COMPUTER-AIDED DESIGN & APPLICATIONS, 2017
VOL. 14, NO. 5, 549–562
http://dx.doi.org/10.1080/16864360.2016.1273575

Flexible parameterization strategies in automotive 3D vehicle layout

Patrick Rossbacher and Mario Hirz

Graz University of Technology, Austria

ABSTRACT

Today, automotive design has to face numerous exciting challenges. The growing globalization causes an intensified competition amongst car manufacturers and forces them to reduce the required development time in order to shorten time to market, to appear first with attractive new products. Efficient and flexible processes and tools are necessary to handle the arising complexity efficiently. Parametric-associative 3D-CAD systems offer ideal conditions to face this challenge in virtual development. The present paper focusses on a special issue in automotive concept phase – the vehicle architecture layout process and required parameterization strategies.

In most cases, parametric-associative relations defined within 3D-CAD models are of rigid kind. This implies that a formula, which is defined within a 3D-CAD model in order to evaluate a specific parameter, cannot change the input/output situation of involved parameters. In most application cases, this disadvantage can be neglected, but not in case of vehicle layouting in the early concept phase. Since geometric boundary conditions which define the geometric base of a vehicle concept can vary significantly, a rigid model parameterization is not the proper solution and prevents efficient reuse of 3D-CAD models. Additionally, rigid parameterization concepts lack of the required flexibility when having to manage multiple design variants in a single model. Therefore, the present paper outlines a possible strategy, which enables the use of advantages of parametric-associative design, while allowing changes of relations-evaluation behavior in context of respective technical issues and simultaneously preserving necessary geometrical model consistency.

KEYWORDS

Flexible parameterization; 3D-CAD methods; vehicle layout; automotive concept phase

1. Introduction

Parametric-associative 3D-CAD programs have become state of the art in design and development of complex high-tech products. Their ability to quickly adapt CAD models to new requirements by the possibility of modifying parameters and input geometry is supporting engineers to reduce development time throughout all phases of the engineering process. Furthermore, it offers the ability of creating and evaluating multiple design solution variants within a short time. Especially in automotive conceptual development, parametric-associative design methods provide great potential for quick and efficient generation of geometry models as well as supply of subsequent engineering processes with required information. Two aspects are fundamental with respect to the conceptual definition phase. On the one hand, parametric-associative models are able to store defining product design knowledge within the CAD datasets, which enables engineers to reuse not only the sole geometry, but also the defining process in behind. On the other hand, modern CAD software offers automation possibilities, which enable an extension of the originating software functionalities by new and specialized ones, and the generation of entirely new engineering tools to support solution finding of specific and complex technical design questions. The CAD functions are provided by the so called API (Application Program Interface), which is externalizing the functions by providing them for instance as *.dll (dynamic link library) or COM (Component Object Model) library. Through this, the CAD program functions can be used by external script languages like VB.Net to create specific software solutions, capable of automating design routines in the CAD program itself up to the entire design process automation.

In order to provide the mentioned advantages effectively, an extensive planning phase is required, prior to the actual geometry modeling process [13]. This is necessary, because one benefit of parametric models comes with their continuous reusing. In order to obtain a maximum reusing factor of the model, it is necessary that modifications of the model can be retrieved either by changing defining parameters or by exchanging the input geometry. This requires that all possible design variations and design options have to be planned and thought

CONTACT Patrick Rossbacher patrick.rossbacher@tugraz.at

ahead, prior the actual CAD model is created. Furthermore, the used features, parameters and relations have to be structured in a way that the model remains readable, and that design changes remain reproducible for involved engineers.

The present paper is focusing on the initial design task in automotive conceptual development – the vehicle architecture layout process. This job represents one of the most complex steps in conceptual development on top vehicle level. There are two reasons for that high degree of complexity: On the one hand, the automotive concept phase has a highly dynamic character. Requirements and boundary conditions can change quickly, since the process has to respond to new developments and situations on the global market [11]. On the other hand, the technology, which enables different functionalities and properties of the vehicle, is either not fully known or not completely described in this early stage of product development. By this, traditional, relatively rigid parameterization strategies, as used in parametric-associative design on regular part and product level, are insufficient. Additional challenges arise when taking into account the design process itself and resulting interactions between OEM (Original Equipment Manufacturer) and engineering suppliers. In automotive development, a considerable share of design-related workload is not handled by the OEM alone, but being processed by supporting engineering suppliers. Because of this situation of workload sharing, a lot of standardization work in modeling and parameterization is required, in order to merge the different types of design data, delivered by several companies, into one coherent 3D vehicle dataset.

In this context, the present paper introduces and discusses an approach that combines the advantages of parametric-associative design with geometrical and relational flexibility at the same time to face the high requirements on data structure and process integration in automotive conceptual development.

2. Problem statement

As introduced in the previous section, parametric-associative 3D-CAD software is not only used for traditional design tasks in automotive conceptual design, like modeling parts or building up assemblies, but also to manage the architectural layout of a vehicle in order to define available space and positions for necessary technical components. This is a challenging task since various partially contradictory requirements have to be considered during the architecture definition process. Fig. 1 shows an excerpt of common requirements onto a vehicle concept.

This leads to a conflicting situation amongst the different influencing parties and requires the target-oriented search for the best compromise regarding desired technical properties of the vehicle. The initial step in conceptual development is building up a so-called layout model, in order to evaluate the basic geometric correlations within the emerging concept. Fig. 2 illustrates a typical 3D vehicle layout at an early development stage.

It contains the general outer and inner vehicle main dimensions, including simplified ergonomical representations of driver and passengers. In order to retrieve an overview of the spatial situation, proportional parametric

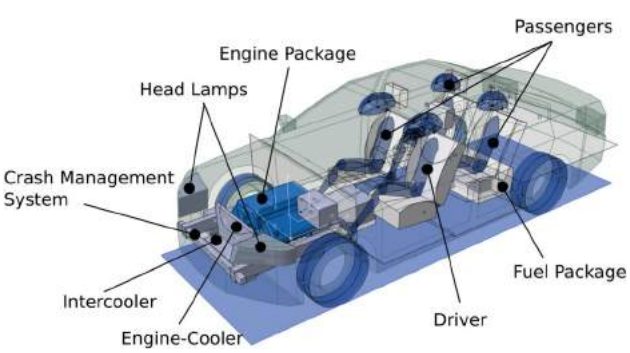

Figure 2. Exemplary basic vehicle layout.

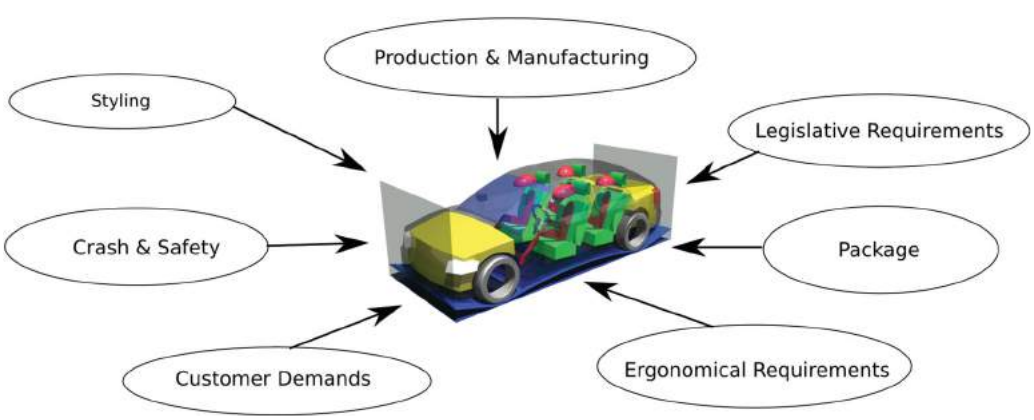

Figure 1. General requirements in vehicle development.

models can be used to estimate the extensions of required space for specific technical components considering their geometrical requirements and the interaction with an initial car body design. Therefore, the 3D-layout model allows a quick derivation of technical chain dimensions of the concept and enables the engineer to efficiently evaluate basic geometric properties of the concept. In order to develop and evaluate the vehicle package, functional based space requirements, like necessary clearances to other components, have to be taken into account, in addition to sole geometric space analysis.

