# CAD-Automation in Automotive Development – Potentials, Limits and Challenges

Alexander Kreis[1], Mario Hirz[2] and Patrick Rossbacher[3]

[1]Graz University of Technology, alexander.kreis@tugraz.at
[2]Graz University of Technology, mario.hirz@tugraz.at
[3]Magna Steyr Fahrzeugtechnik AG & Co KG, patrick.rossbacher@magna.com

Corresponding author: Alexander Kreis, alexander.kreis@tugraz.at

**Abstract.** Computer-aided product development has become indispensable in many branches of industry, with the discipline of computer-aided design in particular playing a crucial role in the development processes of automotive products. Due to a multitude of changing boundary conditions and requirements (e.g. cost reduction, shorter cycle time for development and manufacturing) automotive development processes have to be enhanced. In order to be able to counteract the constantly changing requirements, a further development in the field of computer-aided design must take place. This further development is based on knowledge which is implemented as automation procedures into existing CAD systems. In this context, this paper gives insights into the CAx disciplines used in automotive development processes, lists different levels of knowledge-based automation methods and discusses the potentials, limits and challenges of CAD-automation. Furthermore, selected use-cases of knowledge-based CAD-automation in automotive projects are shown.

**Keywords:** CAD-Automation, Knowledge-Based Engineering, Generic Programming, Process Optimization, Machine Learning
**DOI:** https://doi.org/10.14733/cadaps.2021.849-863

## 1 INTRODUCTION

The automotive industry is subject of continuous changes, which influence – among others – the vehicle development and production processes. To maintain competitiveness, growing requirements on the development and production of cars have to be counteracted with appropriate measures, which include technical aspects, but also consider social factors, the highly iterative nature of the design process as a whole, and cultural differences between disciplines such as industrial design, mechanical design and engineering analysis. In the past decades, development process-related improvements have been influenced by systematic approaches, such as simultaneous engineering, frontloading and the introduction of corresponding computational development methods [8], [11].

The resulting higher flexibility of development processes led to a reduction of process times and thus to an increase of development efficiency, as well as the ability to react more quickly to changing product requirements. Computer-aided design (CAD) plays a central role in car development. It is applied throughout the entire development processes and interlinked with a number of computational and non-computational disciplines. In this context, CAD-automation offers a high potential for further optimization of vehicle development and production. This paper introduces different types of computational engineering methods in the automotive industry and their interactions with CAD processes. Subsequently, the levels of CAD-automation in automotive development are explained and discussed in view of their potentials, limits and challenges. Finally, exemplary applications of CAD-automation are elaborated for specific development tasks in the automotive industry and evaluated regarding the benefits, potentials and efforts [11].

## 2 COMPUTER-AIDED ENGINEERING IN AUTOMOTIVE DEVELOPMENT

Virtual product development includes the application of computer-aided methods and tools and plays an increasing important role in the development of cars. Within virtual development processes, computer-aided design (CAD) is in a central position because the CAD models provide a comprehensive product description [8], [9]. This includes geometrical definitions, materials, manufacturing-related data, structural information in case of assembly groups as well as a number of features and data for product description [23].

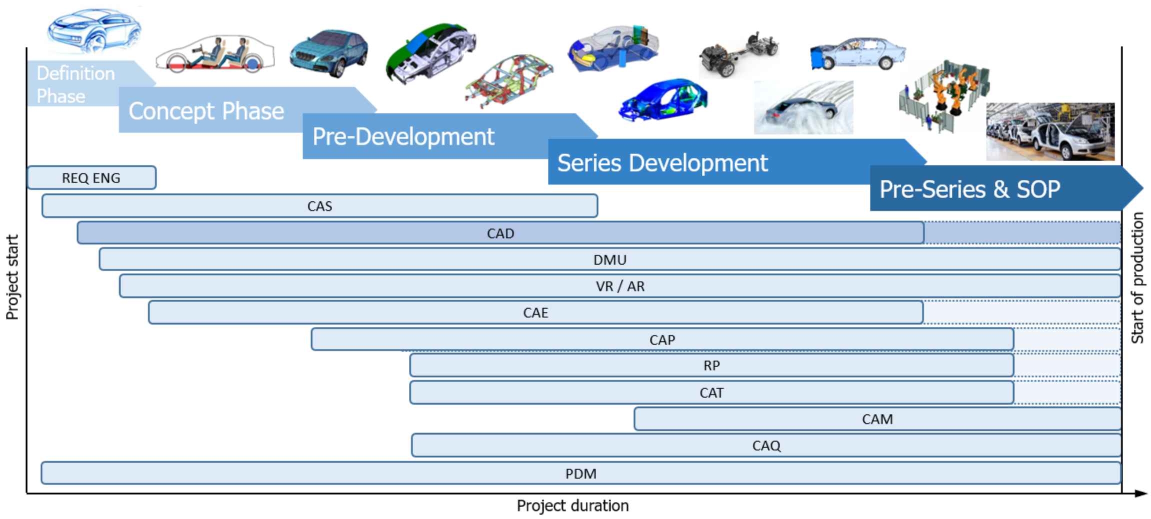


**Figure 1**: CAD and related computer-aided processes in automotive development.

Figure 1 shows an overview of CAD and related computer-aided applications in a general automotive development process. The generic process is divided into five main phases. In the *Definition Phase*, the specifications of the car to be developed are elaborated. This phase includes far reaching investigations of customer demands, legislative boundary conditions in target markets and the evaluation of technologies. The corresponding requirements engineering (REQ ENG) is performed based on multifarious data sources, e.g. benchmark data, specification lists, cost calculations, as well as legislative and technical documents. Vehicle layout studies are worked out as conceptual CAD models that support the development of vehicle requirements in a close cooperation with the other disciplines involved. This early phase also includes initial vehicle styling development by use of computer-aided styling (CAS) software. The created styling surfaces are incorporated with the CAD-based vehicle layout models throughout *Definition Phase*, *Concept Phase* and *Pre-Development Phase* to enable a close integration of styling and engineering development processes. Based on CAD models, assembly groups are established on different levels of the vehicle structure. This includes components and systems, e.g. bodywork and drivetrain, as well as the full-vehicle structure. Vehicle layout and packaging investigations are conducted by use of digital mock-up (DMU), which involves simplified geometry data that are derived from the CAD model structure. Besides in course of initial vehicle layout development, DMU is used in a wide range of other applications of the entire development process, e.g. for ergonomics development of driver and passengers, assembly simulation of components and modules, as well as for the simulation of vehicle assembling processes in manufacturing plants. DMU is closely related to virtual reality and augmented reality (VR / AR) processes, which enable a reality-near representation of vehicle exterior and interior surfaces as well as the integration of human-oriented simulation. In this way, VR / AR supports evaluation and optimization of vehicle styling, the development of user interfaces, control elements in the car as well as manipulation and actuation of workers during production processes [8], [12], [13].

