

# KNOWLEDGE-BASED ENGINEERING IN THE CONTEXT OF RAILWAY DESIGN BY INTEGRATING BIM, BPMN, DMN AND THE METHODOLOGY FOR KNOWLEDGE-BASED ENGINEERING APPLICATIONS (MOKA)

SUBMITTED: December 2020

REVISED: February 2021

PUBLISHED: May 2021

EDITOR: Robert Amor

DOI: [10.36680/j.itcon.2021.012](https://doi.org/10.36680/j.itcon.2021.012)

**Marco Häußler, Dipl.-Ing.,**  
*Chair of Computational Modeling and Simulation, Technical University of Munich;*  
[marco.haeussler@tum.de](mailto:marco.haeussler@tum.de)

**André Borrmann, Prof. Dr.-Ing.,**  
*Chair of Computational Modeling and Simulation, Technical University of Munich;*  
[andre.borrmann@tum.de](mailto:andre.borrmann@tum.de)

**SUMMARY:** Designing railway infrastructure is a knowledge-intensive task. Although there are a number of mature design authoring systems available, their support for dynamically incorporating domain-specific engineering knowledge is very limited. At the same time, a standardized digital representation of railway engineering knowledge (such as building codes and best practice) does not exist. To overcome this deficiency, this paper proposes the use of Knowledge Based Engineering (KBE) to automate routine design tasks by considering multiple knowledge sources. In this scenario, KBE is used to support a Railway design authoring system. To ensure maximum transparency in the design of the developed KBE application, graphical 'Business Process Model and Notation' (BPMN) has been used in combination with 'Decision Model and Notation' (DMN) to formalize the underlying engineering knowledge. The KBE application has been developed according to the Methodology for Knowledge-Based Engineering Applications (MOKA). An evaluation of the BPMN/DMN approach shows that it meets up to 58% of the acceptance criteria found in the literature. In addition, BPMN and DMN can already be used in the early capture phase of MOKA and its workflows can be developed into an executable KBE application in the subsequent phases. The results of the test example discussed here show that time savings of up to 97.5% can be achieved in the execution of the KBE application.

**KEYWORDS:** BIM, Infrastructure, Railway, Design, KBE, BPMN, DMN, MOKA

**REFERENCE:** Marco Häußler, André Borrmann (2021). Knowledge-based engineering in the context of railway design by integrating BIM, BPMN, DMN and the methodology for knowledge-based engineering applications (MOKA). *Journal of Information Technology in Construction (ITcon)*, Vol. 26, pg. 193-226, DOI: [10.36680/j.itcon.2021.012](https://doi.org/10.36680/j.itcon.2021.012)

**COPYRIGHT:** © 2021 The author(s). This is an open access article distributed under the terms of the Creative Commons Attribution 4.0 International (<https://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.



# 1. INTRODUCTION

‘Digital methods like building information modeling (BIM) offer considerable advantages in the construction industry over the conventional methods that still largely prevail in current practice’ (Häußler et al., 2021). With conventional methods, we refer to workflows based on 2D drawings for design, handover to clients and production on site. This applies both to compliance checking, as shown by Häußler et al., as well as to the design of buildings and the associated model creation. Design authoring systems which support object-oriented workflows combined with an parametric modeling approach (also referred to as BIM modelers) are used widely for the design of built structures, see (Amadori et al., 2012; Camba et al., 2016; Rempling et al., 2019; Zou and Feng, 2020). The systems on the market already support design engineers by providing component libraries that also define a certain building logic or interdependencies between components in the software, see also (AKG Software Consulting GmbH, 2021; Autodesk Inc., 2021; Bentley Systems Inc., 2021; CGS Labs, 2021; IB&T Software GmbH, 2021; ProVI GmbH, 2020). In the design of infrastructure, there are numerous such dependencies. The most important element is typically the alignment as nearly all other infrastructure facilities are aligned to it (Reifenhäuser et al., 2018). Railway design authoring systems therefore already provide a framework within which the designer or specialist engineer can work in the course of the design process.

