

## **IMPROVING GEOMETRY MANIPULATION CAPABILITIES OF KNOWLEDGE-BASED ENGINEERING APPLICATIONS BY THE VERSATILE INTEGRATION OF 3D-CAD SYSTEMS**

*Alexander Ortner-Pichler, Christian Landschützer*

Dipl.-Ing., Ass.Prof. Dipl.-Ing. Dr.techn.

Graz University of Technology Institute for Logistics Engineering

### **ABSTRACT**

This article presents the special conditions and success factors regarding the application of Knowledge-based Engineering (KBE) within technical logistics. The authors introduce the KBx approach, which tries to reduce a system's complexity, thus enabling an efficient development of KBx-applications. Considering the disadvantages regarding geometry manipulation of current KBE systems an interface-based method (x2Creo) for the application development is shown.

### *Keywords*

logistics, logistics engineering, knowledge-based engineering, KBE, CAD, KBx, interface-based, API, web based, WebGL, PTC Creo, VBApi, x2Creo

### **KNOWLEDGE-BASED ENGINEERING IN LOGISTICS**

Virtual geometries of physical bodies are constituent parts of development work within nearly every sector of production. From fashion to consumer goods as well as from construction to engineering, nearly all development of new ideas and optimization is based on geometry on different levels of detail. Nowadays all these branches use CAD software to virtually model geometry and further use this CAD-models in different applications like visualization (digital mock-up), calculation (CAE) and mechanical/engineering design. As all development and optimization work is highly iterative, one has to take care on effort within this iterations. One approach within engineering to carefully handle CAD manipulation and modelling is to use knowledge-based engineering (KBE). These engineering approaches are based on formalized, explicit knowledge that is able to modify CAD geometry<sup>1</sup>.

Now engineering is a broadly used terminus in different activities of mechanical engineering. From one industrial sector to another nearly anybody developed own and customized approaches and methods to fulfil engineering tasks. Focussing on design engineering one must take a closer look to various disciplines to identify differences therein. An industry of interest is the logistics one. Several actual

---

<sup>1</sup> [2], [14] delivers insights in history and use of KBE within logistics engineering.

(Mega)trends<sup>2</sup> force producers to deliver more customized products in less time than ever. With an annual volume-increase of more than 7% [12] logistics is one of the fastest growing industries nowadays. Aware of that trend, manufacturers of material handling equipment have to cover this and provide more powerful machinery within faster cycles. This is where engineering processes and single tasks come into consideration for streamlining the whole development.

But engineering in logistics isn't that highly performative and well developed as in other sectors (e.g. the aerospace and automotive industry is used to handle many challenges for years that are arising in logistics engineering actually)<sup>3</sup>. Furthermore KBE isn't easily to adapt for logistics engineering because of the broad variety of material handling products in size and functionality [6]. So material handling objects (the machinery within logistics) are broadly various, and logistic machinery is nearly ever a customized single installation for one special use-case. This arises the idea that highly performative software products and methods can be used to handle complexity and efficiency within this sector.

This paper now summarizes methods, developed within a habilitation Thesis at Graz University of Technology [15] and outlines research perspectives within an ongoing PhD work to rise productivity within logistics engineering work within CAD and geometry modelling. Further insight and methods within engineering for logistics are described in [9].

### *Engineering design and KBE within logistics engineering*

The engineering design process is settled between general requirements definition and production within the product life-cycle [1]. For logistics engineering it can be assigned and depicted in parallel to logistic layouting processes to execute "simultaneous engineering in logistics"<sup>4</sup>. To make simultaneous engineering in logistics work one needs to use knowledge-based methods for faster design iterations. Formalized knowledge within these solutions is the backbone of these systems. The effect of KBL (see chapter "Differing knowledge-based engineering: the KBx-approach") for simultaneous engineering in logistics can be seen in figure 1 and is fully described in [15] and [16].

As history and scope of KBE development is described well [14], the focus here lies on enhancing productivity and technical solutions within KBE systems by CAD integration. Beside further developing fully automated design-systems – having in mind, that most of the approaches failed in the past – it is more efficient to carefully differ between usefully automatable design tasks by introducing the KBx-approach.

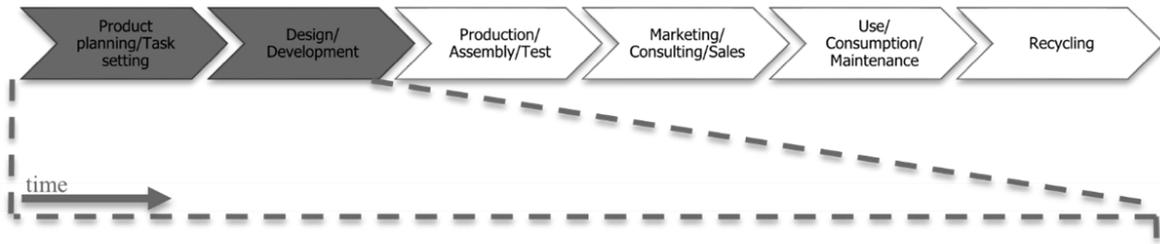
---

<sup>2</sup> See e.g. [4] for connection of megatrends to logistics.