Computer-aided engineering (CAE) comprises a large number of engineering-oriented simulation processes, including finite-elements (FEM), multi-body (MBS) and computational fluid dynamics simulation (CFD). Vehicle development very closely integrates CAD and CAE processes to enable an effective creation and optimization of components, modules and systems. In this way, data exchange between CAD and CAE represents an important task. In general, simplified geometry data are derived from CAD models and provided to CAE processes, whereby the different types of simulation require different kinds and ranges of data exchange. CAE processes are performed in a large scope of the vehicle development process, starting with conceptual simulations and ending up with very detailed verification of vehicle systems. Computer-aided production engineering (CAP) involves the development of manufacturing processes and the corresponding machines and tools. CAP requires a deep integration of supplier industry into its development processes, leading to a spreading of engineering tasks to the different involved departments and companies. Here, the CAD models deliver important data of the components and modules geometries as well as of their characteristics, materials and production-related information. Rapid prototyping (RP) is increasingly applied in modern development processes, e.g. additive manufacturing. In this way, RP enables time-efficient provision of components to support hardware-based testing and evaluation. Besides virtual development, these hardware-based investigations are applied on different levels of car development. This involves components, modules and system tests, e.g. functional tests of bodywork components, comprehensive investigations of powertrain systems, crash tests on full-vehicle level. CAD data deliver geometry as well as feature-oriented information for the setup of the testing facilities as well as for processing measurement and test data [8], [12], [13].

In the *Series Development Phase*, computer-aided manufacturing development plays an important role. This includes the development of manufacturing facilities, logistics and the integration of delivered components and modules from different suppliers. Exemplary, the automotive bodywork manufacturing line includes sheet metal manufacturing, extensive joining technology processes, corrosion protection and paint shop as well as the assembly of body components, e.g. car doors, flaps and window glasses. In the *Pre-Series Phase*, all manufacturing processes are optimized to enable an effective *Start of Production* (SOP). Finally, computer-aided quality engineering (CAQ) is integrated into the different subsequently or parallel performed development processes to ensure the fulfillment of quality requirements in view of both product and manufacturing development. Besides process-oriented data, CAQ relies on a number of product-related information that is provided by the CAD data structures. The large amount of different types, formats and versions of data occurring in automotive development processes is maintained by comprehensive product data management platforms (PDM) [8], [12], [13].

Figure 2 shows the connections between CAD and the mentioned computer-aided engineering processes. The CAD models deliver information about geometry, product features, design history as well as product structure-related information, e.g. in case of assembly groups. These data are provided to the adjacent engineering processes, which are parallel or subsequently performed. Depending on the specific engineering discipline, different types of information are derived from the native CAD models. In most cases, data exchange processes are conducted unidirectional from the CAD model to the corresponding engineering process. Nevertheless, in specific processes, bidirectional data exchange is established, e.g. in case of styling development (CAS), where styling geometry information is delivered by CAS and imported as rigid geometry models into the CAD environment to serve as boundary data for engineering-related development tasks [20]. Bidirectional exchange of detailed geometry data comes to use in case of multi-CAD environments, e.g. in the course of data exchange between different companies [8], [12], [13], [18].

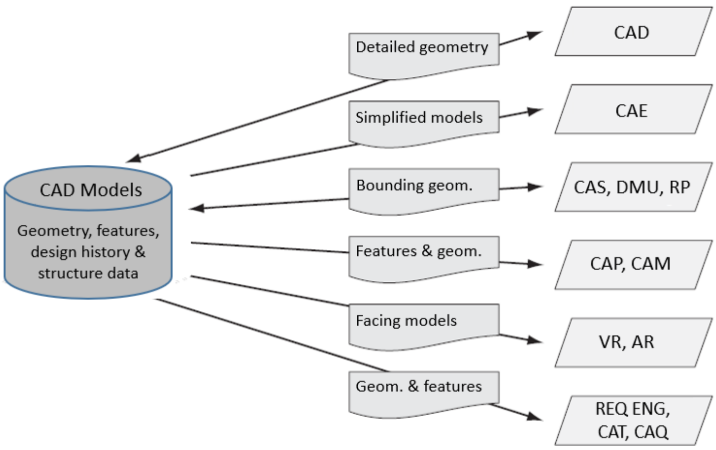


**Figure 2**: CAD models as a central data basis for different computer-aided development processes.

## 3 CAD-AUTOMATION

CAD-automation comprises creation, optimization and maintenance of CAD data, as well as the integration of CAD processes and other parallel or subsequently performed processes (c.f. Figure 1). In addition, CAD-automation involves data exchange between different engineering disciplines (c.f. Figure 2) and the integration of automated procedures into the company and project specific PDM systems. Figure 3 gives an overview of the different levels of CAD-automation, their complexity and indicates the corresponding creation and maintenance effort. In this context, a number of techniques is aligned to the different levels.

Modern CAD systems applied in the automotive industry offer feature-based and parametric methods for the development of CAD models. The so-called parametric-associative design methodology comprises semantics, algorithms and relations for an effective creation of the different types of CAD models and assembly groups [17]. In course of data exchange with neutral geometry data formats, e.g. IGES, STEP, rigid CAD models come to use.

