Title:
A New Approach of a Global Knowledge-based Engineering Infrastructure

Authors:
Markus Salchner, markus.salchner@tugraz.at, Graz University of Technology
Mario Hirz, mario.hirz@tugraz.at, Graz University of Technology
Severin Stadler, severin.stadler@magna.com, MAGNA STEYR Engineering AG & Co KG
Jonathan Ameye, jonathan.ameye@magna.com, MAGNA STEYR Engineering AG & Co KG

Keywords:
Knowledge-Based Design, Design Automation, Multi-CAD, Globalization

DOI: 10.14733/cadconfP.2016.100-104

Introduction:
The ongoing globalization in almost all markets leads to a large field of IT-related challenges. Focusing on development processes in mechanical engineering disciplines, as they occur in automotive industry, globalization requires a continuously and location-independent provision of software tools and project-oriented data. This involves both, the supply of a large amount of different commercial software packages for various engineering disciplines, as well as the handling of large amount of data, configurations, different projects, sites and employee-related information. The broad field of computational design and engineering disciplines within automotive development requires a strong interaction in view of efficient software applications. Especially the CAD-based design phase is characterized by a cooperation of manufacturer and supplier, which often use dissimilar CAD software with different versions or even different vendors. Previous research work [7] focused on the development and application of knowledge-based engineering (KBE) tools within a multi-CAD environment. A method has been proposed for the efficient development of KBE applications, which are able to concurrently support multiple CAD systems and releases. Based on this fundamental approach, the next logical step includes the development of an efficient and secure distribution environment for KBE applications. In this context, the present publication introduces a new approach for the management and distribution of KBE applications, project related CAD environments and templates, as well as product data quality (PDQ) methods in a worldwide context. [4]

Main Idea:
The main idea of the findings includes the development of a holistic approach for an entire KBE process chain. The cognitions presented in the paper consider that certain preliminary tasks and decisions, related to global directives and development environment, have to be set in advance. Under consideration of this fundamental precondition, challenges can be found in the subsequent tasks of distribution and rollout of the developed KBE applications. Not only a secure deployment of the software itself, but also comprehensive and efficient management, maintainability and user feedback handling is required for successful application in daily business. Furthermore, data handling of various project and customer requirements have to be considered in this context. Another challenging task includes the inter-site collaboration considering various development departments all over the world.

Global directives and development environment
Starting with the management decision to set up global directives for the development process, a comprehensive CAD design and CAD automation strategy has to be defined. The KBE automation strategy includes the definition of accepted automation methods, programming language(s), programming environment and supported CAD systems. Certainly, these decisions strongly influence

the upcoming development and operating costs, and at the same time the realizable functionality of the environment. The proposed approach in this paper can only be implemented successfully with the highest level of KBE automation, as introduced in [7]. This requires the use of a CAD independent programming environment, like Visual Studio [5] or Eclipse [2], which again involves common programming language of the selected CAD system(s). It is mainly driven by the provided application programming interface(s) (API) of the chosen CAD system(s). For example, the widespread CAD system CATIA from Dassault Systemés [1] provides a .NET interface for the development of externally coupled KBE applications. Because most of the CAD systems provide a .NET interface, the .NET programming language and Visual Studio [5] as developing environment are recommended. Once the global directives are defined, the next step includes the development of a software development kit (SDK) as basis for the efficient creation of KBE applications. The SDK has been developed over the last years and simplifies the KBE development process (especially for not experienced programmers), reduces repetitive tasks and provides a consistent KBE structure. This includes defined interfaces to the various CAD systems, databases and also third-party applications – Excel, Powerpoint, etc. - are supported. Regarding the CAD system, the SDK provides a variety of basic geometric and infrastructure functions, like analyzation of the assembly structure, the related part names, properties as well as materials. [7]

Software distribution approaches

KBE is a widespread term and can be used for variety of engineering tasks. In this paper KBE applications are standalone software applications, developed on the .NET framework, which use the provided APIs to communicate with the selected systems. An exemplary application is an advanced and customizable BOM (bill of material) application which exports the assembly structure, properties, and further meta information to an Excel file.