<sup>3</sup> Source, assumptions, conclusions and methods to close a gap between actual stage of engineering between aerospace and automotive industry and logistics are one outcome of [15].

<sup>4</sup> Details on simultaneous engineering in logistics in [16], [15]; for differing knowledge-based X into Engineering/SystemDesign/Layouting and scope of use see chap. *Differing knowledge-based engineering: the KBx-approach*

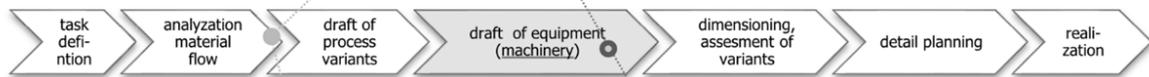
Product life-cycle:



4-step-layouting-process [9]:



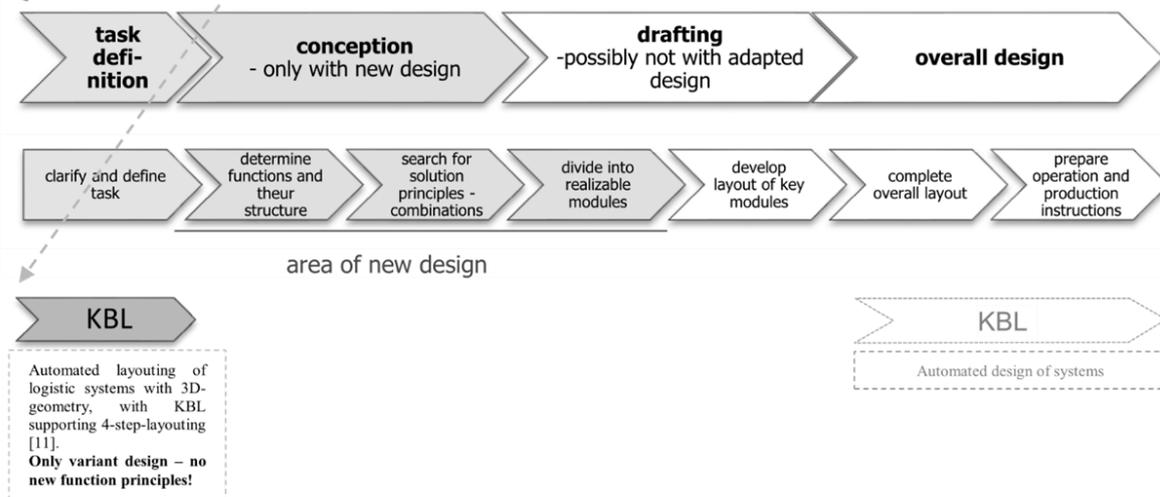
7-step-layouting-process [19]:



4-step-layouting-process [17], [18]:



General approach to design after [1]:



- actual
- simultaneous engineering for new design
- slashed: possibilities with KBL

Figure 1  
Simultaneous engineering for logistics [16]

*Differing knowledge-based engineering: the KBx-approach (following [14])*

To make KBE successful it is necessary, as a key result of literature review [14], to differ between the various degrees of automation in design work. Design work in material handling is completely different if one has to design e.g. a wire rope drum

or if one has to layout a complete storage system. There are certain tasks more or less predestinated for KBE so that with a determination that reflects this degree of automation the authors talk about KBx. The manifestations of automated design in KBx will than have a clear database, interconnections and goals for varying applications. Focusing successfully realized applications of design automation (s. Refs in Tab. 1), and assessing efforts of building automated designs, the authors introduced [14] a way to differ between Knowledge-based:

- Engineering (KBE)
- System Design (KBSD)
- Layouting (KBL)

which have very different scopes of use, functions, powering knowledge and application (table 1).

Table 1  
KBx-definitions

		-----size of systems vs. Degree of automation----->		
		<b>KBx</b>		
		Knowledge-based engineering approaches at different detail design levels		
scope of automated engineering		<b>KBE</b> <small>Knowledge Based Engineering</small> components, parts, machines	<b>KBSD</b> <small>Knowledge Based System Design</small> machines and systems	<b>KBL</b> <small>Knowledge Based Layouting</small> systems
depth of knowledge and used tools and techniques	<b>functions</b>	full automated (detail) design of parts and subassemblies	full automated master and layout design of assemblies and systems, specification of machinery	full automated layouting of systems, specification of systems
	<b>use for</b>	- customizing machinery - tailored products - product families	- dimensioning motors - defining interfaces - CAD top-down design - CAE models (structural, dyn.)	- space requirements - early cost estimation (bidding) - drafting bill of material
	<b>CAD domain</b>	detail geometry models	reduced geometry for CAE	shrink wrap geometry for layout
	<b>data, information and knowledge sources</b> <small>(beside employee know how and workflows)</small>	- standards, best practice - production facilities - manufacturer data <b>engineering theory</b>	- standards, best practice - supplier and engine data <b>eng. and mechanics theory</b> material flow calculation (throughput, capacity)	- standards, best practice - manufacturer database - customer rel. management <b>logistics theory</b>
	<b>practical applic.</b>	wire rope drum [3]	AS/RS [2]	HRL-tool [14]
xKBE	<b>2.0 knowledge sources</b>	web services, online databases, knowledge management systems		
	<b>2.0 technologies</b>	wikis, semantics		
	<b>practical applic.</b>	tagged standards and technical docs, relation matrices		