As a sub-discipline of knowledge-based engineering, knowledge-based CAD provides a systematic integration of knowledge into the design models by use of parametric-associative and feature-based design with extension of problem-specific solutions, e.g. in form of template models. In this way, it integrates specific knowledge about the product, development procedures and production-related information into the design process. Knowledge-based design uses automated routines (e.g. script-based geometry creation) and implements functions, rules and reactions into the CAD model. Template models include different types of algorithms and features to automated support the creation of specific models. Templates are often based on the re-use of variable geometry models in combination with library functionalities. In this way, knowledge-based design supports an increase of efficiency in the design process by periodic use of design knowledge in combination with semi-automated geometry creation procedures and thus represents an important approach in view of CAD-automation. Examples of knowledge carrier in CAD are:

- Sketches
- Parameter structures
- Relations, rules, formulas
- User-defined features
- Model design histories
- Template models
- Attributes and annotations
- Product manufacturing information (PMI) and geometric dimensions and tolerances (GD&T)
- Structures of assembly groups
- DMU-functionalities (e.g. kinematic simulation)

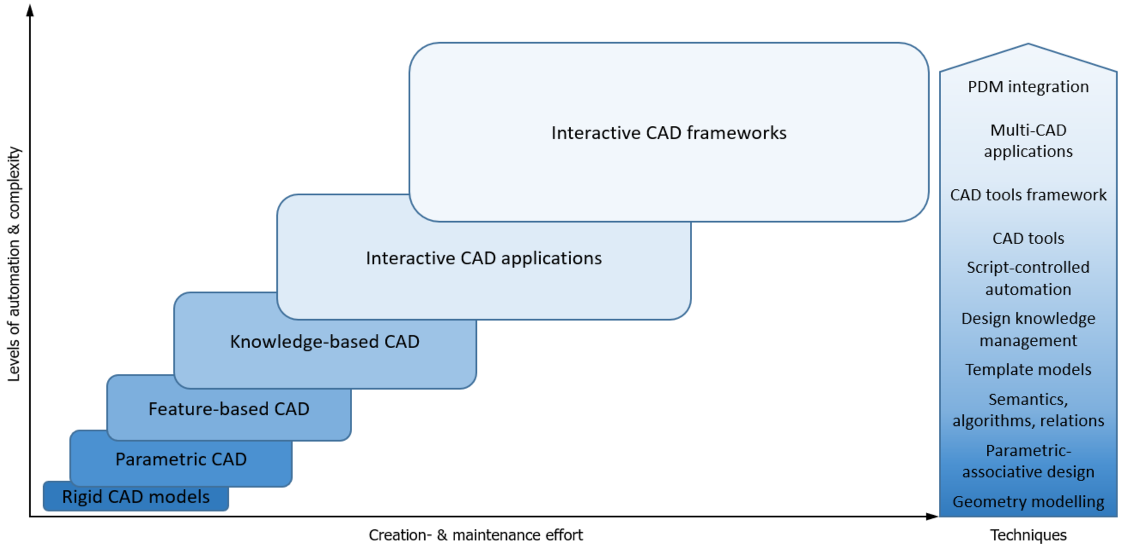


**Figure 3**: Levels of CAD-automation and related design techniques.

Interactive CAD applications represent functional software modules that are embedded into the CAD environment. They support complex and often recurrent engineering tasks effectively by an integration of geometry creation as well as different types of calculation and simulations procedures. Automated geometry creation is accomplished by a combination and automation of the CAD-internal functionalities. In case of complex simulation tasks or the involvement of large data structures into the computation procedures, different types of data sources are integrated, e.g. libraries, CAD-external simulation software [5], [21]. In typical applications, users are guided by graphical user interfaces (GUI) as well as supporting and help functions. In this way, interactive CAD applications integrate methods of knowledge-based CAD, defined working respectively computation sequences and user guidance. Typically, these tools are written in application-oriented program languages, e.g. Visual Basic for Applications (VBA), Visual Basic.NET (VB.NET). The creation of interactive CAD application is similar to software development in general and consequently requires both know-how of the engineering task as well as software programming expertise. In addition, software roll-out, update management and maintenance effort must be considered.

As part of superordinate software tool structures, interactive CAD frameworks integrate different types of engineering design-related applications. In many cases, the frameworks are integrated into existing PDM software architectures, which requires expertise and effort for successful provision of the solutions. The frameworks are defined according to the requirements of a specific company and provide a set of tools that effectively support development projects. In the automotive industry, both car manufacturer and (large) engineering and system suppliers have established their company-specific development tool landscape, which incorporate interactive CAD frameworks as one important section. In a multi-CAD approach, interactive CAD frameworks are able to integrate different CAD software environments in view of model exchange and collaborative development. In addition, multi-domain applications are provided by docking or integration of CAD-external simulation tools to the CAD-related engineering tasks (c.f. Figure 1). Interactive CAD applications represent a comprehensive approach for creation, provision and maintenance of CAD-based engineering landscape. Due to the large extend and the close incorporation into company-specific development processes, PDM integration is mandatory. In addition, interactive tool frameworks require professional setup, development as well as maintenance and service.

### 3.1 Functional CAD-Templates

In the upper levels, starting with knowledge-based CAD, functional templates represent a main component of CAD-automation. Embedded into the corresponding CAD software and development environment, functional templates are used to conduct automated procedures in product design. This includes the creation, adaptation and optimization of geometry models, the implementation of computation and simulation procedures, as well as support of data conversion and management.