Each KBE application consists of one or more application files (*.exe, *.dll, *.config …) which have to be executed on different client machines. The entry point of the presented approach is that only one single application file is distributed - a so called “All in one”. This minimizes the amount of different files, which have to be handled by a distribution portal. Considering a midsize KBE application that consists of about five to ten various programming files, the complexity can be reduced dramatically by this approach. The obvious drawback of this concept is, that software updates cannot be done by replacing single files. Instead, the entire application has to be released new. This disadvantage may become significant within large software packages like a CAD system or an operating system where an installation is mandatory. KBE applications normally do not require an installation instance and they have a manageable size – some megabytes. Therefore, this seeming drawback isn’t one, because administrator privileges are not required to run the different KBE applications. To sum up, in the present approach there is only one single application file (*.exe), which has to be released for the client machines.

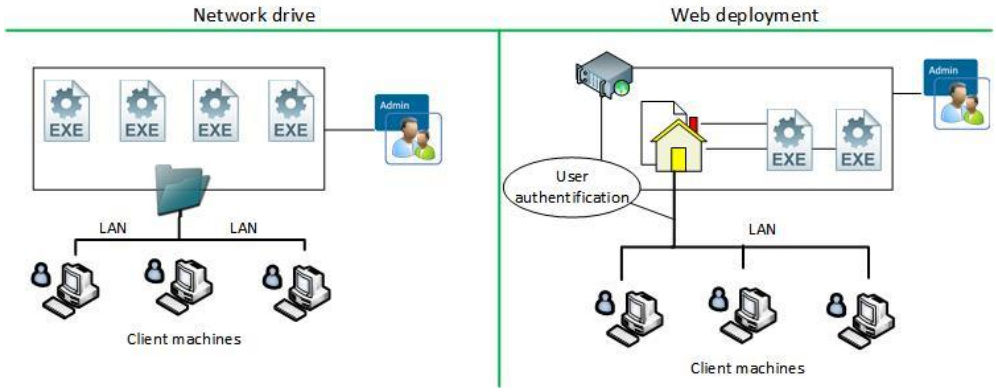

Fig. 1: Common software distribution approaches.

A suitable approach for small and medium-sized enterprises (SMEs) within straightforward project environment and small customer related influences could be a software distribution using a company

wide network drive. Therefore, a centralized storage is used, where the design engineer can run the required applications. The responsible KBE administrator on the other hand can update and maintain the applications with minimum effort. Using this approach, increased effort has to be considered, like the management of user or project related access rights or even a licensing model. Furthermore, the KBE administrator cannot guaranty that all design engineers use the latest software version or they do not make an illegal copy of the software. Considering a multi-site company structure, a secure web access to the network drive, like a virtual private network (VPN) [6] connection has to be set up. Figure 1, left, illustrates this approach of a local area network (LAN). [5], [6]

Another possibility to distribute KBE applications is based on a download portal, as it is used for different commercial applications yet. The KBE administrator can provide the applications and updates on a web server, where registered users can download the required applications. Depending on the web server configuration the download portal page can be accessed from everyone via the internet or just from computers within the LAN - as illustrated in Figure 1, right. This approach is quite similar to the previous one, but with the main difference that user access can be managed in a proper way by assigning roles or other project related qualifiers. However, the user receives a copy of the KBE applications and the KBE administrator has no influence on the further use or misuse of this applications.

With these two common approaches, it is not possible to control the access rights, to fulfill security aspects, to handle project related requirements, to get reasonable feedbacks and to enable an efficient global distribution of KBE applications.

Project environments

Each project within a development process has a predefined environment, including guidelines, rules, naming conventions, etc. A holistic KBE application has to meet these requirements using an object oriented, agile, polymorphic programming approach, but the distribution environment has to be able to support this solution layout. A simple example includes the usage of different KBE application configuration files. This type of files defines the supported CAD system and releases, the used storage folder for meta and output data of the KBE application, like captured images or result files, as well as a naming convention for parts and assembly components. Therefore, one KBE application can be morphed to various project environments, if the distribution portal supports the handling of the defined settings and files. Further project related configurations are CAD systems settings like the overall system accuracy, a predefined structural layout for part and assembly components and the definition of product quality management (PQM) guidelines including their verification prescriptions, as illustrated in Figure 2. The presented approach does not focus on the PDM [8] environment but rather on the illustrated interface between the CAD systems and their environment.