The by interviews captured knowledge<sup>5</sup> in form of best practice, CAD-methodologies and general engineering know-how meets with those from standards, knowledge-databases (like company wikis), supplier data and PLM-databases for CAD. These are the powering sources of data, information and knowledge, which are the informal base for the development of a KBx-solution. With every material handling design solution driven by the two main parameters throughput and capacity, the various sources of knowledge for KBx can be seen in Tab. 1 as an information

<sup>5</sup> For necessity to develop KBx solutions and obstacles therein see chap. "Challenges regarding the geometry Manipulation in KBE Applications"

basis. Bringing all this information together makes common knowledge-based X more powerful and better structured for development and maintenance<sup>6</sup>.

Figure 2 describes graphically what effects can be achieved by KBx. Where a vertical dimension differs objects of material handling into four different sizes, the x-axis leads through the development process. The depth-dimension now depicts different effects of the KBx: powering varying or specializing of products. Coloured arrows depict acceleration in engineering processes graphically. Intelligent filters and 3D-viewing operations allow searching for suitable methods (actually filled with ten different methods form [15]) in different stages at different logistic object's sizes (use [13] and the QR-Code in figure 2 for testing – in german only). It can be clearly identified that all KBx activities help varying special products and so help to fulfil customer demands and tailored products. Of course, KBx is not an answer for mass-products and building sets, but within logistics variation and customization addresses geometries of objects much more, than function principles are changed. So for variant design – with identical function principles – KBx is a powerful answer.

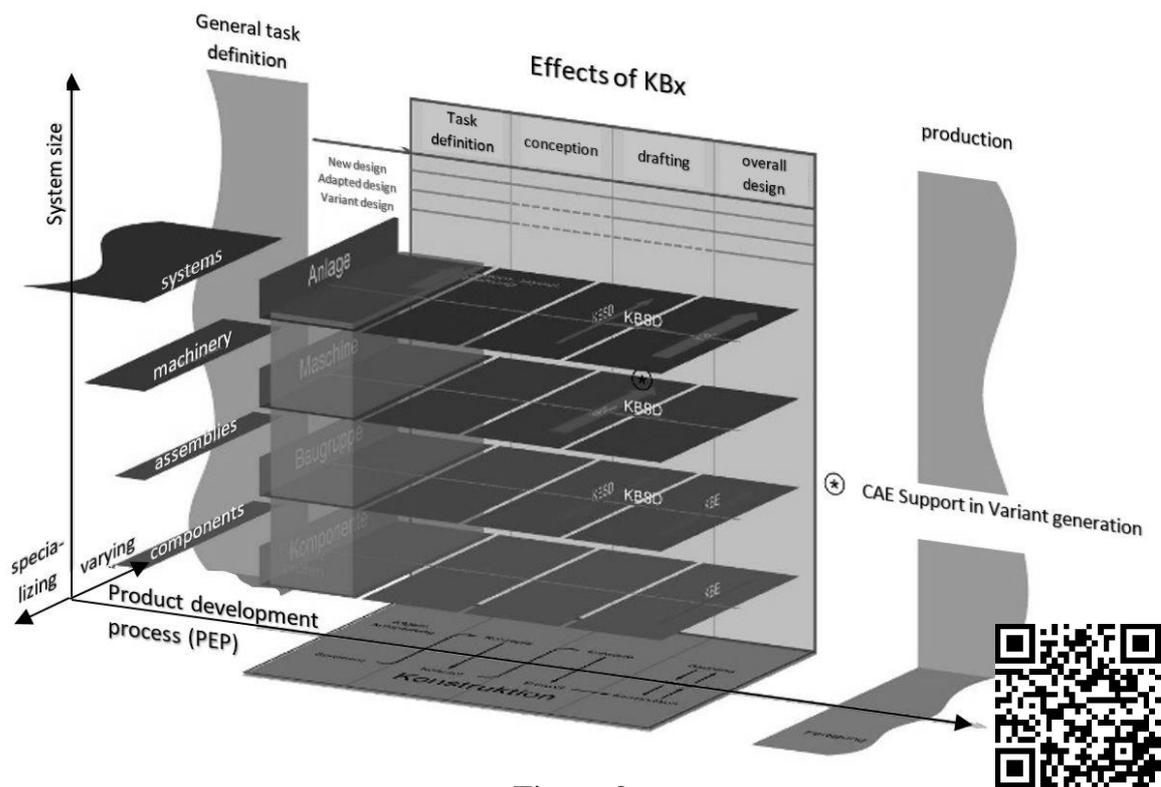


Figure 2  
Effects of KBx in different design stages [15]

<sup>6</sup> [15] and [16] further outline and define a methodology, how to efficiently manage captured knowledge within KBx working tasks. The xKBE-app, developed at the Institute of Logistics Engineering in Graz, is a multidimensional and graphically oriented database (ontology), that is able to extract, store and modify formal knowledge within CAD. A fully functional prototype is available for knowledge-based engineering within PTC Creo 3.0.