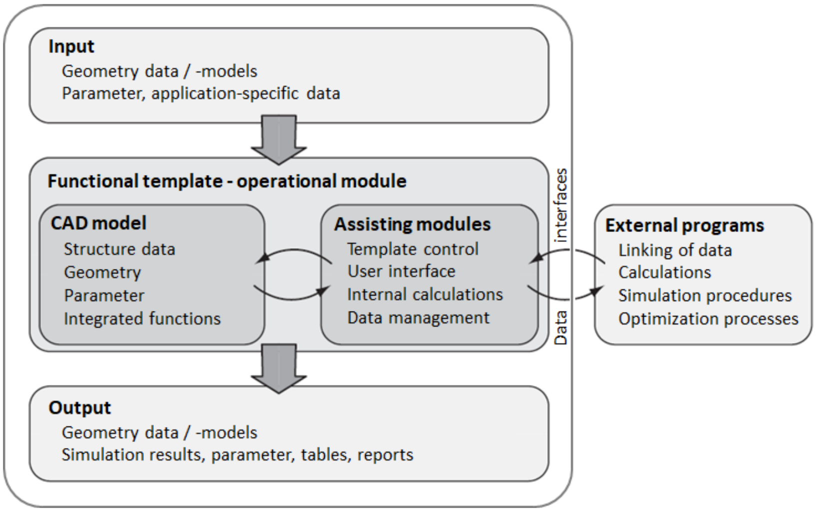


**Figure 4**: Exemplary layout of functional templates, according to [16].

Figure 4 shows the general layout of functional templates. Input data and boundary conditions are defined in the *Input* section. This includes geometry data, different types of input parameters and further application-specific data which are required to perform the computation task. In many cases, data input is conducted by graphical user interfaces, which guide the engineers through the operational sequences of the functional template. In addition, input data are delivered by embedded or connected data sources, e.g. the PDM systems, external data bases. The *Operational module* represents the main unit of functional templates that performs the automated processes of geometry creation, modification and optimization as well as the implementation of externally delivered data structures. This includes automated processing of model structures, geometry, parameter and the execution of CAD-integrated functions. As integrated software application within the standard CAD tools, functional templates are equipped with *Assisting modules* that enable stable and user-friendly operation. This includes template control, user interfaces and internal calculations, e.g. for processing of input data. Finally, a data management module supports proper internal flow of data and an effective integration of external programs via data interfaces. The *Output* section provides the resulting data to the users and for subsequently performed design processes. This includes geometry data as well as simulation results, parameters, tables and reports. Functional templates are integrated in different types of knowledge-based CAD applications, as well as interactive CAD-based solutions and frameworks. Independent from the actual application, the general workflow consists of four phases:

- Phase 1 with the provision of input data. In this phase, the required information is delivered manually by the operator and / or automatically by connected data sources within the pre-defined routines.
- Phase 2 includes the CAD-based functional layout procedures according to the implemented computation algorithms and procedures.
- Phase 3 includes automated geometry creation, integration and control. In addition, geometry verification and stability management are implemented.
- The output data are created and made available in Phase 4, which also includes the presentation of geometry and the submission of data via pre-defined interfaces.

### 3.2 Artificial Intelligence in CAD-Automation

Artificial intelligence (AI) provides a large potential to solve complex problems. In recent years, the technology has been increasingly used in software development, computer graphics, object recognition and different types of optimization problems. The steadily improved performance makes AI also interesting for the application in mechanical product development today. As a subset of AI, machine learning (ML) uses algorithms and statistical models to solve problems without manual interactions by integration of explicit instructions relying on patterns and inference. In this context, deep learning – as a part of machine learning – uses artificial neuronal networks for the recognition of patterns and to interpret relevant data through machine perception, labeling and clustering. Neuronal networks represent the basis of AI software (c.f. Figure 5). They consist of a number of layers including neurons and connections. Each neuron receives, delivers and processes information, e.g. by use of data filtering and data comparison. The output data of neuronal networks represent suggested results that depend on the input data, layer and connection structure, as well as the training of the neuronal network. As a difference to traditional engineering methods, results delivered by AI do not directly base on physical or empirical models and they do not rely on the classical numerical solving approaches as known from engineering and product development [14].

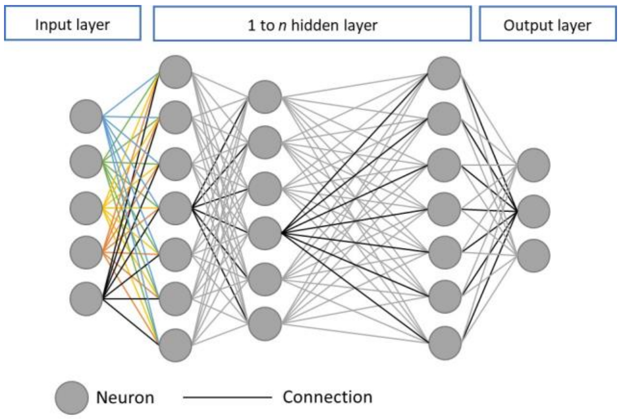


**Figure 5:** Structure of neural networks.

In view of CAD-automation, AI is not part of industrial applications yet. There are research activities, which implement AI conceptual into the development of mechanical products in both geometry creation and optimization, e.g. [15], and works that integrate deep learning for big data applications in CAD and PLM [4]. Figure 6 exemplary introduces the sequences of CAD-based deep learning. In the section data generation, training data are modeled and validated. This includes the provision of a variety of geometry models according to the specific boundary conditions and development targets. Because the success of AI application is significantly influenced by the quality of training, the provision of comprehensive and correct data represents an important aspect. Here, CAD-automation can be applied to generate a broad variety of geometry models for training purposes.

The second section, AI modeling, is related to the selection of a suitable AI software configuration and the definition of a model setup. The different available AI solutions are tailored for specific fields of application – in this way it is crucial to select the right algorithms. In opposite to areas with years of experience, e.g. object recognition, the selection of a proper AI solver represents a challenge for CAD applications because of the missing experience in this field. In this way, the considered AI solutions have to be verified regarding their suitability for the specific problem. Based on the generated data and the selected and verified AI software, training of the neural network is performed. This includes a list of data variations according to the training plan and the evaluation of performance indicators in view of system convergence. Finally, the AI is applied to the target problem to deliver solutions that subsequently are topic of further engineering processes. For that, the solution space is evaluated, e.g. by selective checks of results.

