYNA which is compatible with every other component.

Coordinating and synchronizing all the different disciplines, working together in a project, such as meshing and component set up, component simulation and validation, structural crash, occupant and pedestrian safety, can be quite difficult, also because the sub-projects are often located at different sites. In order to assure consistency throughout the project, these different groups require (limited or full) access to a unified and standardized database containing the different components used in the project.

A full-car crash-model with occupants in current projects at Porsche consists of about 15 million elements leading to a simulation time of about 32 hours using 256 processors of the latest generation. In daily business, reduced models as well as sub-models are also often used in order to focus on the respective region of interest while having faster turnaround times. These smaller input decks are also assembled using the project's pool of components.

By allowing the project's components to be assembled to different configurations for various disciplines and load cases by the engineers with minimum effort, each single component has to be designed following a rigorous and consistent mandatory modeling guideline. The time for the model setup as well as the number of necessary simulation runs to obtain the desired results have to be reduced to a minimum. Consequently, different model organization conventions have to be defined throughout the entire project, such as setting of control cards, contact organization, recommended element types, part numbering ranges and so on, in order to clearly establish modeling rules for how components have to be set up. Aiming at an automated post processing, the components also have to comply with a defined nomenclature for history data. In order to assure high-quality full car models reliably predicting the actual structural behavior without requiring extensive (and hence expensive) debugging, these modeling rules must apply throughout the entire process. This means that also the component set up and its validation process have to be strictly in accordance with the guidelines. As a consequence, this allows exchanging the components among disciplines without adjusting them to the respective simulation context.

In the following, a modular modeling approach is presented as it is currently applied for crashworthiness projects at Porsche. The focus is laid on the modeling guidelines for setting up single components and their organization within the project's database. An automated component check procedure is introduced, supporting the engineer to comply with the modeling guidelines while setting up new components or modifying existing ones. Finally, the application of simulation data management (SDM) is described in order to assemble the components to simulation models for different load cases. This also enables access to the project data from different sites while automatically keeping the database synchronized. The modularized strategy, the rigorous quality requirements and the application of automation provide extensive support to the engineers focusing on their core business which is the development of high-quality vehicles.

Modular Modeling Strategy

Modeling Guidelines

Assembling reliable and robust full-car simulation models with a minimum error rate requires the establishment of a generalized modeling guideline. As every model is assembled using a subset of the project's components, this guideline has to be binding for each component type but also must define rules for the component to fit into the context of the different simulation models. This leads to a desired workflow within the crashworthiness project as described in Figure 1.