## *Challenges regarding the geometry Manipulation in KBE Applications*

When developing a KBx application another differentiation has to be made, if the application should be realized as full-KBE or augmented CAD KBE<sup>7</sup>. Applications based on the augmented CAD-KBE system provide functionality for a stable geometry manipulation due to their deep integration in the CAD-system, but this integration arises shortcomings in generative modelling capacity matters. Existing systems are too not language-oriented which limits their functionality regarding rule capture and knowledge handling [11].

While eliminating these shortcomings of augmented CAD-KBE systems, existing full-KBE systems encounter major disadvantages in geometry manipulation. When facing that more than 70% of the coding and debugging of a KBE application is geometry related, whilst only less than the half of a product's definition is, these shortcomings have to be taken into account to a large extent [20], [21].

In order to deal with the disadvantages of both full-KBE and augmented CAD KBE regarding geometry manipulation one research direction of the KBx-approach works on the integration of CAD system in KBx applications. The basic idea behind this concept lies in using separated system elements for knowledge capture and use as well as geometry representation and thus following the KBx-approach of reducing a system's complexity. In its most basic form the two core elements can be a calculation scheme implemented in a capable software tool and a parametric CAD model. In order to combine them to a full featured application they are bidirectional interconnected to each other via a specialized interface. The main reason to develop this method for creating KBx applications is that the needed core elements already exist in many cases, thus reducing the effort for the implementation.

## DEVELOPING INTERFACE-BASED KBX APPLICATIONS

Figure 3 shows the basic structure of an interface-based KBx application. The strict separation of the calculation scheme and the geometry generation and manipulation can be seen. Both elements are interconnected via an interface enabling a bidirectional communication for the exchange of parameters, as well as the extraction of visualization and analyzation data (e.g. images, non-parametric geometry and bill of material) of the CAD model.

Beside the configuration of the parametric CAD model, by setting design parameters, the interface to be introduced in the following chapters provides functionality to build custom assemblies of non-parametric components (parts, subassemblies) allowing to create flexible layouting applications specialized for KBL applications.

---

<sup>7</sup> See [14] on details

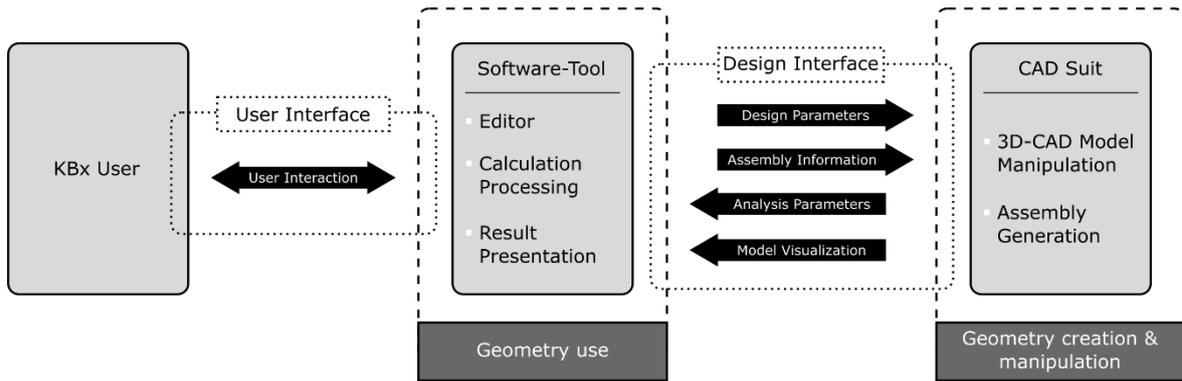


Figure 3

Basic structure of an interface-based KBx application

The capabilities of this function allows the use of existing machinery models, without the need of these models being parameterized, within the overlying system and thus creating layouts of whole facilities.

## X2CREO

The developed interface is named x2Creo, where the X stands for any application providing basic capabilities of writing a plain text file and starting an external executable within a Microsoft Windows system environment. Creo is a reference to PTC's<sup>8</sup> Creo, but almost any other high-end 3D-CAD suit capable of parametric modelling, provides comparable functionality regarding its APIs (application programming interface) and thus enabling an external control. That means that many other 3D-CAD systems could be controlled in a similar way. The core part of the interface enabling control of Creo uses PTC's VBApi that provides both synchronous and asynchronous connections when controlling the CAD suit. Even though VBApi is not the API providing the most powerful functionality of the various APIs included in Creo, the API was chosen, because it provides all functionality to configure parametric CAD models and to build assemblies through referencing the assembled parts between each other, but also keeping things simple enough to handle the complexity of controlling the CAD suit.