The individual calculation and design steps are often based on knowledge that is already set out in standards and guidelines in the form of rules, see (Häußler et al., 2021). At present, however, the construction industry lacks standardized methods for digitally mapping existing knowledge in such a way that it is machine-readable, so that it can be evaluated and reused, compare (İlal and Günaydin, 2017). The extension of the standardized data model Industry Foundation Classes (IFC) by railway infrastructure objects (IFC rail) recently reached the candidate status (Jaud et al., 2020). While this standardized object-oriented description will provide a good basis for representing and exchanging railway facilities with rich semantics, it does not provide capabilities for representing and applying the knowledge required for designing them.

Especially the digital integration of different knowledge sources is where knowledge-based engineering (KBE) comes in (Stokes, 2001). KBE stands at the intersection of diverse fundamental disciplines, such as artificial intelligence (AI), CAD and computer programming (La Rocca, 2012). KBE systems can be applied to achieve a variety of goals, the most important of which is to increase efficiency by using automated systems for routine tasks, as described, for example, in (Stokes, 2001). ‘Although various research approaches have been developed in the area of KBE since the 1980s, there is no uniform and universally applicable description for the industrial environment, with which a KBE application can be implemented and operated’ (Verein Deutscher Ingenieure e.V., 2017). Häußler et al. present a promising approach to mapping guideline contents in a machine-readable form with the help of ‘Business Process Model and Notation’ (BPMN) in connection with ‘Decision Model and Notation’ (DMN) and demonstrate its use for code compliance checking. In this paper the extent to which BPMN and DMN can be used as basis for a KBE application is investigated.

The paper is structured as follows: Section 2 gives an overview of the state of the art with regard to the terms “knowledge” and “engineering”, and proceeds to discuss KBE as a technological symbiosis of these two domains. Following on from this, the Methodology for Knowledge-Based Engineering Applications (MOKA), as well as BPMN and DMN are presented. Section 3 describes the methodology used in the present study and the separate MOKA phases: Identify, Justify, Capture, Formalize and Package. Finally, Section 4 presents its use in a case study.

## 2. STATE OF THE ART

KBE synthesizes aspects from the domains of “knowledge” and “engineering”. Both are first discussed independently and KBE is presented afterwards.

### 2.1 Knowledge and knowledge management

ISO 30401:2018 defines knowledge as a ‘human or organizational asset enabling effective decisions and action in context’ (ISO 30401:2018, 2018). A common means of representing how knowledge arises is the DIKW hierarchy (**D**ata, **I**nformation, **K**nowledge, **W**isdom) as described in (Rowley, 2007) and depicted in the pyramid in Fig. 1. Information is created by processing and interpreting data. The processing and combination of information creates knowledge, which in turn results in wisdom (Rowley, 2007). Bellinger et al. define the terms as follows:



**Data** represents a fact or statement of event without relation to other things. **Information** embodies the understanding of a relationship of some sort, possibly cause and effect. **Knowledge** represents a pattern that connects and generally provides a high level of predictability as to what is described or what will happen next. **Wisdom** embodies more of an understanding of fundamental principles embodied within the knowledge that are essentially the basis for the knowledge being what it is. Wisdom is essentially systemic' (Bellinger et al., 2004).