In a variable 3D-CAD layout model, the geometrical dimensions are controlled by parameters to enable the creation and analysis of different dimensional constellations and to evaluate their influences onto the vehicle architecture. Depending on the specific design questions, the sequence of development steps for the layout definition of a new car model may vary significantly for one project to the other. Initial car development can be done for instance from inside to outside or vice versa. Another possibility is to start from an existing vehicle platform, where basic dimensions are already defined.

As illustrated in Fig. 3, a possible determination of the vehicle length (L103) includes the summation of the overhang front (L104), the wheelbase (L101) and the overhang rear (L105). The wheelbase itself results in this example out of the difference of the wheel center x-coordinates. In terms of a CAD model, the following formulas are required:

$$\begin{aligned} L103 &= L104 + L101 + L105 \\ L101 &= x_{RearWheel} - x_{FrontWheel} \end{aligned} \tag{2.1}$$

A modification of one of the input parameters will consequently lead to an automatic evaluation of the defined formulas and provides the output parameter values to be updated. As mentioned before, the boundary conditions in conceptual development may change rapidly, which may for instance require that the vehicle length is not evaluated anymore. But it may serve as an input to the system, because it has been defined as a binding target for the vehicle project. It's obvious in case of defining the vehicle length, that there is no unique solution for the problem, since the relations are not of bijective kind.

More complex situations arise, when integrating proportional models into the layout, leading to more chain dimensions, which have to be evaluated, as shown in Fig. 4.

As visible in Fig. 4, conventional parameterization strategies fail in this case. This points to the need of alternative parameterization strategies, capable of supporting the vehicle layout definition under consideration of rapidly changing boundary conditions. Therefore, three main challenges can be identified for automotive conceptual layout, in the context of flexible parameterization strategies:

1. Ability of model relations forming
2. Variable input data coverage
3. Preserving entire model integrity

Forming of relations is essential for a parameterization strategy concerning automotive 3D vehicle layout, as demonstrated before. Since the sequences of design processes are always different from project to project, the challenge for the development of a universal method for vehicle layout lies in the creation of a parametric-associative model, which supports model modification through adaption of parameters and exchangeable geometry. Because of this, there is no standard for required relations or their behavior available, which makes a default model parameterization impossible.

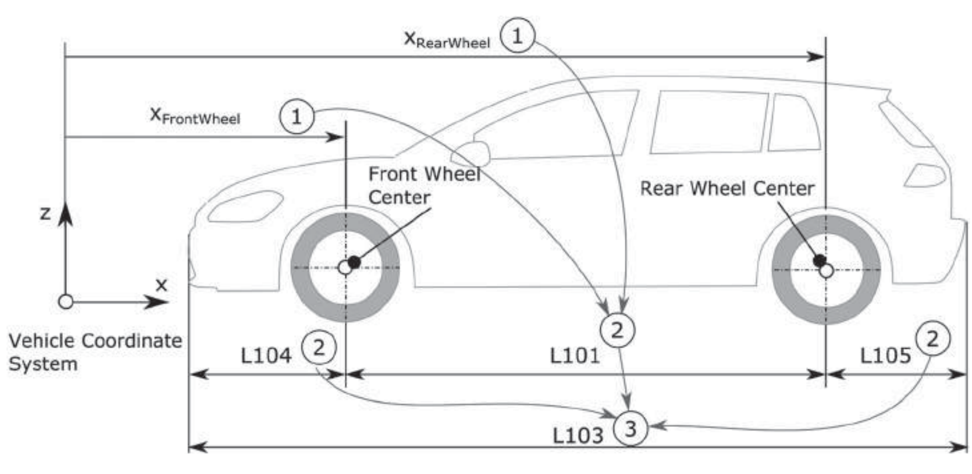

Figure 3. Typical determination of the vehicle length.

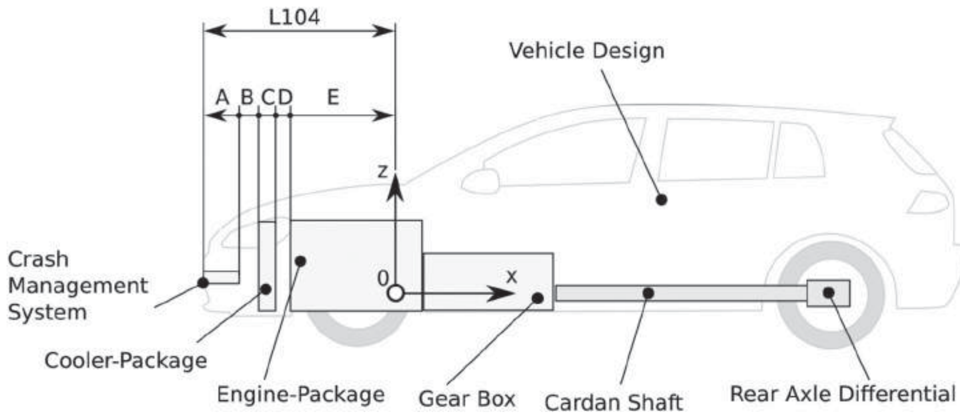

Figure 4. Exemplary chain dimensions in the vehicle front.

Variable input data coverage results from the descri bed situation above, that both technology and required functions are not fully defined in this initial product development stage. Focusing the example above, the relations in the vehicle front are known, but the values for clearances and the dimensions of the proportional models might be not. A possible parameterization strategy has to preserve model stability and data integrity, even if input parameters are missing.

The requirement for preserving model integrity, results from the demand of uniquely reproducing the parametric situation of the CAD-model at any time. This is an important key in understanding the resulting characteristics of a vehicle concept.

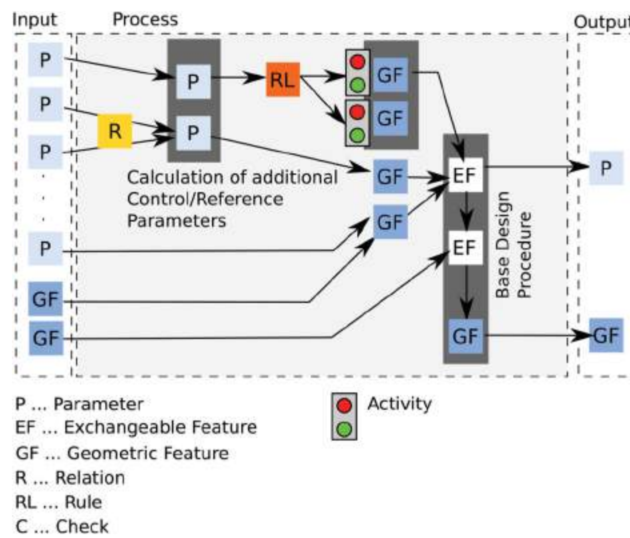

Figure 5. General parametric data workflows within a CAD model.