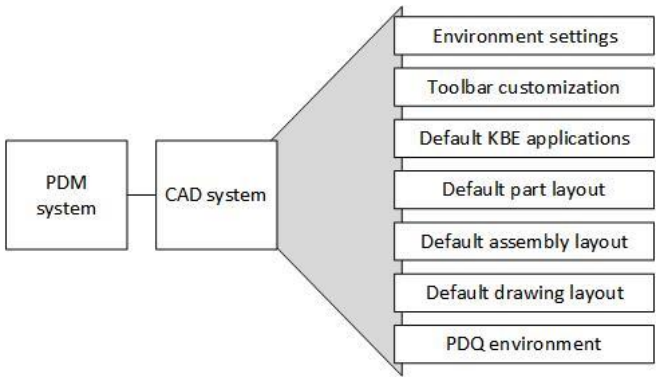

Fig. 2: CAD related project environment.

Proposed approach

The proposed approach covers the mentioned requirements for the distribution of KBE applications within a global environment. This includes an efficient management environment, the control of access

rights to fulfill security aspects, the handling of project related necessities and a reasonable feedback method. The approach consists of four interlinked main modules:

1. Intranet site
2. Global administration management system – database and application
3. Site databases with managing application
4. The final client environment

The intranet site provides the basis for knowledge distribution and feedback-related functions. The first objective is to inform the design engineers about available and planned applications. This includes a short description of each KBE tool, the current development status and a short video clip tutorial. Furthermore, each design engineer can announce personal needs for the listed applications. The development department gets a profound feedback regarding the demand of different applications - and so it can adjust the release roadmap. Additionally, the homepage provides an upload site, where engineers can propose new ideas and improvements. This can be done via written text on the page and upload of related files, like images, presentations or even existing automation scripts. In an exemplary project, the homepage was programmed within the Visual Studio environment using the ASP.NET [5] technology. Therefore, the homepage is based on the .NET framework, which allows the implementation of developed methods within the SDK and the use of database functionalities. In this way, the main content of the homepage can be coupled with the master database, as explained in the next paragraph.

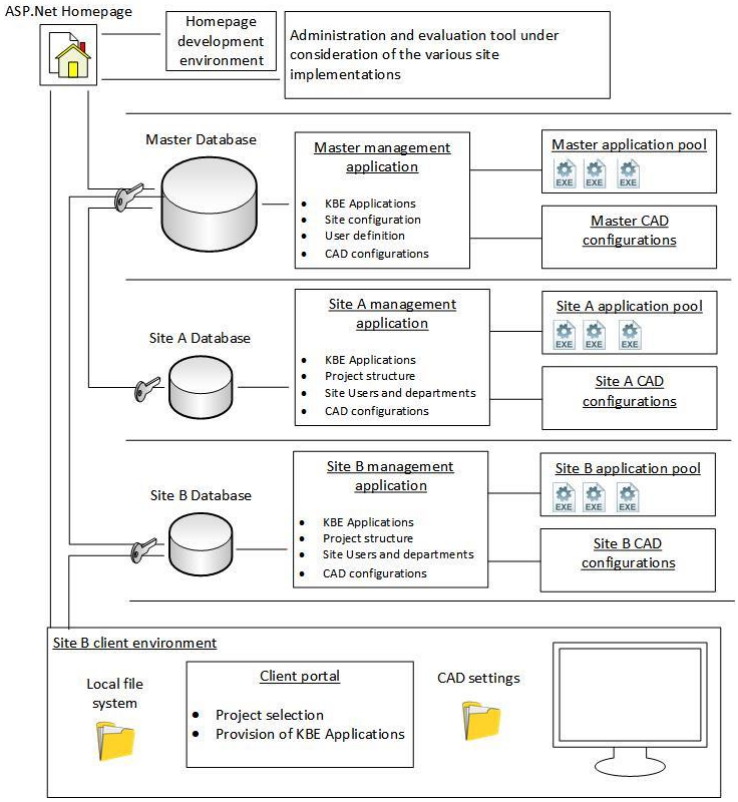

Fig. 3: Global distribution concept.