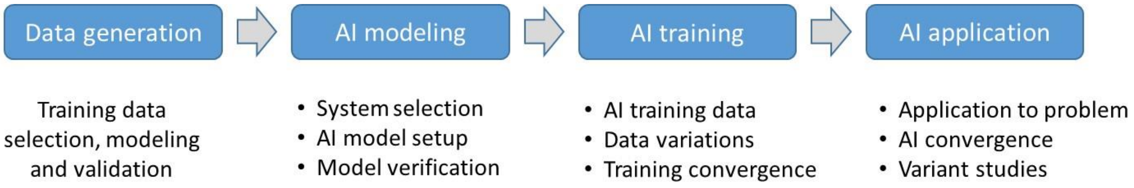


**Figure 6:** Sequences of CAD-based deep learning.

From the viewpoint of today's state-of-the-art, AI has potential so solve CAD-oriented problems with good approximation, but the technology is limited in view of a satisfying consideration of geometrical details. Exemplary, AI can be used to develop suggestions for the general shape of a geometrical object on a basis of pre-defined boundary conditions, but the detailed structure of the geometry cannot be developed sufficiently. In this way, the technology is applied to develop suggestions of solutions, which are evaluated and optimized by the engineers involved. Based on intensive research in this field and increasing computation capacities of the systems, an increasing share of application in mechanical product development is expected. Related to the structure of functional template models shown in Figure 4, AI tasks are performed as "external processes", with the results are fed back into the CAD environment to be evaluated by the engineers. In future, a closer integration of AI into the operational module of functional CAD-templates is expected.

## 4 POTENTIALS, LIMITS AND CHALLENGES OF CAD-AUTOMATION IN AUTOMOTIVE DEVELOPMENT PROCESSES

Automation in the field of CAD has increased significantly in recent years due to the high potential for process optimization. Especially in different types of development and production engineering processes, the use of CAD-automation can lead to a considerable added value, which is represented in savings of resources, costs and time. In course of different applications of CAD-automation, which involve exchange of data, standardized formats have been developed during past years, e.g. [10], [22]. Furthermore, there is a high potential for the application of CAD-automation in engineering-related niche areas. This is of particular interest if there are no or only limited suitable off-the-shelf software tools and standardized data exchange formats available on the market. CAD-automation can also help to increase quality of development and production processes. This is enabled by the fact that the working principle of these tools is always based on their integrated functions, which enables parallel computing sequences. An example of resource savings and simultaneous increase of quality that can be achieved through the integration of CAD-automation represent automotive body development processes with a large share of recurring working steps and large amounts of data. Automotive bodies include a number of joining elements, e.g. up to 10.000 weld spots in a steel body structure. Data of these joining technology elements are managed in tables today, e.g. in MS Excel. Each of the 10.000 elements contains several parameters, resulting in a data set of over 100.000 entries. If these 100.000 entries had to be checked manually (e.g. by CAD engineers) due to quality control, this check would be very time-intensive and probably not error-free. If, however, CAD-automation software packages are used for this purpose, this recurring inspection process can be carried out with large data sets, without errors and with fewer resources.

Related to Figure 3, the choice of automation levels in a specific case depends mainly on the cost-benefit ratio. In this context, the efforts for creation and maintenance of the automation package must be taken into account for the expenses, which might lead to lower automation levels for specific applications. An example of how this can be achieved represents a multi-CAD approach in CAD-automation. If the majority of company-internal development processes are conducted in one CAD system, the effort required to create an automated multi-CAD-based environment must be compared with the potential benefits and it should be evaluated, if the use of standard conversion programs and neutral data formats is advantageous. A further limitation of CAD-automation can be attributed to the large CAD models commonly used in the automotive industry and the associated limitation due to the available computing performance. Automotive development involves a large number of different vehicle components (more than 10.000 parts [27]) and the complexity of the CAD models is increased by associated information (e.g. parameters, material, annotations). Of particular interest is, how the high amount of data can be handled effectively. Finally, software training and maintenance effort has to be taken into account, which can require considerable share of effort in a CAD-automation project. In this context, planning of CAD-automation should consider the cost-benefit ratio very carefully [8], [11], [24].

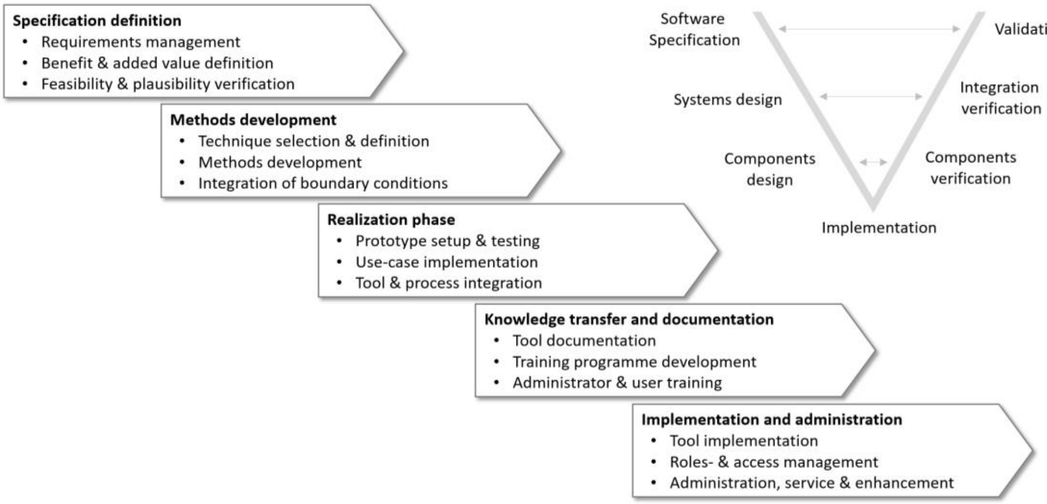


**Figure 7**: Lifecycle of CAD-automation in industrial environment and SW-development process according to the V-model.

The lifecycle of CAD-automation tools in industrial environment can be divided into five main areas (c.f. Figure 7). An important role plays the first section, *Specification definition*, including requirements definition and the investigation of potential benefits and added values. In this phase it is important to estimate the efforts for creation, implementation and maintenance of a new solution to be developed in comparison with already applied – mostly manually performed – processes. In this context, the feasibility and plausibility verification not only considers technology-related and data management-related aspects, but also takes under consideration economical viewpoints. In case of positive verification, the selection of suitable techniques to fulfill the prior specified characteristics is conducted in course of the *Methods development.* This includes the development of the key-functionalities under consideration and integration of different boundary conditions resulting in a prototype of the operational modules. In the *Realization phase*, the software packages are put together and tested according to pre-defined use-cases. Finally, the tool is integrated into the development environment (including PLM and CAD system integration) and prepared to be applied within the industry-specific development process landscape.