3. State of the art

Modern commercial 3D-CAD software provides a huge set of different parametric functionalities out of the box, and as known from traditional design-oriented software functions, which include specific knowledge-based features like formulas, rules, checks and the possibility of integrating repetitive tasks by automation routines or scripted actions. A general overview of parametric data occurring within a CAD model is illustrated in Fig. 5.

From a general point of view, a CAD model is characterized by an input-process-output (ipo) structure. The input of the model can contain parameters (P) and defining geometric features (GF). The process component contains used design features, in order to generate required geometry and geometry-related data like center of gravity, inertia or other measurable properties. The geometric features are structured chronologically according to the intended design history of the model. On top level of the used geometric functions, knowledge features can be applied for different use cases of the model. Mathematical relations (R) can evaluate functional properties of the model, or calculate further design parameters. Rules can control different design options of the model by in-/activating specific areas or sole features of the CAD model, or switch evaluating mathematical relations for the determination of functional properties. Exchangeable features allow the adaption according to the requirements of different input geometry without the need of modifying the model interactively. Another way to integrate knowledge into a model includes the use of automation interfaces. They enable the engineers to create user-specific functions, which can also be used by other designers, and to create entirely new tools based on and within the originating CAD system.

In this way, parametric models can become quite complex, when having implemented lots of knowledge-engineering features and multiple design variants. This complexity can lead to problems in view of design work and data management, like irreproducible changes, lacking transparency and maintenance effort. Furthermore, applied parameterization strategies are varying according to the respective methodological skills of the creating users. This prevents a continuous provision of constant data quality, which is required to enable a stable data process. Therefore, the use of such models is often restricted to a small range of specialized engineers [7]. For efficient work, it is a necessity to understand the implemented parameterization strategy of the parametric dataset.

Focusing on automotive development there exist a variety of several specialized approaches targeting the support of the early conceptual development stage by parametric tools like [1], [2], [3], [4], [5], [6], [8], [10] and [11]. All have the attempt in common, to provide essential package models as soon as possible within the development process to be able to evaluate both geometrical and functional concept properties already in the early conceptual development stage. All approaches are based on a more or less rigid parameterization concept. Mathematical relations, used to obtain relevant parameter values, are built in the model. If relations have to be formed, the model has to be adapted by hand. This may result in high modification needs, which often lead to a complete new building-up of the model, or to avoiding the changes at all.

Therefore, the present approach is examining an idea of a flexible parameterization method for layout models in early vehicle design. The intention is to provide a basic strategy how to use and combine the present functionalities of a CAD-system in order to benefit from the parametric-associative capabilities of modern 3D-CAD-systems and preserve required flexibility of defined relations within the model in face of forming them according to the respective project demands.

4. The flexible parameterization framework

One essential key to reduce development time in automotive design includes the optimization of workflows in the early concept phase. Within the concept phase, so-called expert tools are used within the responsible technical departments and on architectural level, in order to obtain initial 3D models, regarding the functional needs. Those expert tools are, in terms of CAD, often of parametric-associative kind. An excerpt of possible expert tools is illustrated in Fig. 6.

In order to provide an efficient design process and consistent concept 3D data within the concept phase, the different expert tools have to work on the same parametric data basis. This is not easy to be implemented since expert tools can have very complex input/output data correlations and may have dependencies to other tools (see Fig. 7).

As shown in Fig. 7 an expert tool is considered to be a CAD dataset (part or assembly). Such CAD datasets can have relatively simple input definitions like shown in figure (a) up to complex database dependencies, including multiple parametric variants of the part, illustrated in the figure (b).

By this, there is a steady demand for new strategies, supporting virtual development in automotive design. One possibility to support an efficient linking of different parametric CAD expert tools includes the introduced flexible parameterization framework (FPF) which is being discussed in detail in the following sections.

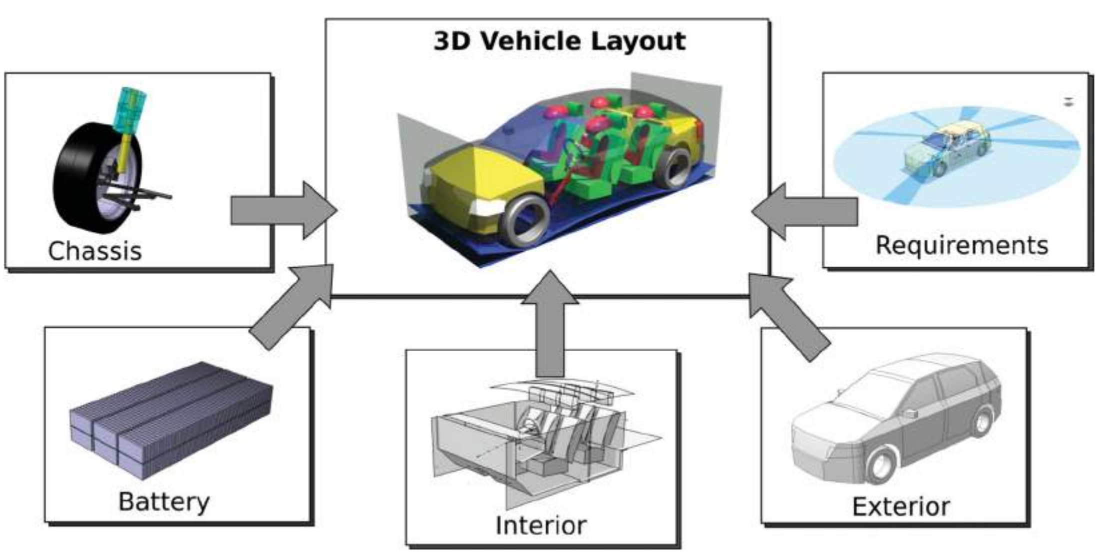

Figure 6. Different applications of expert tools in automotive conceptual design.

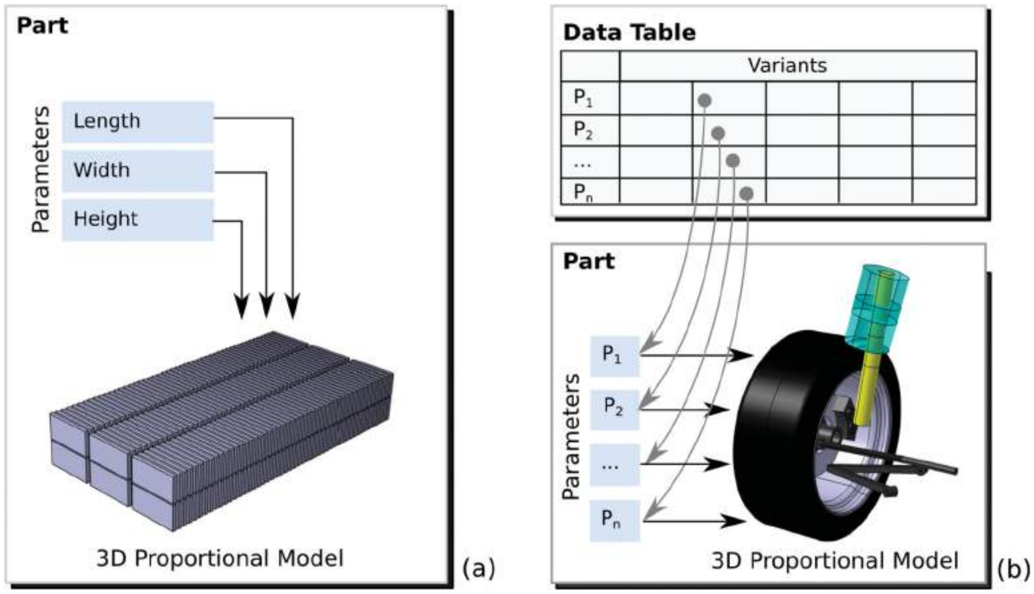

Figure 7. Possible expert module structures.