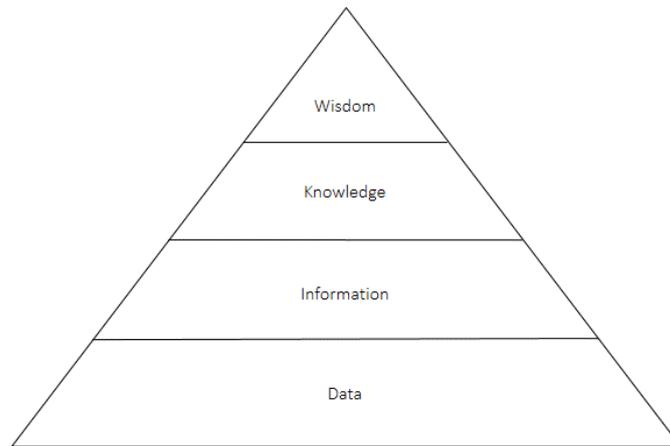


FIG. 1: DIKW hierarchy (Data, Information, Knowledge, Wisdom) as described in (Rowley, 2007)

According to Schreiber et al. (2018) the definitions of data, information and knowledge are widespread, as can be seen for example in Rezgui et al. (2010), Premkumar et al. (2014), Girodon et al. (2015) and Roth et al. (2010). From a philosophical point of view, the term “knowledge” is not as simple or clear-cut to define as Bolisani & Bratianu (2018), among others, have shown, but an epistemological study of the philosophical basis of knowledge is not relevant in the context of this paper. Rather, the observations made here are based on the interplay of data, information and knowledge, since there is consensus on this in the literature (Schreiber et al., 2018). In the literature there is only ‘limited discussion of the nature of wisdom, and even less discussion of the organizational processes that contribute to the cultivation of wisdom’ (Rowley, 2007), as also confirmed by Liew (2013).

Aside from their terminological definition, Roth et al. describe knowledge in terms of type, character, form, location and knowledge quality. Fig. 2 shows relationship between the types of knowledge identified by Roth et al. and lists the different types of knowledge (explicit/implicit, structured/unstructured, etc.).

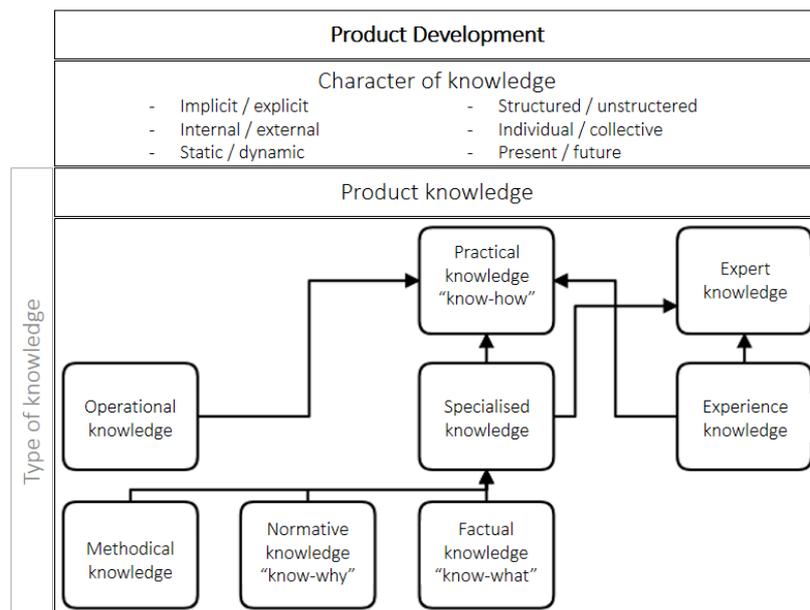


FIG. 2: Types of knowledge based on (Roth et al., 2010)

Storing knowledge and making it available is the subject of many studies. The works of *Haller (2011)*, *Premkumar et al. (2014)*, *Girodon et al. (2015)*, *Cheng et al. (2012)*, *Wu et al. (2012)* as well as research on the topic of “Design Structure Matrices” (DSM) (*Bhaskara, 2010; Tang et al., 2010*) are examples of this. A DSM can help clarify dependencies, connections and interfaces of system elements. The integration of knowledge management and BIM is the subject of the investigations of *Liu et al. (2013)*. *Chassiakos et al.* describe an approach of using a knowledge-based system (KBS) for the maintenance of bridge structures (*Chassiakos et al., 2005*).