An important section represents *Knowledge transfer and documentation* with the target to disseminate the gathered improvements to the target groups. This requires effort for documentation as well as the development and execution of training programs for user and administration resources. Finally, *Implementation and administration* includes tool implementation in the development environment. In many cases, CAD-automation is performed within existing CAD software packages and PDM structure. In case of higher complexity, e.g. interactive CAD applications or frameworks, the implementation requires significantly more effort to fulfill the corresponding demands on tools integration, data management and tool usage. In any case, roles and access management must be defined accordingly and a plan for administration and CAD-automation tool service and enhancement has to be introduced.

The development of CAD-automation follows typical processes of software development, e.g. the shown V-model [26] (c.f. Figure 7). The V-model divides the process into two main areas: specification and design (left branch) and verification and testing (right branch). The different levels in each branch are connected by recursively performed optimization procedures, e.g. design of software functions and testing them later in the corresponding verification step. The bottom part of the V indicates the integration of software components. One challenge in the development of CAD-automation tools lies in the effective integration of the "top-view" processes, which are often sub-sequential stage-gate processes and the recursively designed development processes, which include different stages and repeating design and verification steps.

Out of gathered experience from several academic and industrial use-cases, the following factors should be considered for successful development and application of CAD-automation:

- Comprehensive and clear specification of the tool to be developed.
- Clear definition and verification of potential added value to existing design methods.
- Definition of a stable and available solution space of the CAD-automation.
- Transparent application and easy user-handling according to the specifications.
- Integration into development processes, software- and data management environment.
- Continuous maintenance and advancement.

## 5 EXEMPLARY APPLICATIONS OF CAD-AUTOMATION

The previously discussed potentials, limitations and challenges of CAD-automation are demonstrated by two examples of CAD-automation tools that have been developed in co-operation with automotive industry in the past years (named *Joining Converter* and *Stone Chip*). Both tools are embedded in the vehicle body development and manufacturing process with the target of resource savings and a faster time-to-market of automotive products.

## 5.1 CAD-Automation Package – Joining Converter

As described in the introduction, the automotive development process is divided into different main phases, in which several computer-aided systems are used. Exemplary, 3D modeling of the vehicle body components takes place in an appropriate CAD environment. The models include different types of information: geometry-related data (e.g. geometry and dimensions of metal sheets, center of gravity of the sheets) and joining technology-related metadata (e.g. type of joining technology, coordinates, joining technology-specific parameters, such as welding spot diameter, length of the screw). Both, geometry-related and joining technology-related data, are exchanged between the CAD source environment and several simulation environments (e.g. CAE, CAM) in the course of the development process. This exchange process manages both the required information for simulation tasks (e.g. fatigue strength, crash tests) and the feedback of simulation results to the CAD environment to be considered for design optimization.

Since there are no general and uniform standards for the data exchange of joining technology elements defined, joining technology source data can be available in a wide variety of formats. On the one hand, this includes diverse native CAD data formats (e.g. CATIA [1], NX [19], Creo [2]) as well as neutral data exchange formats (e.g. list-based (e.g. Excel), XML-based (e.g. xMCF [3])). With the target to improve the generation and management of joining technology data, a specific tool has been developed that offers an approach for data exchange between different sources and several computational engineering environments [12]. The tool *Joining Converter* provides a generic approach to process the different types of data and to convert them into a uniform data exchange format by means of suitable script-based automation processes. Figure 8 shows the integration of the tool into an exchange process of joining technology data from the source environment into the exemplary CAE target environment. The tool is designed in such a way that it either converts CAD-based joining technology data into a list-based or XML-based data format, or vice versa. Related to the classification in Figure 3, the tool can be defined as interactive CAD application, which is implemented into an interactive CAD framework of an automotive engineering company.

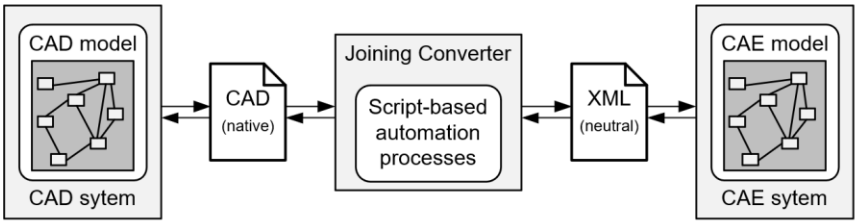


**Figure 8**: Integration of the tool *Joining Converter* in CAx data exchange.

In relation to the potentials and limits of CAD-automation, the tool supports both the data exchange process between different engineering environments and the entire development process of vehicle bodies. In addition to an increase in data quality, which is due to a lower error rate [11], [25], this also leads to savings of resources, costs and time. Especially the cost and time saving potential resulting from the integration of the automation package is of crucial importance for competitiveness and supports an earlier market entry of automotive products. Due to the increasing data quality the tool *Joining Converter* also contributes to increased quality of development and production processes. This in turn, leads to a lower failure rate of products. Referring to Figure 3, this tool is based on a multi-CAD approach, which has been chosen due to the high frequency of application in car body development. Since this tool has a high flexibility regarding the input sources (i.e. support of a large number of native and neutral data formats), it can be applied for a broad range of projects. Under economic aspects, the benefits justify the high efforts required to create, implement and maintain the automation package.

## 5.2 CAD-Automation Package – Stone Chip

Another example that supports the development process of automotive bodies represents the tool *Stone Chip*, which is used in the early phases of automotive development to simulate stone impacts that hit the car outer skin during driving (c.f. Figure 9). This tool is directly embedded in the CAD environment and can be operated by the CAD engineers through a provided GUI. The software package makes it possible to adapt and optimize the design of the outer skin in the early phases of development, so that the amount of post-processing required in later phases can be kept to a minimum. As a basis, body styling data (created in CAS tools) available in the initial stages of the development process are imported into the CAD environment. As prior mentioned, the CAD environment provides a high level of integrated knowledge-based engineering capabilities. This means that formulas, rules, check operations, mathematical or logical functions are implemented to calculate the stone movement path and to simulate the impact. If the imported surface data of the geometry are available in the CAD environment, the surfaces are prepared with the help of suitable pre-processing measures. After all surfaces have been arranged, they are imported as non-parametric surface elements into a predefined CAD-template, which provides all parameters and properties required for the simulation steps [6], [7].