4.1. Essential components of the flexible parameterization framework

The basic idea of the introduced strategy is to provide a flexible parameterization framework for linking different parametric CAD expert tools together, with the target to support a consistent conceptual development, whereas the parameterization framework is considered to be on top of the actual CAD system. By this, parameter and relation modifications are not performed within the single CAD datasets, but through the superior flexible parameterization framework.

All parametric and logical relations in the CAD model have to be "watched" by the parameterization framework. Since most of the models are not made from scratch and already stem from existing databases within a company, the flexible parameterization framework must be able to handle existing parametric models and to preserve required data consistency as well. This is achieved by using hierarchical structuring of the applied design features within the CAD dataset.

This requires that every expert tool receives the proper geometric parameters, boundary conditions and input geometry. This is a huge challenge, since the involved expert tools may require the same parameters or input geometry but with different granularity or ranges. Therefore, this type of framework provides a continuously running service, which supports definition of parameters and relations in conceptual modeling processes on top of the regular functions of the used CAD system. An overview of the general structure of flexible parameterization framework is presented in Fig. 8.

The basic components of the parameterization framework are:

- Consistency Control
- Parameter Model
- Parameter Interface

The consistency control provides a service, which is attached to the parameter model. It continuously checks if parameters reside within their defined boundaries (plausibility conditions), if they have to be in-/activated, or if an update is necessary, when a parameter value or a relation forming occurs.

The parameter model includes the definition of model-wide parameters and relations and the relation solver. In context of the present approach, relations are equal to every formal connection between parameters. They are classified into formulas, chain dimensions and measurements, obtained from geometric features of the CAD model, whereby formulas as well as measurements are not formable in comparison to the chain dimensions. Internal relations or parameter within an expert tool are not taken into account.

The parameter interfaces serve as transferring mechanisms of parameter values between the parameterization framework and the corresponding expert models and includes a geometrical stability control module, which observes the geometric model stability (see section 4.2).

4.1.1. Consistency control

The consistency control represents the superior system to the parameter model, providing essential functionalities

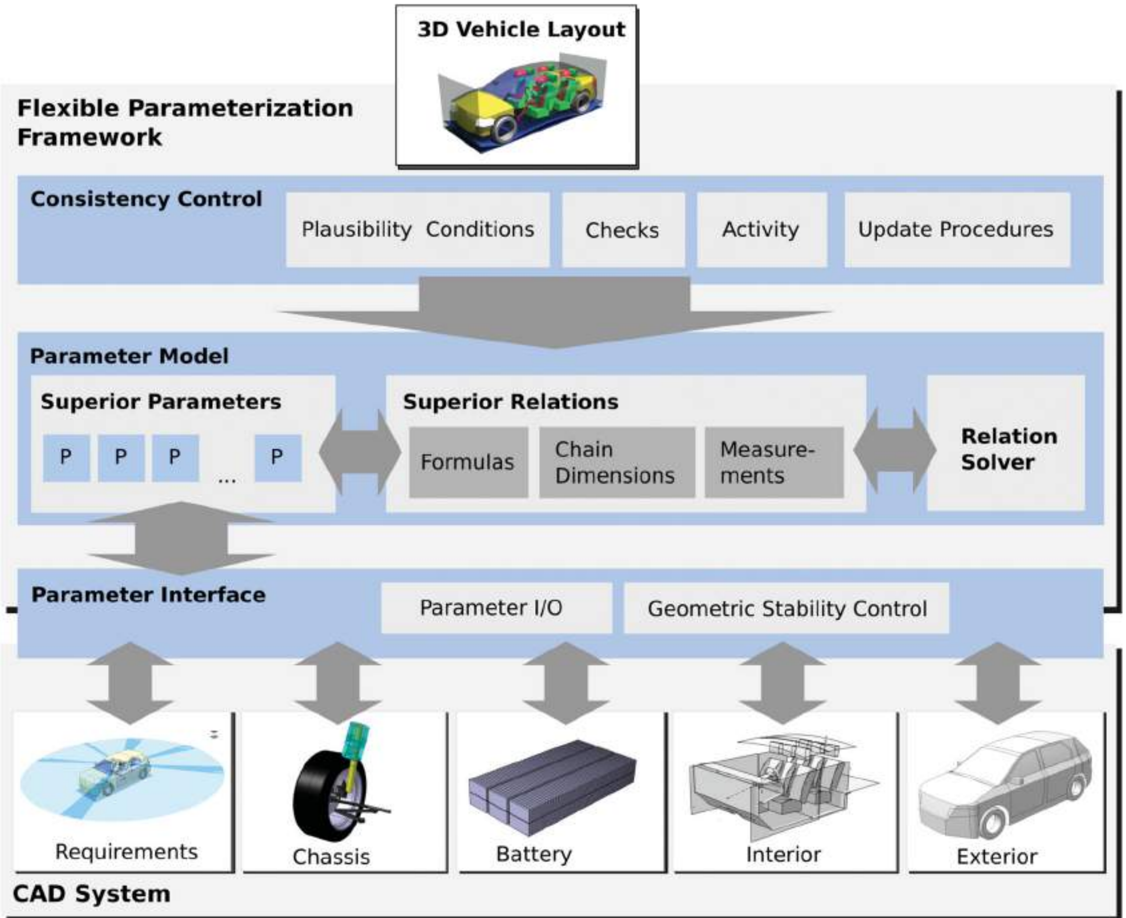

Figure 8. General structure of the flexible parameterization framework.

to preserve data integrity and logical consistency within the parameterization framework. The consistency control is responsible for the following tasks:

1. Preservation of the defined parameter value limits and tolerances (plausibility)
2. Check routines
3. Activation/Inactivation of parameters due to design variants or parameterization conflicts
4. Initiating update procedures when changing parameters or modification of relations

The preservation of parameter limits and values is important to the concept development, since many technical properties are relevant to homologation, which means the legislative compliance of the car. In order to assure it already in the initial concept phase, the respective parameters values have to remain within their legally defined value ranges.

The check routines represent functions which are more complex than simple verifications of parameter value limits. For instance, it could be necessary to trigger some action if both the mass and moment of inertia of the car exceed specified limits. An additional necessity for checks arises, due to possible incompleteness of defining conceptual data. For instance, a creation of a special loading plane is only possible, when all its necessary input parameters are defined. Until the definition is not complete, the geometric feature and all its dependent parameters have to remain inactive. Through this, both the plausibility and the extended check functionalities are directly connected to the activity routines, turning on/off required parameters. Geometric features are activated/inactivated through their associated activity parameters.

The update procedure is responsible for launching all necessary update workflows within the flexible parameterization framework (see also section 4.2). It initiates the transferring of parameters to the respective CAD modules and also manages conflicts which might occur due to conflicting definitions out of different parametric datasets. For instance, if a certain parameter is to be defined as a control parameter, it must not be evaluated somewhere else in a parametric dataset otherwise circular definitions would result. In this case the flexible parameterization framework would either try to modify all inflicting relations and parameters or would ask the user what to do.

4.1.2. Parameter model

One major challenge in virtual automotive layout development is to get a clear picture of the parametric situation within the layout model. This includes traditional relational illustrations like existing parent-children-relations, but also the display of useful meta-information like required target values, descriptive comments, definition sources and the priority of bindingness in the context of the respective vehicle project. In the present approach, a parameter is considered as a value, enriched by the following meta- information (see Fig. 9).