In order to enable a simple, but powerful method to control Creo a batch-like command syntax was designed, which is interpreted by the interface and translated into VBApi calls. At the current state of development, there are three ways of delivering the command list to the interface. The first way is simply writing the commands in a plain text file and then calling the interfaces' executable file. For an easy integration into Microsoft Excel a method of parsing the commands directly from an Excel sheet was developed, thus enabling the call of x2Creo by the use of the VBA (Visual Basic for Applications) capabilities of Excel. The probably most

<sup>8</sup> Formerly known as Parametric Technology Corporation, with formerly known Creo as ProEngineer

powerful way is the command transfer via an AJAX call (Asynchronous JavaScript and XML). This method is described in detail in chapter “Extended Capabilities”.

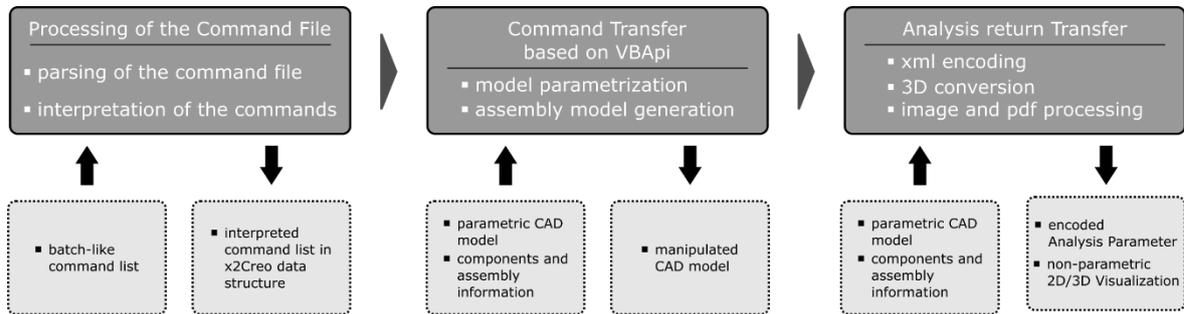


Figure 4  
Workflow of x2Creo

Figure 4 shows the workflow that can be realized using x2Creo. After the generation of the command file by a software tool that executes the calculation for system’s draft, the command file is parsed and interpreted by the interface. The interpreted commands are transferred to Creo manipulating or assembling a 3D-CAD model. Once the update of the model has finished, the interface can extract data from it such as analysis parameters, images or non-parametric geometry for visualization and 2D drawings or the bill of material in pdf form, allowing the software tool the invoked x2Creo to present the results of the manipulation to the user.

### Syntax

The syntax of the x2Creo command file is built in a batch-like style, meaning that each line contains one command starting with its name optionally followed by one or more attributes controlling the command’s behaviour in detail. Attributes and the command name are separated using any whitespace character. Whitespaces leading and following the command statement as well as text following a sharp sign (#) are ignored within the parsing process allowing the structuring of the command file and basic documentation using comments.

The commands can be grouped into three different categories:

- Session-based commands
- Model-based commands
- Grouped commands

While session-based commands allow the control of a Creo instance providing functionality to establish a connection either showing or hiding Creo’s graphical user interface and to set up the current working directory, model-based commands are used to manipulate a specific component by setting design parameters or to save an image of the model displayed. Using model-based commands, the interface can further control Creo to load and save 3D models as well as export 3D models in various formats such as images, non-parametric geometry files or pdf files.

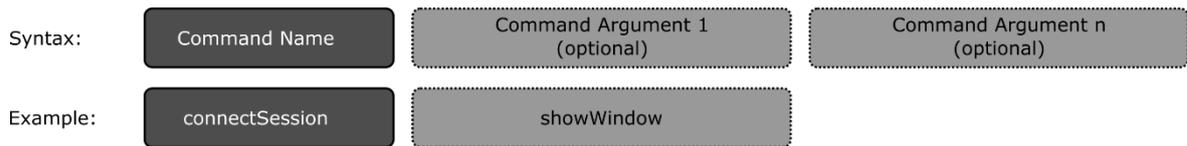


Figure 5  
Session-based Commands

Figures 5 and 6 show the basic structure of session-based and model-based commands. The main difference between these two groups is that model-based commands require a unique model identifier as first argument that specifies the component the command is applied to. This identifier is set up within a call to retrieve a model to the computer's working memory and stored in x2Creo in a hash table.



Figure 6  
Model-based Commands

The third category are grouped commands using a multi-line syntax. That means one or more commands are enclosed within an opening and a closing command. This syntax is for example used when assembling components. The opening command is used to specify the assembly being modified and the component to be assembled. Following different geometry references can be listed to specify the component's position within the assembly. The closing command invokes the assembling.