The master layer of the distribution concept comprises a master database and a management application for administration tasks. The application enables mainly the configuration and setup of the relational master database. This includes the setup of default CAD configurations, site management and the control of access rights to this database. Certainly, this layer is mainly responsible for the provision of the different KBE applications within the database [3].

The site layers, illustrated in Figure 3, receive a partial copy of the master database accordingly to the defined access rights. The data consistency between the master and the site layer can be guaranteed due to a replication approach. This abstraction leads to some advantages regarding security aspects, adaptability and customization options, as well as to performance improvements. Supplementary, each site has a management suite, where the responsible employees can configure their environment to their specific needs. This can be the definition of a particular project / department structure or site specific KBE-applications. A mandatory task for the site administrator is the definition of process or user related access rights. This task can easily be achieved within the management application. The consistence of the entire approach can only be fulfilled due to an information and data exchange with the master database.

The final layer represents a client environment within a specific company site. Figure 3 illustrates a common client machine for design engineers. The depicted elements are the local file system, where the specific software packages and the CAD programs are installed. The design engineer can work within this native environment. The next block illustrates the client portal. This portal is linked to the site database and allows the design engineer to select the appropriate project. On the one hand, this provides a listing of all available KBE applications, which are activated for the selected project or the current user. On the other hand, the project related CAD settings are extracted to the local installation folder, depicted as CAD settings folder. This configuration represents itself as a background task, and so the designer has not to take care about it. As soon as the configuration is completed, the design engineer can start the CAD application from a button within the client portal. In contrast to other approach, the KBE applications are not copied to the local file system. The applications are invoked within the client portal, so there is no local copy of the KBE application on the client machine, and therefore the engineer cannot make a copy of the application or use it in illegal manner.

Conclusions:

The presented approach of a global knowledge-based engineering infrastructure provides a global available groundwork for the management and distribution of KBE applications as well as for the definition of project related CAD environments including part, assembly and drawing templates. This includes an efficient management environment, the control of access rights to fulfill security aspects, the handling of project related necessities and a reasonable feedback management structure. In addition, the development of a database concept, the definition of a generic CAD environment and an enhanced server concept is supported. In addition to the approach presented in this abstract, the final paper will include a discussion of product data quality related topics, exemplary for the application of the CAD package Siemens NX [8], and their influences on the entire KBE process chain. The introduced approach has been implemented in an industrial application and is currently in beta testing phase. Intermediate test results show a great benefit in terms of increased efficiency in to effective development, global provision and management of knowledge-based engineering applications within CAD environment.

References:

[1] Dassault Systemés, http://www.3ds.com, access date: 2015-12-14.
[2] Eclipse, https://eclipse.org/, access date: 2015-12-12.
[3] Firebird, http://www.firebirdsql.org/, access date: 2015-04-13.
[4] Hirz, M.; Dietrich, W.; Gfrerrer, A.; Lang, J.: Integrated computer-aided design in automotive development: development processes, geometric fundamentals, methods of CAD, knowledge-based engineering data management, Springer, 2013, ISBN: 9783642119392, http://dx.doi.org/10.1007/978-3-642-11940-8.
[5] Microsoft, http://microsoft.com, access date: 2015-12-14.
[6] Microsoft Developer Network, http://msdn.microsoft.com/de-at/dn308572.aspx.
[7] Salchner, M.; Stadler, S.; Hirz, M.; Mayr, J.; Ameye, J.: Multi-CAD Approach for Knowledge-Based design methods, Computer-Aided Design and Applications, 2015, 216-268. http://dx.doi.org/10.1080/16864360.2015.1131540
[8] Siemens PLM, http://www.plm.automation.siemens.com, access date: 2015-12-12.