The attribute *Target-Value* has informational character and refers to defined project target of the specific parameter. The attribute *Concept-Value* is the actual parameter value. In order to ensure proper validity within geometrical and project bounds, it has two additional boundary properties – Limit and Project Limit. Limit refers to a general geometrical consideration of a value. For instance, a wheelbase will always be a positive value, whereas a coordinate value may have a negative value, too. Project limit defines a project context specific boundary, which is allowed to be exceeded in comparison to the general limit. The attribute *Activity* defines whether the parameter is active or not. This is required since not all input data for a concept may be defined, which leads to immediate inactivation of the parameter. The modification status defines whether the parameter can be defined by user or is set by the flexible parameterization framework itself. As long as the parameter is evaluated by any relation, it is locked for modification automatically. The last attribute *Bindingness* refers to the project specific bindingness of the parameter. For example, a parameter, representing a value resulting from a legislative requirement, is always locked, since it has to be fulfilled anyway because it is necessary for the legal homologation of the car.

In context of the presented approach, a parameter value can have different sources within the parametric layout model (see Fig. 10), which is afflicting modification status. Parameter values can be obtained through direct modification by user, from measurements within the model, defined mathematical relations or result from a chain dimension.

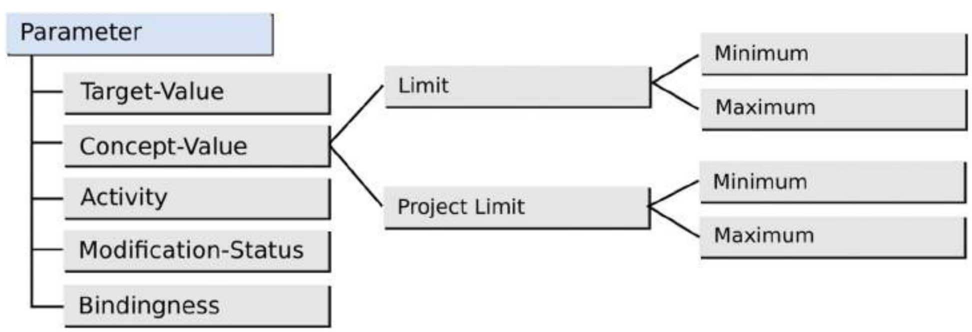

Figure 9. Properties of a parameter.

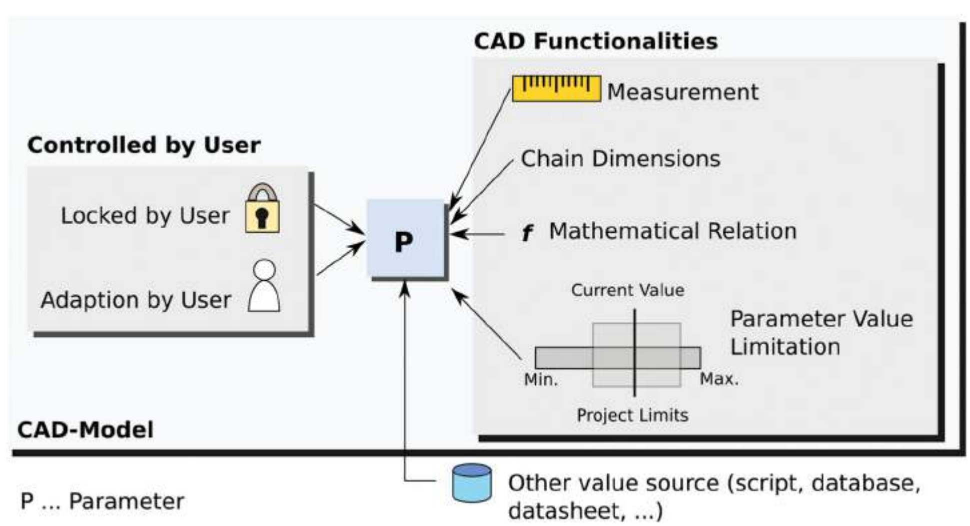

Figure 10. General sources for parameter values.

Relations are classified in general mathematical relations (formulas), chain dimensions and general input/output relations. Formulas represent mathematical relations between parameters of any kind that the applied CAD system supports. Formulas are not formable and therefore they are used for the representation of correlations intended to have a fixed character, or that do not represent a linear equation like chain dimensions. Chain dimensions are special relations, which possess the property of formability in comparison to "rigid" formulas that do not have this feature. The property of formability is restricted to linear equations only because of the mathematical nature of chain dimensions. Measurements are relations, which are connecting a geometrical or geometry based property value to a parameter. This could be for instance a volume, mass, center of gravity, or any other property provided by the applied CAD system.

4.1.3. Parameter interface

The parameter interface is responsible for synchronizing the parameters of the flexible parameterization framework with the ones in the respective parametric CAD datasets. The synchronization includes the update of values, but also the generation or removal of missing or obsolete parameters in the connected parametric datasets.

4.2. Established workflows in the flexible parameterization framework

A key to success in the efficient application of an automation-supported parameterization strategy is implied in the definition of proper change workflows, which requires parametric consistency at any time. Fig. 11 demonstrates the general workflows during parameter modification within the parameterization framework.

Initial check for the workflow is the validation of parameter boundaries. If a value is exceeding defined primary limits, it is immediately set back. If the value is lower than the minimum limit, the value is set to minimum; in case of exceeding the maximum limit, the maximum value is set. Concerning project limits, the user receives a warning that the limits are exceeded without further action from the parameterization framework.