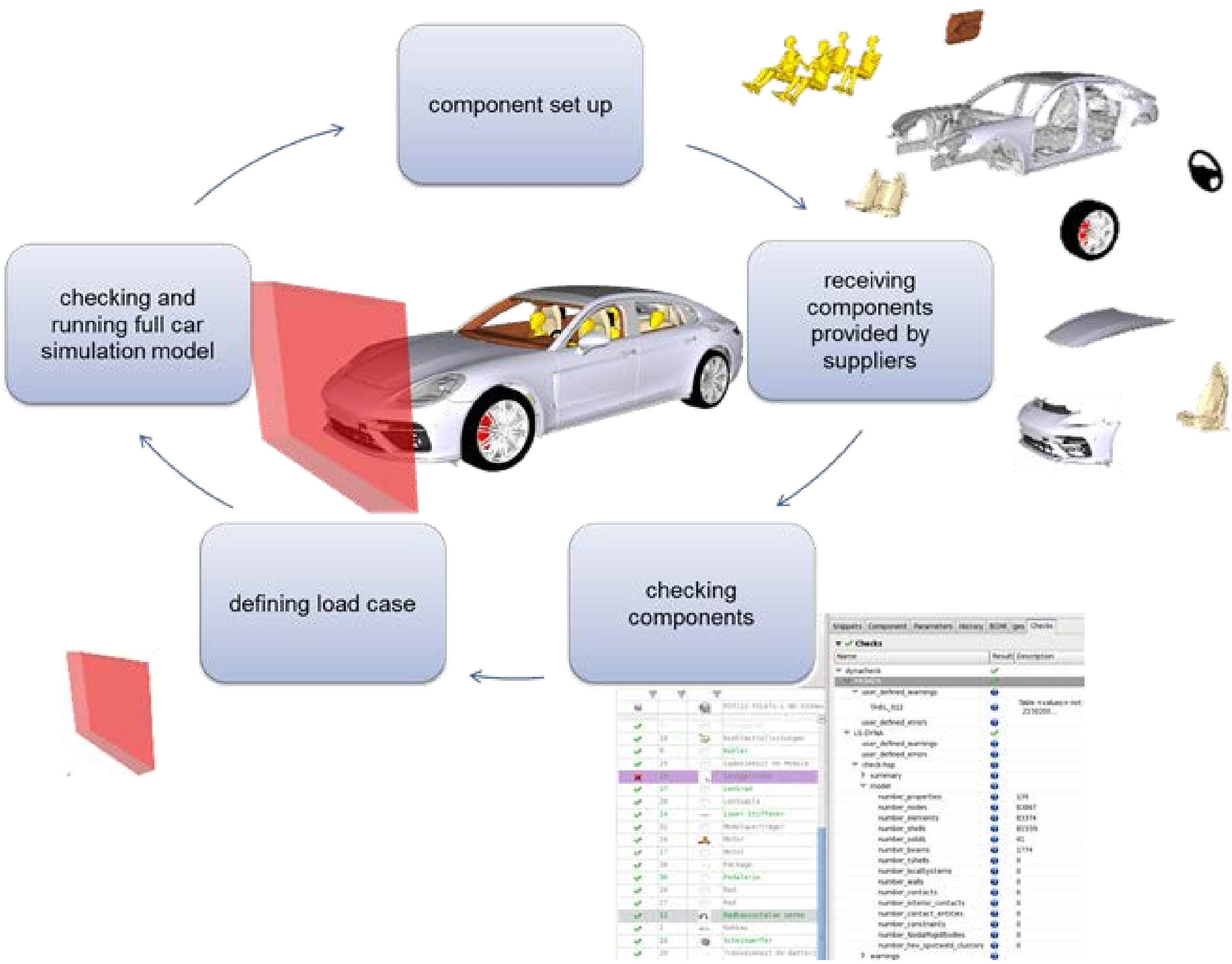

Figure 1: Workflow in crashworthiness projects

In order to serve the entire project, the guideline has to be formulated in a way to satisfy all the requirements of the fully equipped full-car models while simultaneously being valid for the components used in the context of smaller sub-models, functional simulations or component validations. Everything necessary to assure the component's optimal performance in each of its relevant environments has to be specified.

In order to better illustrate this concept, we shall use a more practical example. For instance, in a full-car crash model it is strongly recommended to keep the number of contacts as low as possible. Therefore, the global contacts are organized in a way that each component provides contributing lists of parts (this can be achieved by using the LS-DYNA keyword `SET_PART_COLLECT`). In order to assure adequate component performance without requiring extensive and time consuming adjustments, these contact lists are specified in the modeling guideline while the contact formulations are identical for each of the different disciplines throughout the project.

Further aspects prescribed by the modeling guidelines are, for example, numbering conventions, inter-modular connections, organization of central material databases, among several others. Besides model organization aspects, LS-DYNA specific settings are also taken into account such as control card settings, required solver versions, etc. For several component types, some additional component-specific guidelines must be formulated. This is the case for seatbelts or airbags whose modeling aspects might be irrelevant for other components.

Once a set of modeling guidelines is found, valid and – more importantly – benefiting for the different disciplines of the entire project, precisely establishing these rules is essential to allow the derivation of objective quality criteria for a single component. In turn, this enables automated component checks already in an early stage of modeling in order to assure that the final component be in compliance with the guideline and hence contribute to a high-quality simulation model.