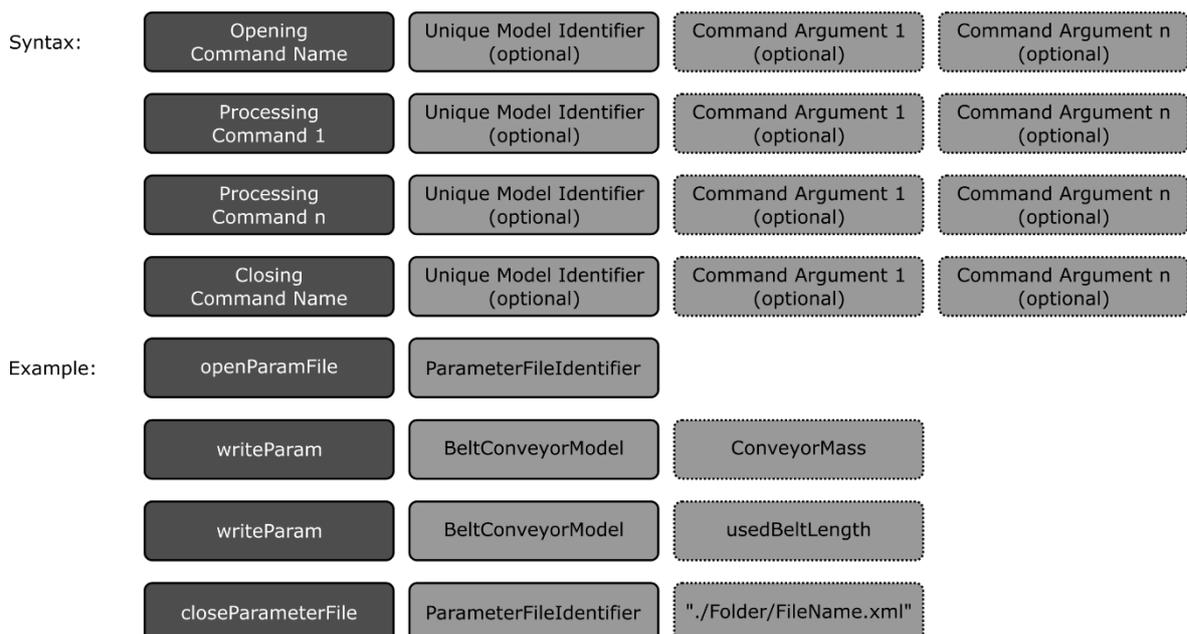


Figure 7  
Grouped Commands

The second grouped command is shown in the example in figure 7 and used to extract design and analysis parameters from Creo. Using the opening command the location of an xml file can be specified in which the parameters specified within the following commands are written to. In order to use this command within Microsoft Excel a VBA subroutine was implemented converting the xml file into a data structure similar to a hash table simplifying its use.

### *Extended Capabilities*

Within a second development branch of x2Creo was extended to enable web based operation. Capabilities to allow a client server based AJAX communication were added so a work flow as shown in figure 8 can be realized.

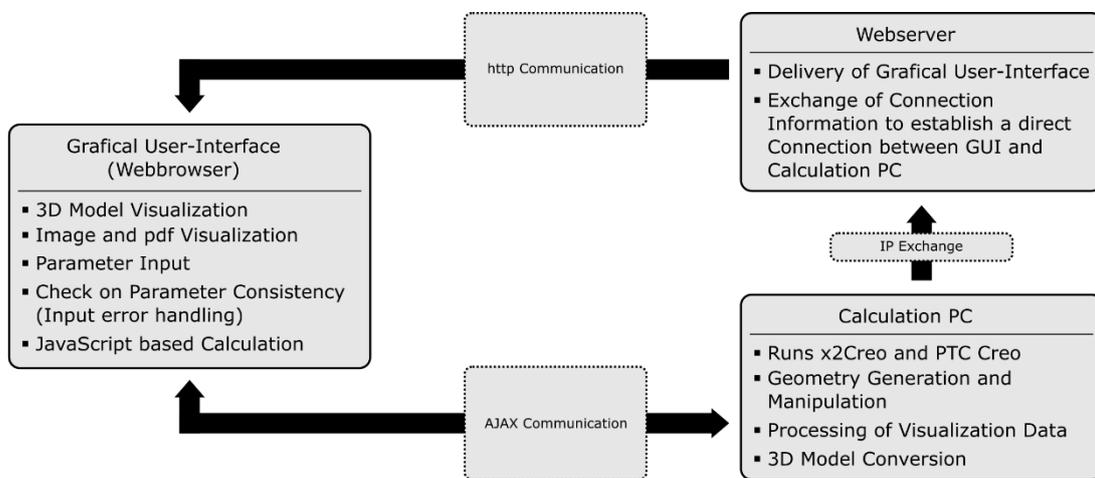


Figure 8

Basic architecture of x2Creo in web based operation mode

The second core feature of x2Creo's extended version is the implementation of a web based user interface using WebGL to display a 3D-CAD model and allow an interactive configuration and layouting process within any modern web browser. The use of standardized technologies such as html5 and WebGL allow the use of KBx applications on many different platforms including mobile devices. In order to provide this visualization capabilities a specialized geometry format was developed. This format enables a fast conversion process starting by the export of a 3D model from Creo into a vrmf file (virtual reality modelling language) and enabling an efficient data transfer by its lightweight structure.