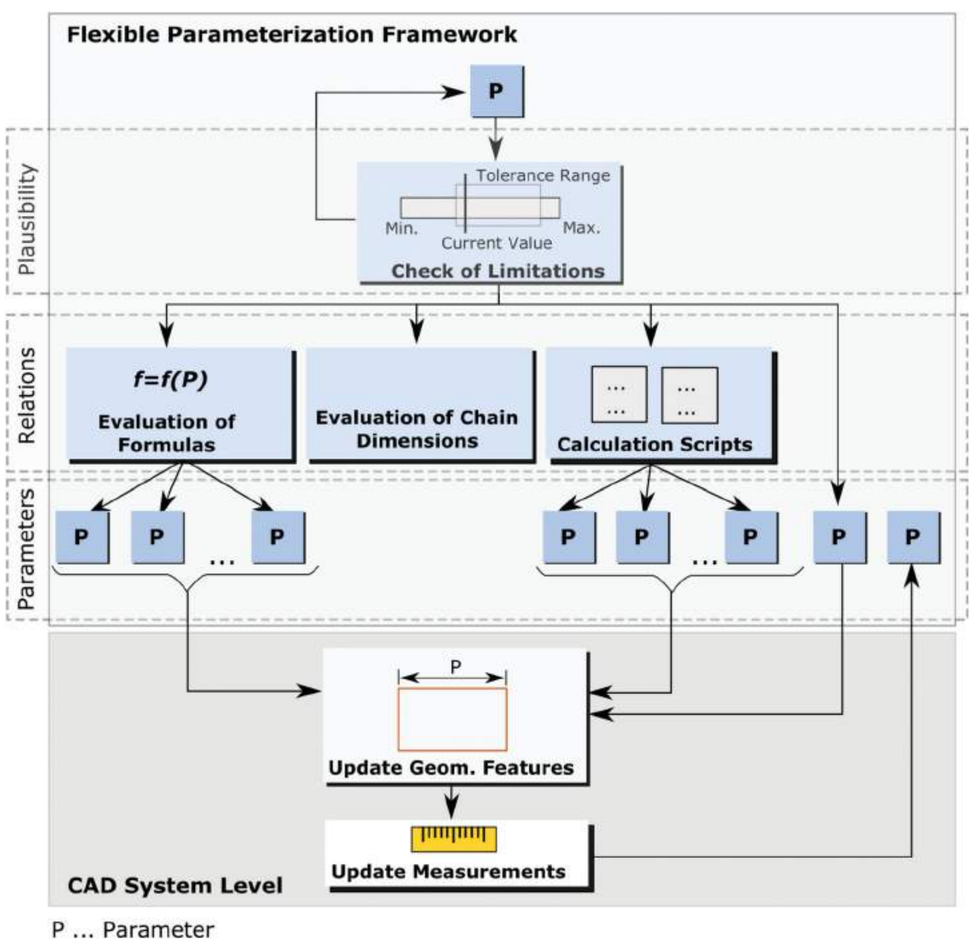

Figure 11. Change workflow for parameter modification within the parameterization framework.

After successfully passing the initial plausibility test the framework relations are being evaluated. This includes the update of formulas, chain dimension and the execution of more complex calculations, which can reside in separate scripted functions. In either case, updated parameter values are being calculated which are transferred by the parameter interface to the corresponding CAD modules. Since there is no guarantee that the embedded models in the framework can be updated correctly with respect to the resulting parameter values, the update process of the CAD system is being replaced by a script based update of the parameterization framework (see Fig. 12).

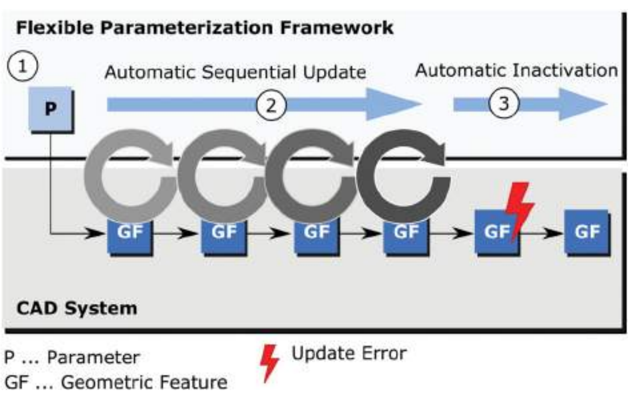

Figure 12. Automatic update workflow due to stability control of the parameterization framework.

In the present approach, the update procedure is performing the geometry actualization step by step. It uses the ability of CAD systems of refreshing single geometric features by the API. This means that the procedure is updating every single geometric feature (see Fig. 12), which is occurring along the geometric feature path, separately. As soon as an error occurs, all dependent geometric features are inactivated automatically. By this, the user receives a direct graphical and logical response of the current parameter constellation in the model. The following example should demonstrate the behavior:

In conceptual development proportional models are used to analyze spatial requirements. These models are controlled by parameters. As such models can get quire complex the update behavior cannot be predicted for every parameter constellation, especially when using geometric features like trims. Therefore, Fig. 13 shows an exterior proportional model with continuously decreasing chassis width (W116). If the value for the chassis width is getting too small, the required trims would cause an update error when using the regular update function of the used CAD system. Experienced users would maybe able to activate/inactivate the proper model

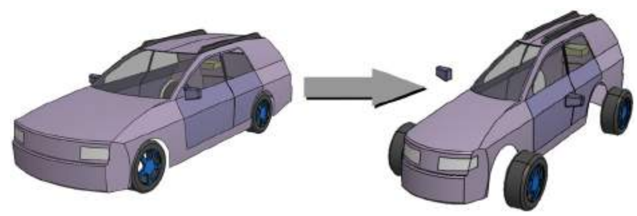

Figure 13. Degeneration of geometry due to unsuited parameter constellations.

areas on their own, whereas unexperienced users or such which do not use the dataset every day might not. In this case the parameterization framework is inactivating automatically all dependent geometric areas and also checks linked datasets if the geometric stability could be ensured. As long as a stable solution is not possible, the inflicting features remain inactive. As soon as a stable result can be generated, the features are activated by the parameterization framework.

4.3. Integration of the flexible parameterization framework into a CAD system

The functions provided by the API of CAD systems vary significantly. Because of this, the presented approach states the following requirements on the API of the used CAD system:

1. Creation/Modification of parameter objects
2. Creation/Modification of relation objects
3. Access to feature update functionalities

All other required functions are taken over by the parameterization framework itself. These requirements stem from the necessity of transferring parameter values to parametric CAD models. The demand for controlling relations is necessary, since existing parametric models and their internal parameterization should be continued to use.

The software components of the parameterization framework include the required routines, mentioned in section 4.1 and the provision specialized user-interfaces (see also section 4.4) in order to use the parameterization framework functionalities within the respective CAD system.

The advantage of this procedure is that the originating parametric CAD functions of the software can still be used, while the parameterization framework is managing the top-level parameters and relations. The separate parametric CAD datasets are collected within a corresponding master model, which is known to the parameterization framework and works as an interface to the

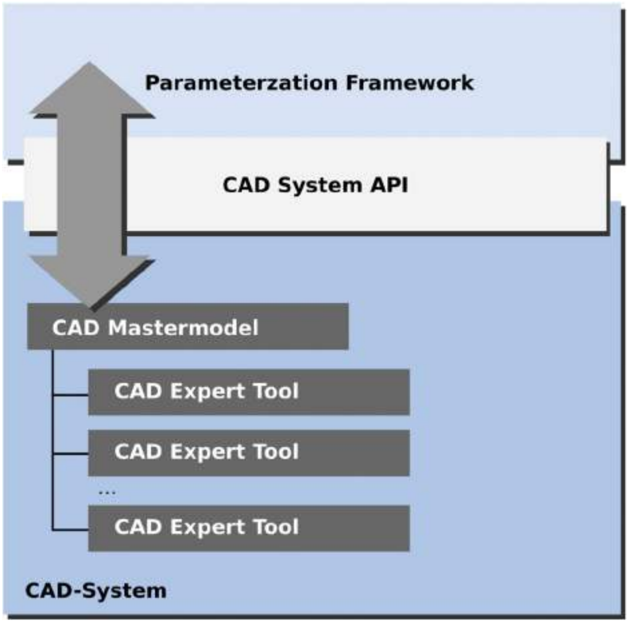

Figure 14. Integration of the parameterization framework into a CAD system by API.

implemented datasets (see Fig. 14). Through this, existing CAD modules can be integrated into the parameterization framework with very little effort. By target oriented use of findings of approaches like [12], a parameterization framework can be enhanced additionally in order to be independent of the API of the authoring systems at all.

4.4. *Usage of the flexible parameterization framework*

Within the outlined approach, modification of parameters and relations are performed by the flexible parameterization framework. In order to provide a user-friendly application, the various functionalities of the flexible parameterization framework are accessed by using graphical user interfaces (see Fig. 15).

This allows modifying existing relations and parameter values and supports the definition of new parameters and relations in context of the used CAD models. Through this, the compliance to data, parameterization and process standards can be ensured at any time. Furthermore, the user interfaces are based on design workflows which allows the guidance of the user. This enables also non experienced users to work with the flexible parameterization framework efficiently.