## 2.2 Engineering design process

Alongside the concept of knowledge, it is important to understand how engineering and the associated design process works. According to *Ertas*, design can be investigative, creative, rational (logic-based) or decision-oriented (value-based) (*Ertas, 2018*). According to *Calkins et al.* the design of a product is ‘an ordered set of steps that are performed to accomplish a task. The design of a product traditionally proceeds through a series of well-defined stages or phases including:’ (*Calkins et al., 2000*).

- conceptual design (concept exploration and development)
- preliminary design
- detail design (production design)

The design process of products is a gradual and iterative process within which different participants solve different tasks and design individual components of the product (*Obergriesser and Borrmann, 2012*).

## 2.3 KBE

The preceding sections discussed knowledge and the engineering design process independently of one another and placed them in the context of this study. KBE links the two domains and can be used to automate the routine parts of the work steps. ‘A prevalent definition of KBE emphasizes “the capture and systematic reuse of the product and process engineering knowledge” to automate “repetitive and non-creative design tasks” and to support “multidisciplinary design optimization in all phases of the design process”’ (*La Rocca, 2012; Zhao et al., 2015*). *La Rocca* states that ‘Knowledge-based engineering (KBE) is a technology based on the use of dedicated software tools called KBE systems, which are able to capture and systematically reuse product and process engineering knowledge, with the final goal of reducing the time and costs of product development’, see also (*Stokes, 2001*). In contrast to KBS, as used in the context of KM, KBE applications have the capacity to influence the geometries of a design, see also (*La Rocca, 2012; Stokes, 2001*). ‘The basic objectives that have to be supported by KBE are: solve a particular design problem using a KBE application (short-term), and retain the domain knowledge required for solving design problems in the same domain (long-term)’ (*Bermell-García and Fan, 2002*).

According to *La Rocca* and *Cooper et al.* KBE systems have “generative” and “integrated” modelling capabilities (see Fig. 3), which means that ‘a set of input values is assigned to the parameters that are used in the product model, the KBE system applies the rules to process the input values and, finally, the engineered design is generated, with little or no human intervention’ (*La Rocca, 2012*). ‘A generative model differs from a geometric model, which is a typical output of advanced CAD systems. Where a geometric model is a model of a designed product with fixed features (dimensions and configuration), a generative model is a generic representation of the product. The generative model [...] is built on the basis of a geometric model and is enhanced by the engineering rules that determine its design’ (*Skarka, 2007*).

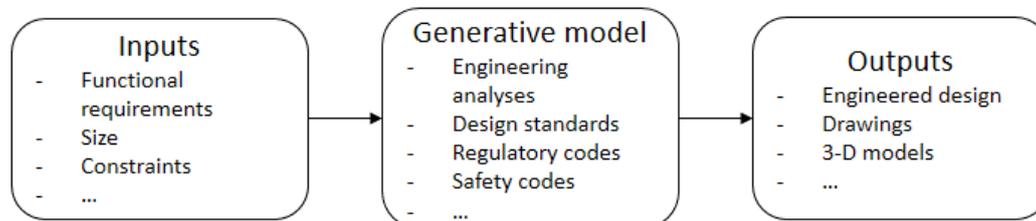


FIG. 3: The generative model of a KBE application takes input specifications, applies relevant rules and automatically produces an engineered design (based on (*La Rocca, 2012*))

KBE systems can either be integrated directly into a design authoring system or be kept separate from the design authoring system and coupled via interfaces, as shown in Fig. 4.