The converted files are transferred to the web browser through an established AJAX connection and then display using the JavaScript framework three.js which is used within the web based user interface to render the geometry. An example of a possible web based user interface is shown in figure 9. Using an AJAX connection allows an asynchronous manipulation process thus enabling fully interactive KBx applications within a web browser.

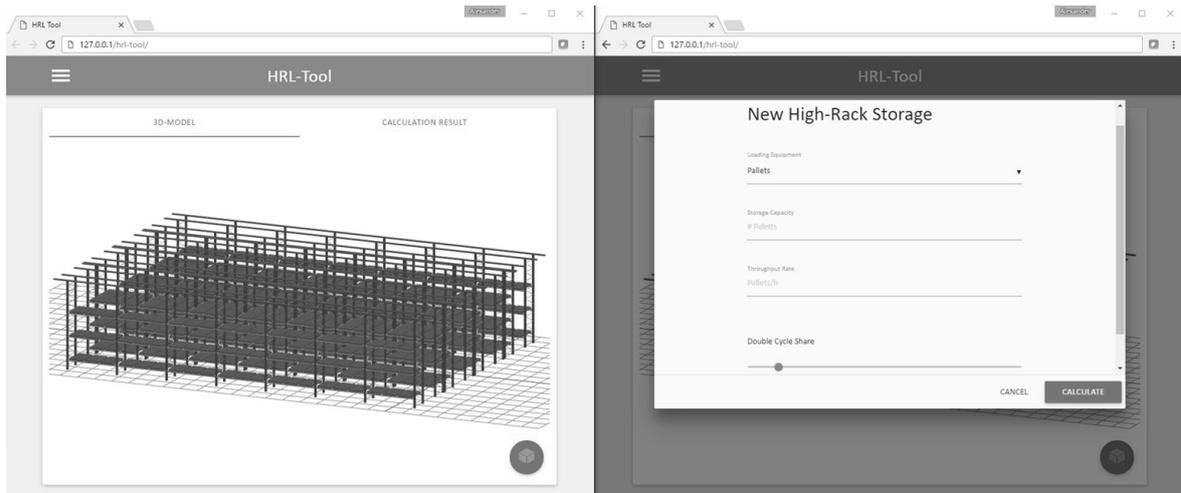


Figure 9

Left: 3D Visualization<sup>9</sup> / Right: Basic Parameters for calculation draft

### Application – use cases

Table 2 shows some sample realizations, made using x2Creo. The broad scope of applications it was used for proves its flexibility and reliability.

Table 2  
Realized applications using x2Creo

Object	x2Creo	Type	Description & Functionality
wire rope drum [3]	local	KBE	<ul style="list-style-type: none"> <li>design and construction of a wire rope drum</li> <li>detailed calculation according to the DIN 15018 Standard</li> <li>generation of technical detail drawings</li> </ul>
roller conveyor [15]	local	KBSD	<ul style="list-style-type: none"> <li>design and construction of a roller conveyor</li> <li>calculation and configuration of the drive system and the rolls</li> <li>generation of technical detail drawings</li> <li>generation of system's bill of material</li> </ul>
wood chip heating [23]	local	KBSD/KBL	<ul style="list-style-type: none"> <li>draft design of a wood chip heating</li> <li>definition of rooms, where the heating unit and the storage should be installed</li> <li>draft construction of the overall system assisted by a guidance system, that helps the user during the design process</li> </ul>
sortation system [5], [22]	local	KBL	<ul style="list-style-type: none"> <li>rough planning of a sortation system</li> <li>configuration and planning process either by the use of predefined layouts or free configurable with reduced geometry support</li> <li>integrated tool to help the user choose the most suitable distribution conveyor according to freely configurable criteria</li> <li>detailed configuration of the distribution conveyor (layout, drive units, loading and discharging)</li> <li>generation of system's bill of material</li> </ul>
HRL-Tool [14]	local & web based	KBL	<ul style="list-style-type: none"> <li>rough planning of an automated high-rack storage</li> <li>integration of product data of stacker cranes and rack systems of various manufacturers in a database</li> <li>fully automated system design and calculation of the system's throughput</li> <li>enabling a simple comparison of different design variants</li> <li>the tool is implemented in both local and web based modes</li> </ul>

<sup>9</sup> The 3D model visualized in the picture is in an early state of development. The actual rendering quality can be much more detailed.