4.5. *Extension of parameters and relations*

A general possible disadvantage of parametric-associative models results from the maintenance point of view. Since not all parametric-associative models are carefully planned and structured according to standardized specifications, there is always a specific know-how required, especially when administration of the model is not performed by the initial creator. The present approach is seeking to avoid this situation by excluding the user from the entire parameterization process as described in the previous sections. Through usage of script-based extensibility functions, the model parameterization is always corresponding to defined standards, which ensures a consistent model quality, and as a further consequence the required process integrity. In this way, flexibility and model stability ensures future model use and satisfying new demands.

5. Application example

In order get a better understanding of the flexible parameterization framework, the following example should demonstrate the basic intention. The example is assuming, that there is a layout model dataset, which contains some basic equations to calculate essential dimensions of the car. These include the vehicle length (L103) and the wheelbase (L101). The initial condition of the layout model should be:

$$\begin{aligned} L103 &= L104 + L101 + L105 \\ L101 &= x_{RearWheel} - x_{FrontWheel} \end{aligned} \tag{5.1}$$

By this the entire vehicle length is evaluated by summing the overhang front (L104), the wheelbase (L101) and the overhang rear (L105). The wheelbase is defined by the difference of the wheel center x-coordinates.

The first step is to integrate the parametric dataset into the CAD master model (see Fig. 16). This step is carried out by using the built in functionalities for assemblies of the respective CAD system. The present example is using CATIA V5 but the idea can be applied to any other CAD system which fulfills the stated requirements.

When starting the flexible parameterization framework, it detects that there is a new part and reads its parametric information (see Fig. 17).

At this point the flexible parameterization framework knows, that there are 6 parameters, whereas L101 and L103 are evaluated by a formula (see Fig. 17). Since the flexible parameterization framework does not know, if the relations represent an equation or a formable chain dimension, the user has to define the type of relation when importing it into the flexible parameterization framework (see Fig. 18).

By defining both the wheelbase and the vehicle length as chain dimensions, the flexible parameterization framework creates 2 new chain dimension objects. Since they

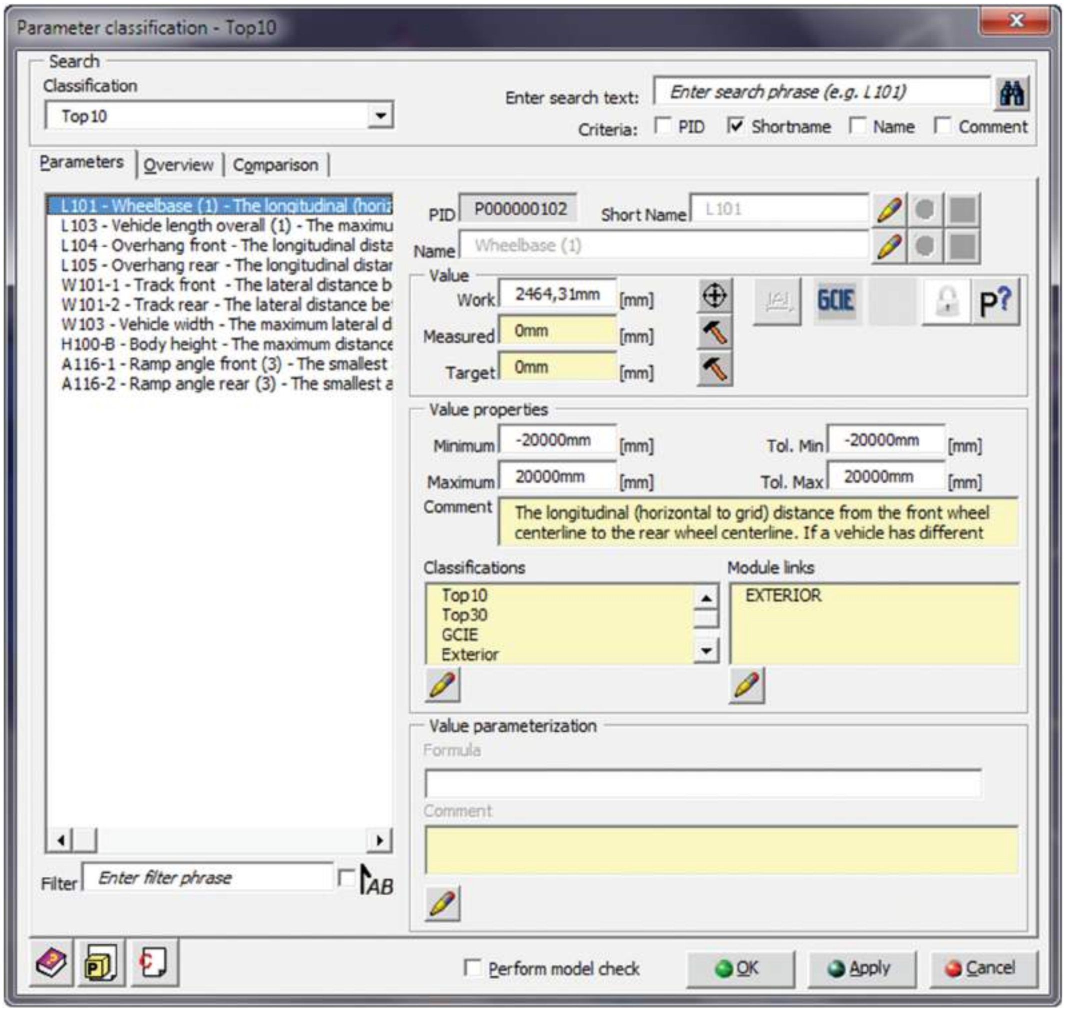

Figure 15. Parameter control by automation based user interfaces, exemplary interface.

Figure 16. Layout model within the master model.

are now controlled henceforth by the flexible parameterization framework, the corresponding CAD relations are being inactivated. The added chain dimensions are now available within the chain dimensions dialog of the flexible parameterization framework (see Fig. 19).

If the behavior of the chain dimension should be changed, the user has to select the desired behavior variant. In the present example the vehicle length (L103) should be changed from an evaluated parameter to controlling parameter. This leads to a conflict, since the vehicle overhang front, the wheelbase and the vehicle overhang rear are already control parameters to the vehicle length, which makes the equation over-determined. The flexible parameterization framework detects this conflict and provides possible solutions for solving the problem. In the present example, the program would suggest to change either L104, L101 or L105 into an evaluating parameter. If the wheelbase would be selected as evaluating parameter, the same type of conflict would arise, since it is evaluated by the difference of the wheel center x-coordinates. Again the system would suggest a possible solution for the situation: Either the wheel center front x-coordinate or the wheel center rear x-coordinate has to be changed into an evaluating parameter. Therefore, the cascading relation check process ensures a proper definition of all chain dimensions within the flexible parameterization framework at all times.

6. Conclusion

Main objective of the automotive concept phase is to determine those design variants amongst various possible solutions, which promise to be the best possible

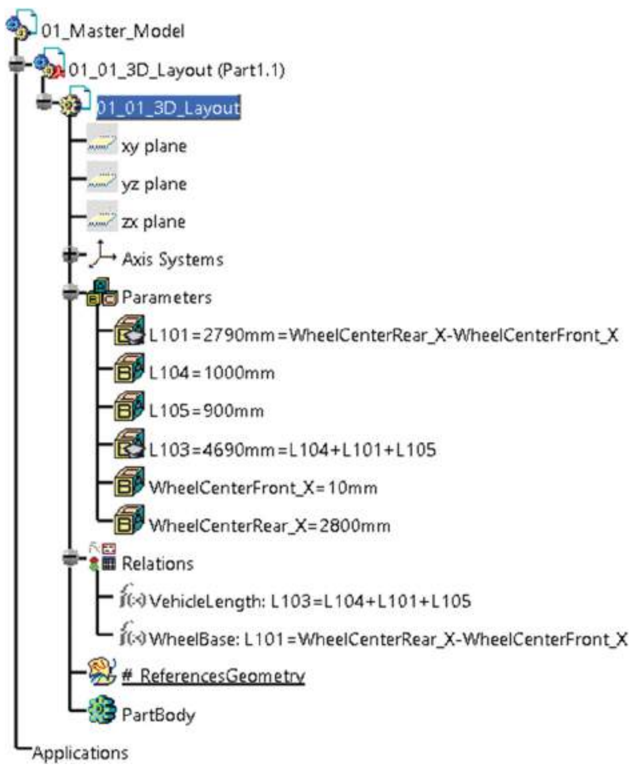

Figure 17. Existing relations within the layout dataset.