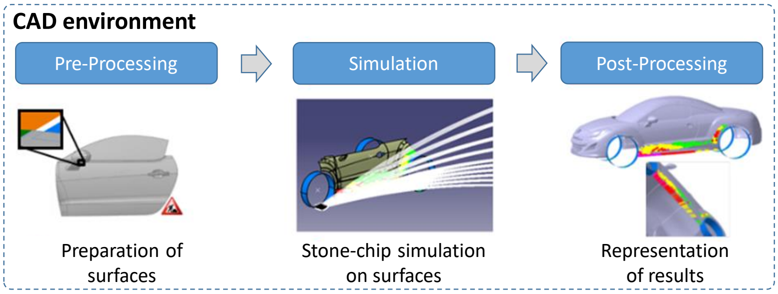


**Figure 9**: Process steps of the tool *Stone Chip*, according [6], [7].

Adapted to the reality in which the stones are accelerated from the ground to the outer skin of the vehicle via a combination of rotation of the tires and translation (movement of the car), this process is taken into account via knowledge-based functions. These functions consider the physical behavior and are integrated into the functional template as mathematical equations [6]. In addition, further parameters such as tire dimension, steering angle, forward or backward movement of the vehicle, speed, size of the stones etc. are defined via the provided GUI in order to run the simulation. In post-processing, the simulation results are processed by calculation of a damage value, representing the damage potential of the vehicle outer skin under consideration of several parameters, e.g. stone mass, velocity, movement path direction, impact direction, impact energy equivalent. The different parameters are assigned a tailored weighting and presented directly on the car body surface in areas with different color coding. The different damage values for the impact areas of the stones serve as basis for the color gradient. Figure 10 shows the color scale that the CAD engineers use to draw conclusions about the stress of the areas on the body surface [7].


**Figure 10:** Colorized damage scale to represent the stone impacts.

Figure 11 shows in an exemplary result of post processing which areas are stressed due to stone impacts. In this example, the color scale ranges from blue (rear area of the vehicle body) to yellow (front area of the vehicle body). This means that the exemplary BIW design data show areas with low damage potentials up to areas with a middle damage potential.

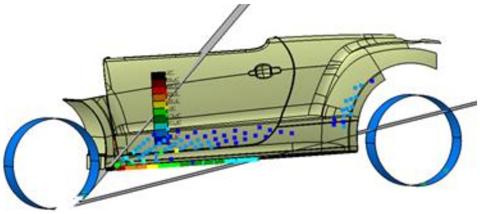

**Figure 11:** Exemplary post-processing.

Regarding the potentials in terms of CAD-automation, the exemplary shown tool provides the integration of complex formulas and parameters that interact within the functional CAD-template. However, the user has the possibility to make different settings via an integrated GUI, which in turn increases the development and maintenance effort. The presented CAD-automation is limited mainly by the complexity of the existing surfaces of the outer skin, which are widely used in the automotive industry, especially in course of styling development. Therefore, the CAS surfaces are to be prepared and simplified (e.g. merging several, small individual surfaces into a comprehensive geometry model). The high number of simulated stones and their impact spots as well as repeated simulation in case of styling variations lead to an increase of requirements on the computer performance – in this way the application has to be planned carefully in the development process. The example represents an application of functional templates with integrated complex mathematical representation of physical effects. In addition, the post-processing module enables user-friendly evaluation of the simulation results. The tool is applied in industry to speed up the development of automotive bodies, to reduce the number of errors of physical prototypes and thus save development cost.

## 6 CONCLUSION

Increasing complexity, integration levels and data volumes in computer-aided engineering applications as well as steadily rising time and cost pressure call for effective tools and methods in automotive development. Due to the central role of CAD in automotive product and manufacturing development, the automation of both CAD-based engineering tasks and data exchange plays an important role for process optimization.

While the exemplary introduced tool *Joining Converter* is based on a type of data conversion that state-of-the-art CAD programs cannot perform, the software package *Stone Chip* focuses on the automation of recurring tasks with automatic parameter changes during the simulation. Both tools offer an approach to close gaps in development and production engineering processes of automotive bodies. The initially high effort for creation and maintenance of automated CAD applications, respectively software packages, is counteracted by the benefits that these tools accelerate the development in a large number of projects. In the creation of CAD-automation, smart solutions have to be implemented to enable effective data processing and geometry creation and manipulation. This also can lead to a reduction of the necessary computational power and thus the simulation processes can be completed faster. In addition, the creation of CAD-automation solutions requires specific know-how so that the tools can optimally intervene in the development and production processes. The implementation of machine learning into automotive development represents a new approach with large potentials. In view of CAD-automation, machine learning involves sophisticated CAD applications in both, the provision of comprehensive data for training of artificial intelligence as well as the implementation of neuronal networks into the processes of geometry creation and variation.

In this way, CAD-automation is able to contribute to more efficient development processes and improvement of development quality at the same time – especially in case of complex and recurring working tasks. On the other hand, the creation, implementation and maintenance of automated CAD processes requires effort and specific know-how. In this context, CAD-automation activities have to be planned comprehensively, tailored and conducted under consideration of the specific boundary conditions and influencing factors. Keys for successful implementation are a clear understanding of the working tasks to be automated, a well-balanced cost-benefit analysis and the development of suitable solutions according to the actual requirements. At the end, CAD-automation has a great potential to support the creativity and productivity of the involved development engineers and to optimize automotive development effectively.