Generalized Cross-Project Databases

Some components required to set up simulations are not developed within the course of the car project or not even developed by Porsche but rather purchased from external companies. These components, sometimes applied across different projects, are treated independently from the parts that actually constitute the car. They are included in independent databases organized in a generalized way in order to be applicable for parallel or later projects. Typically, these generalized databases are not modified while setting up the load cases for the car project.

Most of the material models, including the simulation models representing the connections, are not bound to one specific project since the same materials and connection strategies are used in different cars. These models are supervised by a specialized group of experts at Porsche and organized in a cross-project database. The different material cards are referenced in the structural modules by material IDs agreed across projects. All the material models required for specific projects are organized in specific LS-DYNA include files that do not contain any structural information (i. e., nodes or elements).

Most of the crash load cases typically require either stationary or moving barriers located at a specific position. The finite element models, especially those for the highly complex deformable barriers, are generated and validated by an external company. The barrier's boundary conditions, such as initial velocities, are specified for the respective load case where also its position is defined using, for instance, geometrical parameters of the specific car. The identical models are then applied for load cases in different projects as well.

Occupant safety simulations are intended to predict the behavior of dummies in the respective load case. The dummy models are accurate representations of the actual crash test dummies provided by specialized companies such as Humanetics or DYNAmore. Porsche purchases these models and, besides the positioning, which is usually performed in the form of a pre-simulation, the include files are generally not modified. Therefore, the dummy models constitute a third database which has to be treated separately from the components of the car.

Component Checking

In order to assure the quality of all the components, required by the cars project, as well as the compliance with the modeling requirements described in section *Modeling Guidelines*, a two-stage model check as schematically shown in Figure 2 has been implemented. The model check is carried out for each one of the car project's structural components while set up and whenever modified. The motivation is to obtain a pool of high-quality finite element models and consequently be able to assemble high-quality full car simulation models for each load case. The two-stage model check has been realized at Porsche in the form of a batch program, internally named as *DynaCheck* written in Python and organized as follows:

First, a model check is performed using a pre-processor, mainly in order to identify errors necessarily leading to an error termination in LS-DYNA. Currently, Porsche applies the pre-processor PRIMER® provided by Oasys Ltd., the software house of Arup [1], which could be replaced or also enhanced by other tools. PRIMER is able to perform over 3000 LS-DYNA specific checks and output the results as an ASCII-File. For example, a Part-ID, referenced in a `*SET_PART_LIST` which does not exist in the component, is going to be reported as error by the PRIMER check since this would inevitably lead to an error termination in LS-DYNA. The corresponding error message can be identified by a unique string (here PART_ABS) which is then used to add it to the list of relevant checks in *DynaCheck*. This list can be edited by the user in order to configure *DynaCheck* to be suitable for the entire project or also to specify individual check configurations for the different disciplines. Additionally, user-defined checks can be specified using PRIMER's built in JavaScript API in order to enhance its functionality. As another example, Porsche generally avoids the use of `*MAT_ELASTIC` for components in crash simulations due to the fact that energy conserving materials sometimes lead to numerical instabilities. However, using `*MAT_ELASTIC` for a structural part is not regarded as an error during the PRIMER check. Therefore, a user-defined routine has been implemented reporting every part using `*MAT_ELASTIC` which is then also added to the list of relevant checks in *DynaCheck.*

The different checks performed are classified in two categories, closely following LS-DYNA's concept of warnings and errors. If one or more errors are reported during the first stage, the check is aborted and the user is requested to revise the component. Warnings, on the other hand, are simply reported to *DynaCheck*'s log file and have no further consequences for the user. If none of the specified errors are detected, the program continues with the second stage.