## SUMMARY & OUTLOOK

Knowledge-based engineering and the, for logistics customized, approach with KBx is able, to support customizing and tailored products for modern customers. Even though many methodologies to create KBx applications exist, shortcomings especially regarding geometry manipulation can be found. This paper introduced one possible way to deal with these shortcomings by an interface-based coupling of KBE and CAD systems thus combining the advantages of both. The authors show various examples built using x2Creo and thus prove the practical usability of the interface-based methodology.

One current research direction on the Institute for Technical Logistics focuses on the further enhancement of the methodology introduced. Within this research a software draft is developed and implemented in order to evaluate an automated process to parse families of existing 3D constructions (e.g. continuous conveyors of the same type containing a broad variety of different geometric characteristics) and then mapping them into an object-oriented, hierarchical structure. The so captured variants can be used in semi-automated or assisted design processes.

## REFERENCES

- [1] **VDI 2221**, Methodik zum Entwickeln und Konstruieren technischer Systeme und Produkte, Beuth, 1993.
- [2] Landschützer C., Jodin D., Wolfschluckner A.: **Knowledge Based Engineering – an approach via automated design of storage/retrieval systems**. Proceedings in Manufacturing Systems, Vol. 6, Issue 1, 2011
- [3] Landschützer C., Jodin D.: **Knowledge-Based Engineering (KBE) für Hubseiltrommeln**, 20. Kranfachtagung, Dresden, 2012, p.35-52.
- [4] Kartnig, G., Grösel, B.; Zrnic, N.: **Past, State-of-the-Art and Future of Intralogistics in Relation to Megatrends**. Belgrade: FME Transactions, 2012, p.193-200
- [5] Landschützer C., Fritz M., Jodin D.: **Knowledge based engineering and modern CAE for sorting systems**, Proceedings in Manufacturing Systems, Vol. 7, Issue 1, 2012, p.69-76.
- [6] Landschützer, C.; Jodin, D.: **Wissensgestützte Methoden und Werkzeuge zur Geräteentwicklung und -konstruktion in der Technischen Logistik**. Jahrbuch der Logistik. 2014, p.40-45
- [7] Alavi, A.; May, J.; Mohammed, J.: **Application of Computer Aided Design (CAD) In Knowledge Based Engineering**. Proceedings of The 2008 IAJC-IJME International Conference. Nashville: IAJC, 2008
- [8] Verhagen, W. J.C. et al.: **A critical review of Knowledge-Based Engineering: An identification of research challenges**. Advanced Engineering Informatics. Amsterdam: Elsevier, 2012, p.5-15

- [9] Heinecker M.: **Methodik zur Gestaltung und Bewertung wandelbarer Materialflusssysteme**. Dissertation, Technische Universität München: Fakultät für Maschinenwesen, 2006.
- [10] Alcyon Engineering PVT LTD: **Introduction to Knowledge Based Engineering**. Alcyon, 2012 [accessed:07.11.2012]. PDF Format: <http://www.alcyon.co.in/whatiskbe.pdf>
- [11] Richard Curran et al.: **Air Transport and Operations**. Proceedings of the First International Air Transport and Operations Symposium 2010, IOS Press, 2010
- [12] **Umsatz der deutschen Fördertechnik- und Intralogistikbranche im Jahr 2014**. Statista [accessed 28.4.2014]: <http://de.statista.com>.
- [13] Landschützer C.: **Methodeneinordnungsmodell**. [accessed 14.2.2017]: <http://ortner-pichler.at/Methodenmodell>.
- [14] Landschützer C., Jodin D.: **Knowledge-based methods for efficient material handling development**. Progress in Material Handling Research 2012, International Material Handling Research Colloquium, 2012.
- [15] Landschützer C.: **Methoden und Beispiele für das Engineering in der Technischen Logistik**. Habilitationsschrift TU Graz, 2016.
- [16] Landschützer C., Jodin D.: **Engineering for logistics**. Proceedings in Manufacturing Systems, Vol. 11, Issue 2, 2016. p.63-70.
- [17] Arnold D., Furmans K.: **Materialfluss in Logistiksystemen**. Springer, 2007.
- [18] **VDI 2498 Blatt 1**, Vorgehen bei einer Materialflussplanung – Grundlagen, Beuth. 2011.
- [19] ten Hompel M., Nagel L., Schmidt T.: **Materialflusssysteme: Förder- und Lagertechnik**. Springer. 2007.
- [20] La Rocca G.: **Knowledge based engineering: Between AI and CAD. Review of a language based technology to support engineering design**. Elsevier. 2012.
- [21] Chapman C.B., Pinfold M.: **Design engineering – a need to rethink the solution using knowledge based engineering**. Knowledge-Based Systems 12. 1999.
- [22] Schilcher, M.: **Knowledge-based Engineering zur Grobplanung und Analyse von Sortiersystemen**. Master Thesis, Graz University of Technology, 2016
- [23] Ortner-Pichler, A. et al.: **Biomasselogistik – innovative Ansätze zur bedarfsorientierten Versorgung von Hackgutfeuerungen**. 20. Fachtagung Schüttgutfördertechnik, 2015