*Alexander Kreis,* http://orcid.org/0000-0002-2898-1731
*Mario Hirz,* http://orcid.org/0000-0002-4502-4255
*Patrick Rossbacher,* https://orcid.org/0000-0002-9184-1870

## REFERENCES

[1] CATIA, https://www.3ds.com, Dassault Systemes, date of access: 11.05.2020.
[2] Creo, https://www.ptc.com/en/products/cad/creo, PTC Creo, date of access: 11.05.2020.
[3] Economidis, N.; Franke, C.; Golumba, J.: xMCF – A Standard for Describing Connections and Joints in the Automotive Industry v3.0, FAT, Berlin, Germany, 2016,
[4] Dekhtiar, J.; Durupt, A.; Bricogne, M.; Eynard, B.; Rowson, H.; Kiritsis, D.: Deep learning for big data applications in CAD and PLM - Review of research, opportunities and case studies. Computers in Industry, 2018. https://doi.org/10.1016/j.compind.2018.04.005
[5] Gfrerrer, A.; Lang, J.; Harrich, A.; Hirz, M.; Mayr, J.: Car Side Window Kinematics, Computer-aided Design, Elsevier, 43(4), 2011, 410-416. https://doi.org/10.1016/j.cad.2011.01.009
[6] Ernst, M.; Hirz, M.; Stadler, S.: A Method of CAD Based Automation and Simulation by the Example of Virtual Stone Chipping Testing, Computer Aided Design and Applications, 11(3), 2014, 295-304. https://doi.org/10.1080/16864360.2014.863495
[7] Fragner, A.; Kreis, A.; Hirz, M.: Virtual Tools to Support Design and Production Engineering – Early detection of stone chips to optimize production processes, IEEE Xplore, paper accepted, 2020.
[8] Hirz, M.; Dietrich, W.; Gfrerrer A.; Lang, J.: Integrated Computer-Aided Design in Automotive Development, Springer, Berlin, Germany, 2013. https://doi.org/10.1007/978-3-642-11940-8
[9] Hirz, M.; Rossbacher, P.; Gulanova, J.: Future Trends in CAD – from the Perspective of Automotive Industry, Computer-Aided Design and Applications, 14(6), 2017, 734-741. https://doi.org/10.1080/16864360.2017.1287675
[10] ISO 14306:2017: Industrial automation systems and integration — JT file format specification for 3D visualization, https://www.iso.org, date of access: 10.05.2020.
[11] Kreis, A.: Tailored Data Exchange Processes for Automotive Body Development, Ph.D. Thesis, Graz University of Technology, Austria, 2020.
[12] Kreis, A.; Hirz, M.; Stadler, S.: A Contribution to Optimized Data Exchange Supporting Automotive Bodywork Engineering, Computer-Aided Design and Applications, 17(1), 2020, 178-189. https://doi.org/10.14733/cadaps.2020.178-189
[13] Kreis, A.; Hirz, M.; Stadler, S.: Optimized Information Exchange Process between CAD and CAM, The 5th International Conference on Industrial Engineering and Applications (ICIEA), Singapore, IEEE Xplore, 2018, 184-188. https://doi.org/10.1109/IEA.2018.8387093
[14] Kreutzer, R.; Sirrenberg, M.: Understanding Artificial Intelligence - Fundamentals, Use Cases and Methods for a Corporate AI Journey, Spring Publisher, Switzerland, 2020.
[15] Marr, B.: Artificial Intelligence in Practice: How 50 Successful Companies Used AI and Machine Learning to Solve Problems, Wiley Publisher, New York, United States of America, 2019.

[16] Rossbacher, P.: Contribution of Integrated Concept Models to the Virtual Entire Vehicle Development in the Early Concept Phase, Ph.D. Thesis, Graz University of Technology, Austria, 2018.

[17] Rossbacher, P.; Hirz, M.: Flexible Parameterization Strategies in Automotive 3D Vehicle Layout, Computer-Aided Design and Applications, 14(5), 2016, 549-562. https://doi.org/10.1080/16864360.2016.1273575

[18] Salchner, M.; Stadler, S.; Hirz, M.; Mayr, J.; Ameye, J.: Multi-CAD Approach for Knowledge-Based Design Methods, Computer-Aided Design and Applications, 13(4), 2016, 471-483. https://doi.org/10.1080/16864360.2015.1131540

[19] Siemens NX, https://www.plm.automation.siemens.com/global/en/index.html, Siemens Digital Industries Software, date of access: 11.05.2020.

[20] Stadler, S.; Hirz, M.: A Knowledge-Based Framework for Integration of Computer Aided Styling and Computer Aided Engineering, Computer-aided Design and Applications, 13(4), 2016, 558-569. https://doi.org/10.1080/16864360.2015.1131552

[21] Stadler, S.; Hirz, M.; Thum, K.; Rossbacher, P.: Conceptual Full-Vehicle Development Supported by Integrated Computer-Aided Design Methods, Computer-Aided Design and Applications, 10(1), 2013, 159-172. https://doi.org/10.3722/cadaps.2013.159-172

[22] STEP AP 242 (ISO 10303-242): Managed model-based 3D engineering, homepage of the STEP AP242 Project: http://www.ap242.org/, date of access: 10.05.2020.

[23] Thum, K.; Hirz, M.; Mayr, J.: An Integrated Approach Supporting Design, Simulation and Production Engineering of Connection Techniques in Automotive Body-In White Development, Computer Aided Design and Applications, 11(4), 2014, 411-416. http://dx.doi.org/10.1080/16864360.2014.881183

[24] Troll, A.: CAx Data Exchange with Neutral Data Formats, Ph.D. Thesis, Technical University of Bayreuth, Germany, 2011.

[25] Vajna, S.; Weber, C.; Zeman, K.; Hehenberger, P.; Gerhard, D.; Wartzack, S.: CAx for Engineers, Springer, Wiesbaden, Germany, 2017. https://doi.org/10.1007/978-3-662-54624-6

[26] VDI Guideline 2206: Design methodology for mechatronic systems, 06/2004, available at the VDI site: https://www.vdi.de, date of access: 10.05.2020.

[27] Weber, J.: Automotive Development Processes – Processes for Successful Customer Oriented Vehicle Development, Springer, Munich, Germany, 2009. https://doi.org/10.1007/978-3-642-11940-8