In the second stage, an LS-DYNA run is executed applying the command line option 'mcheck=1' in order to perform a full initialization and terminate after 10 compute cycles. Subsequently, several of DYNAmore's LS-DYNA Tools [2] – e. g., *check-hsp*, *check-c*, etc. – are executed to extract information from the d3hsp, binout and message files in a structured XML-format. By parsing this XML-file, several checks are performed. For example, Porsche's post-processing processes expect the names of time history nodes to be in a specific format. If a `*DATABASE_HISTORY_NODE_ID` definition is not in compliance with this format, an error is reported by *DynaCheck*. In addition to the quality checks, further information (e.g., component mass) is extracted from LS-DYNA's output and stored – besides the check results – as metadata with the component.

Basically, every component, which is necessary to assemble a complete simulation for any of the project's load cases, has to undergo the rigorous check process. A few exceptions are, for instance, components without structural content like the global settings in the Control- and Database-Keywords. Although these include files are not checked explicitly, every component is verified in the context of the project settings. For example, the time step typically used in crash simulations has to be properly set in *CONTROL_TIMESTEP for the model check. Furthermore, components coming from generalized databases which are not developed during the project are not individually checked either. Currently, the checking strategy is only applied to structural crash and occupant safety load cases. However, this shall be extended to other disciplines in the near future.

Enforcing every structural component to be checked individually leads to a project pool of components in compliance with the high quality standards of Porsche. This process minimizes the probability of model errors regarding the different load case simulations as they are performed using a subset of checked components. Another benefit of this approach is enabled by features provided by LS-DYNA such as the previously mentioned `SET_PART_COLLECT` for organizing contact sets. It allows straightforward exchanging, adding or removing components from assembled load case simulations with a minimized probability of compatibility errors. The assembly process will be described in the following section.

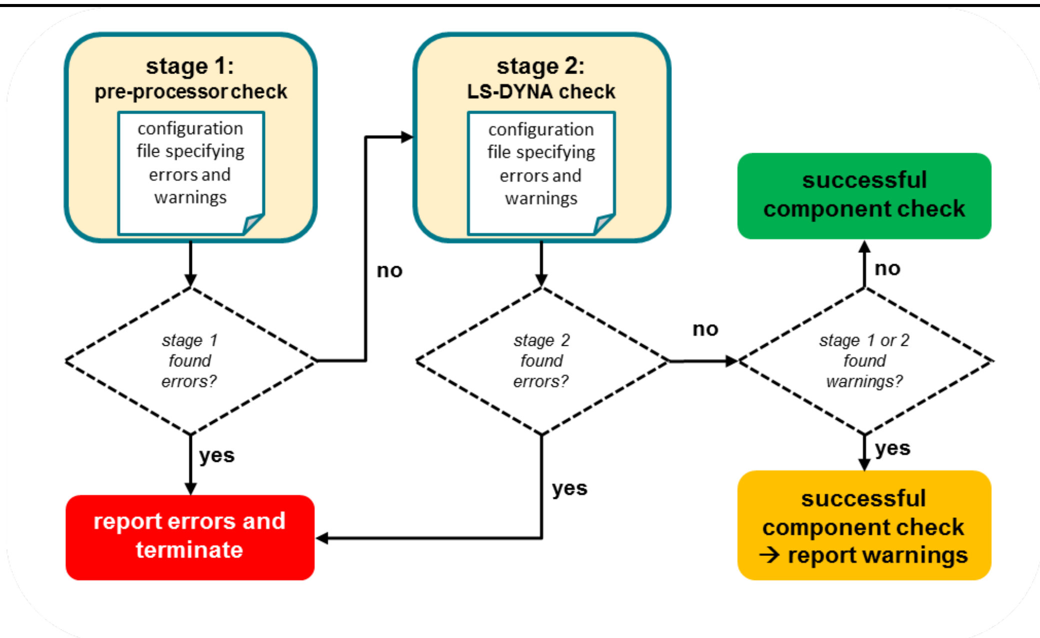

Figure 2: Program flow of check environment *DynaCheck*

Assembling Crash Simulation Models

The final crash simulation models, i. e., the different variants and derivatives of the car in various load cases, are now assembled using a subset of the pool of checked components together with components from the cross-project databases described above. Figure 3 shows a schematic view of the assembly process. In this example, component #1 is present in each of the three load cases while Assembly #2 does not require one of the two available components #2. Additionally, the general cross-project databases are organized and referenced in the correct simulation models.