Figure 19. Chain dimensions control of the flexible parameterization framework.

compromise in context of required vehicle functions and properties. This is impeded by the fact, that in early stages of product development the required vehicle functions and geometric models are often not fully defined, which leads to high geometric and functional uncertainty. In addition, boundary conditions of the vehicle project may change rapidly within the concept phase, since car manufacturers are forced to react on new

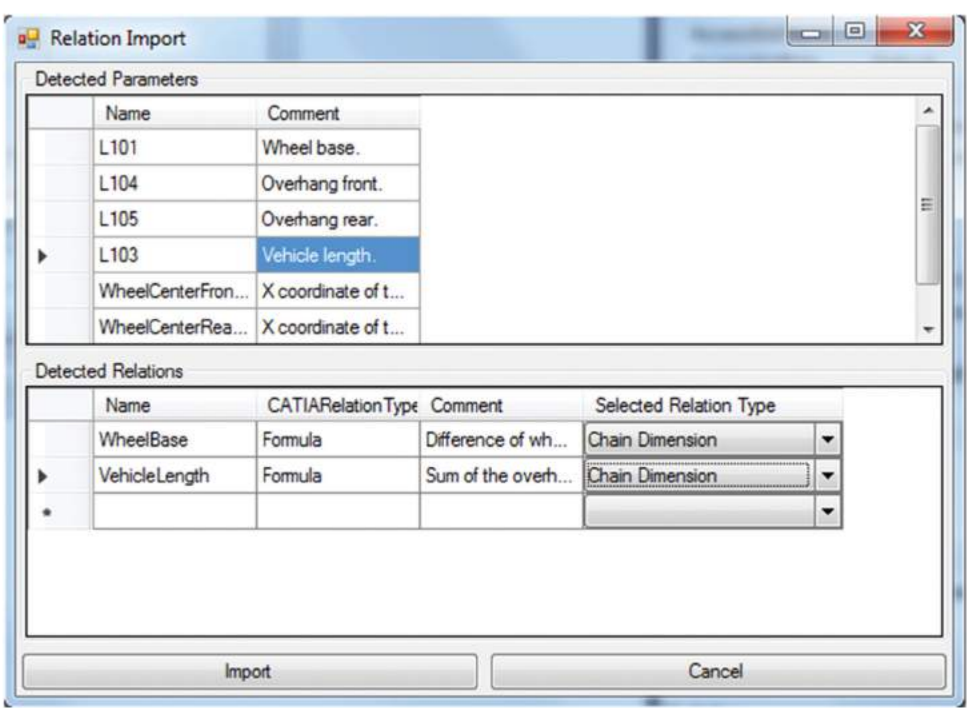

Figure 18. Relation import dialog of the flexible parameterization framework.

developments on the global market. Because of these reasons, parametric-associative models are applied to quickly generate conceptual 3D models, to evaluate vehicle functions and resulting properties and to choose optimal design solutions.

In this context, the present paper introduces an approach for supporting automotive architectural layout definition – the flexible parameterization framework. It integrates advantages of traditional parametric-associative design and relational flexibility at the same time. Automation interfaces of CAD software are used to implement algorithms, which form geometrical and functional relations according to current project requirements, and enable consistent integration of existing expert tools. Through the flexible and open character of the approach, an extendibility for future adaptions is being provided. Consistent development processes are supported by reproducible analyses within the layout model and efficient maintenance. The approach shows that the application of CAD-based automation methods offers a great potential for enhancing flexible parameterization strategies in automotive 3D vehicle layout processes.

ORCID

Patrick Rossbacher http://orcid.org/0000-0002-9184-1870
Mario Hirz http://orcid.org/0000-0002-4502-4255

References

[1] Böhme, M.: Ein methodischer Ansatz zur parametrischen Produktmodellierung in der Fahrzeugentwicklung, PhD thesis, TU-Berlin, Germany, 2004, VDI Fortschritt-Bericht, ISBN 3-18-356712-1.

[2] Forsen, J.: Ein systematischer Ansatz zur methodisch parametrisch-assoziativen Konstruktion am Beispiel von Karosseriebauteilen, PhD thesis, Universität Duisburg-Essen, Shaker Verlag, Aachen, 2003, ISBN 3-8322-1862-9.

[3] Gessner, K.: Package-Features für die Kommunikation in den frühen Phasen der Automobilentwicklung, PhD thesis, TU-Berlin, 2001.

[4] Göber, T.; Hirz, M.; Krammer, S.: Method for Externally Controlled Parameterized Automotive Design at an Initial Development Stage, Conference Virtual Product Development in automotive Engineering, Lake Chiemsee, Germany, 2006.

[5] Göber, T.; Hirz, M.; Krammer, S.; Pollheimer, D.: Conceptual automotive design and digital mock-up methods supported by external parameter control, SAE World Congress, 2007, Detroit, USA.

[6] Hirz, M.; Hirschberg, W.; Dietrich, W.: Integrated 3D-CAD Design Methods in Early Automotive Development Processes, International Federation of Automotive Engineering (FISITA) World Congress, 2008, Munich, Germany.

[7] McMahon, C.; Salehi, V.: Development and Evaluation of an Integrated Approach for Parametric Associative Design in an Industrial Context, Computer Aided Design, 7(a), 2010, 7–14. http://dx.doi.org/10.3722/cadaps.2010.07-14

[8] Niemirski, S.: Parametergesteuerte Karosserie- Generierung im PKW-Vorentwurf, TU-Berlin, PhD thesis, 1988.

[9] Rasenack, W.: Parametervariationen als Hilfsmittel bei der Entwicklung des Fahrzeug-Package, PhD thesis, TU-Berlin, 1998.

[10] Rossbacher, P.; Hirz, M.; Dietrich, W.: 3D-CAD parametric design strategies with interlinked CAE reference object creation for the overall vehicle layout optimization in the early auto-motive concept phase, International Federation of Automotive Engineering (FISITA) World Congress, 2010, Budapest, Hungary.

[11] Rossbacher, P.; Hirz, M.; Harrich, A.; Dietrich W. et al.: The Potential of 3D-CAD Based Process – Optimization in the Automotive Concept Phase, SAE Int. J. Mater. Manf. 2(1), 2009, 250–257. http://dx.doi.org/10.4271/2009-01-0475

[12] Salchner, S.; Stadler, S.; Hirz, M. et al: Multi-CAD Approach in Knowledge-Based Design, Computer Aided Design, 2015, 243–247. http://dx.doi.org/10.14733/cadconfP.2015.243-247

[13] VDI Richtlinie 2209, Verein Deutscher Ingenieure: 3D product modelling, Düsseldorf, VDI-Verlag, 2006, pp. 44.