Nowadays, a single engineer is responsible for a huge number of different load cases and variants and has to be able to assemble variants on short notice. For this reason, it is necessary to provide adequate support to ensure an error-free assembly process. A solution can be found in the application of a simulation data management (SDM) tool which the previously described process of component checking can be integrated into. The access to the project database from different sites, while keeping it synchronized, is managed by SDM as well. This is described in the following section.

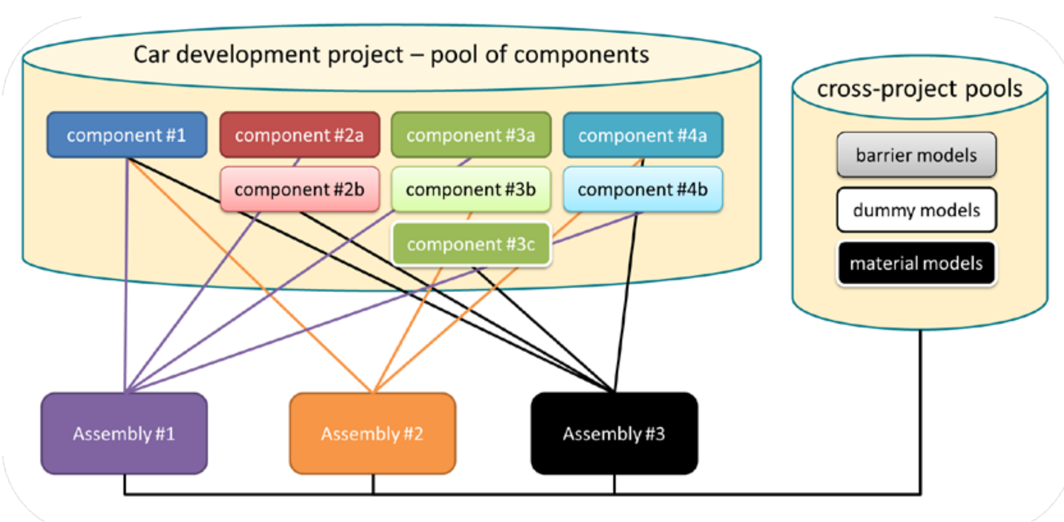

Figure 3: Assembling different crash simulation models

Model Administration with Simulation Data Management

Simulation data management applies ideas from product data management (PDM) and software engineering to simulation data. It is important to note that SDM differs from PDM due to the huge amounts of data created, the relatively short lifecycle of that data and the fact that simulation data is created in iterative learning cycles [4].

Porsche uses an SDM system called LoCo which has been developed by SCALE [3] in close collaboration with the German automotive industry over the past 10 years. LoCo is a client-server system providing features for model assembly, version control, access control and the distribution of and remote access to data.

The model assembly process is attribute-based. Standardized attributes are assigned to every component to describe the component's range of application. During the model assembly process, LoCo chooses suitable components based on a list of attributes describing the simulation run. Figure 3 shows a graphic representation of that process. Currently, Porsche defines 70 attributes together with their admissible values. Examples include vehicle derivative, discipline, analysis type, load case, world region, regulation, speed, engine, gear box, seat type, braking system, tire size, roof, seat type, side of impact, dummy, etc.

Version control (the management of changes to data) is an important aspect of simulation data management as simulation is "change driven" and simulation data is created in iterative learning cycles. LoCo's version control system is based on a copy-modify-merge approach which allows every simulation engineer to work with an independent copy of all data. This is the natural choice for an environment in which individual engineers must be able to observe the effects of their own changes in isolation from the changes made by others. The "branches" emerging from this approach are routinely merged to form a new baseline for further development. Figure 4 shows a section of a project's history graph. Every box in this graph represents a snapshot of the simulation input data at a certain point in time. The vertical axis is a time axis with new versions appearing at the bottom. In this example, two branches emerged from David's version 559: David's version 563 and John's version 562. Later, John merged his own changes (version 562) and David's changes (version 563) creating version 564.

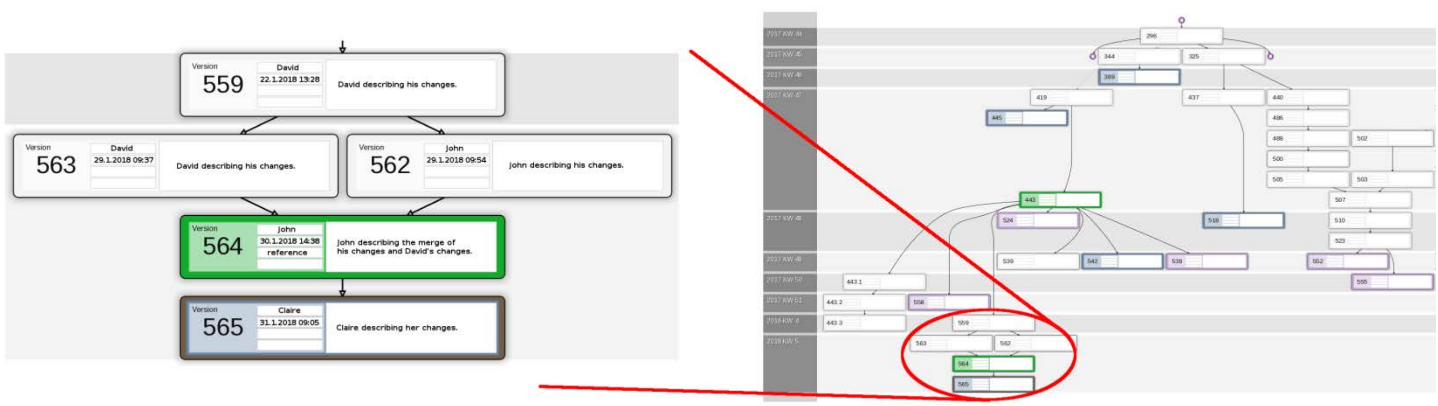

Figure 4: Section of a History Graph in LoCo.

Recently an alternative version control mechanism designated "*live mode*" was introduced in LoCo driven by Porsche. Engineers may temporarily switch to live mode in order to collaboratively work on the same copy of data for a certain period of time [5]. The live mode feature is mainly used in situations where the normal version control mechanism seems inappropriate. An example for such a situation is the "model set up phase", where engineers work together closely and would create a lot of unnecessary versions using the standard version control approach.

Besides the actual input files for LS-DYNA, various scripts and configuration files controlling pre- and post-processing tasks as well as the interaction with the HPC system (e. g., submit scripts) are also treated as components and managed and versioned by the SDM system. As a consequence, the same attribute-based logic described above can be used to control the content and execution of these scripts and configuration files. Furthermore, consistent version control of these scripts and configuration files is very important as it guarantees reproducibility of previous results.

Data distribution is based on the idea of automatic synchronization of relevant data between the LoCo server and the simulation engineer's computer where the LoCo client software is running. This makes access to already synchronized data very fast and keeps the LoCo client operational even if the user is offline. For users without direct connection to the LoCo server, an alternative synchronization process via file transfer is available which can be used in order to cooperate with external suppliers.

The component checking software *DynaCheck* described in Section *Component Checking* has been closely integrated into the SDM system. Component checks can be initiated manually or automatically whenever a component is modified. A mini-model is then assembled containing the chosen component and other essential include files (e. g., material definitions, control cards, etc.) and *DynaCheck* is started as shown in Figure 2. The necessary configuration files specifying errors and warnings are also organized in LoCo and treated as a cross-project database described in Section *Generalized Cross-Project Databases*. *DynaCheck* returns the check results which are displayed directly in the LoCo client software, as shown in Figure 5. Besides, the component check results are also transmitted to other users over the network so that double checking of the same component can be avoided.

In the course of the component check, the component's mass (as computed by LS-DYNA) is also determined and stored as meta data. When a simulation run is assembled, the total mass of the car is computed as the sum of all component masses and additional non-structural mass can be distributed automatically in order to obtain a certain target vehicle mass.

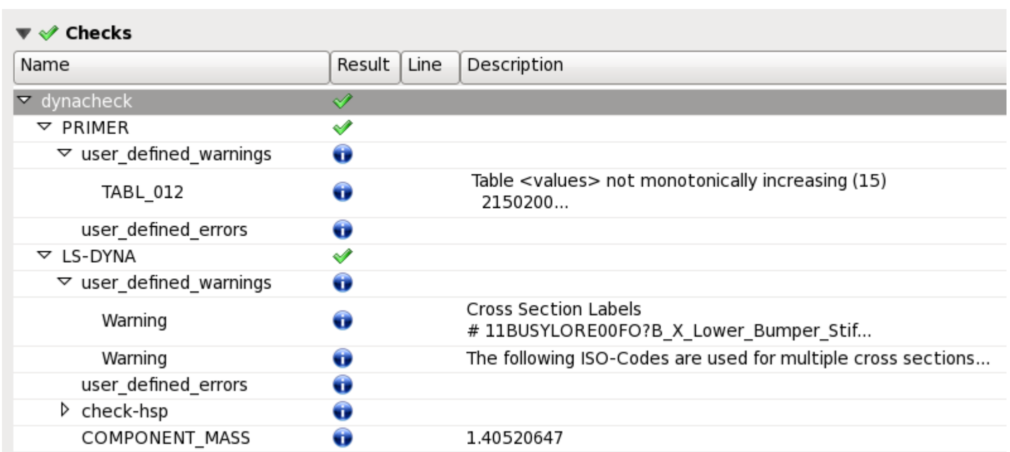

Figure 5: Results from a successful component check with DynaCheck as shown in LoCo.

The complete integration of the modular modeling strategy together with the automatic component check allows the quick and error-free assembly of different load cases. This greatly supports the engineer's work, saves time and minimizes the number of useless simulation runs.

Flexible Assembly of High-Quality Model Variants

The described concept of a modular approach based on components following a strict modeling guideline sets a foundation for high quality automotive crash simulation models. The strategy assures that most errors and shortcomings are detected already during model set up, that is, before a component is actually integrated into a complex load case simulation. This saves time for debugging and increases quality and reliability of the simulations. Adequate automation and integration of the process steps guarantees an objective application of the quality requirements to each component. The simulation engineers receive consistent and, as far as possible, error-free components, being in compliance with the requirements prescribed by the project. Integrating the component organization, checking and the model assembly into LoCo allows for many engineers to work together and supports them to meet today's challenge of handling different car derivatives and variants in many load cases while, at the same time, providing significant and reliable results in complex and detailed simulation models.

The concept presented in this paper is to be extended gradually in order to face the future challenges in various simulation-based disciplines. The automatic model assembly process described above allows for the quick and error-free assembly of a large number of models, which is of paramount importance for robustness studies, the identification of numerical or structural sensitivities and mathematical optimization. Comprehensive modeling guidelines devised by experts from several disciplines will help to extend the models' range of applicability, promote the exchange of models between disciplines and will open up for new possibilities in multidisciplinary analysis.

References

[1] Oasys PRIMER, http://www.oasys-software.com/dyna/en/software/primer.shtml, 2018

[2] LS-DYNA Tools, https://www.dynamore.de/en/products/tools, 2018

[3] LoCo, https://www.scale.eu/en/products/loco, 2018

[4] "What is simulation data management?", NAFEMS, available online: https://www.nafems.org

[5] Bitsche, R. D., Thiele, M., Landschoff, T., Koch, M., Recent Developments in LoCo - Instant Collaboration in Simulation Data Management. Presented at: 11th European LS-DYNA Conference, 9 - 11 May 2017, Salzburg, Austria.