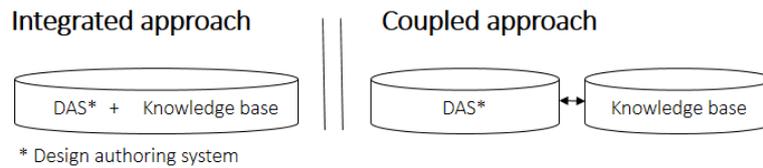


FIG. 4: Integrated and coupled approach of KBE and Design authoring system (Verein Deutscher Ingenieure e.V., 2017)

There are numerous practical examples of the development of KBE applications. These range from KBE systems in the field of aerospace (Bermell-Garcia et al., 2012; Corallo et al., 2009; Steenhuizen and Van Tooren, 2012) or aerodynamics (Bermell-García and Fan, 2002; Chiciudean et al., 2008), for cost estimation (Yildiz et al., 2014; Zhao et al., 2015), building design (Singhaputtangkul et al., 2013), building operation (Dibley et al., 2011; Motamedi et al., 2014; Motawa and Almarshad, 2013), building analysis (Kim et al., 2013; Wang and Leite, 2016), sustainability (König et al., 2013), safety planning (Zhang et al., 2015), structural design (Cavieres et al., 2011; Rempling et al., 2019) and bridge modelling (Flurl et al., 2015; Sandberg et al., 2016; Singer, 2014; Singer et al., 2016; Singer and Borrmann, 2015). *Johansson et al.* present an approach to representing the knowledge contained in a KBE application using graph theory (Johansson et al., 2018; Vilgertshofer and Borrmann, 2017). The ability of a KBE system to support decisions is particularly important, as the investigations of *Chen et al.* (2010), *An et al.* (2020), *Egemen & Mohamed* (2008) and *Wang et al.* (2019) show.

*Garcia & Ip-Shing* (2002) and *Sainter et al.* (2000) state that long-term problems in development of KBE systems exist. These are:

- Knowledge loss, due to poor modelling of the applications
- Knowledge loss, due to the development language
- Knowledge misuse, due to incorrect selection of the applications being developed
- Increased maintenance costs, due to the lack of standardization of applications, and
- Knowledge underutilization, due to the difficulties in sharing and reusing knowledge.

According to *Sainter et al.*, many of these long-term problems can be mitigated using standardized development and management methods, languages and frameworks (Phillip Sainter et al., 2000).

According to *La Rocca* (2012) and *Tripathi* (2011) KBE systems are also referred to as expert systems. *Tripathi* defines the components of a KBE system as follows (see also Fig. 5): 'A rule-based expert system contains a knowledge base, inference engine, knowledge acquisition, explanation facility and user interface' (Tripathi, 2011).

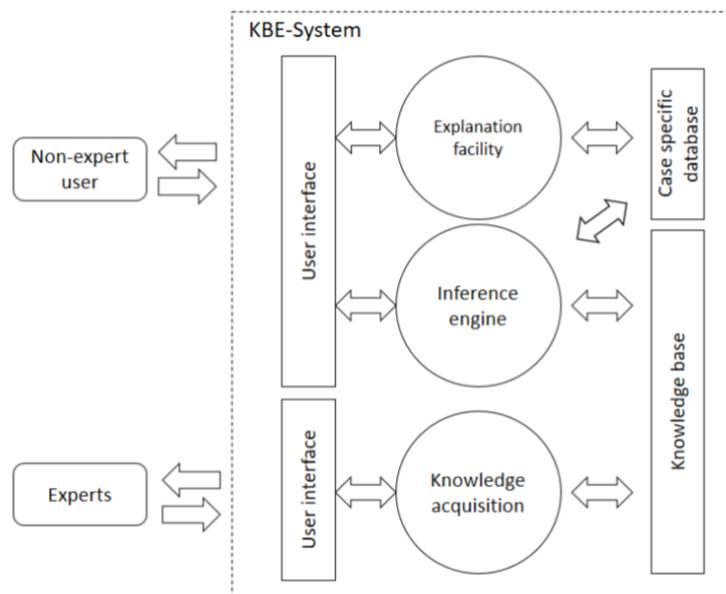


FIG. 5: Structure of a KBE system according to Tripathi and Singer (Singer, 2014; Tripathi, 2011)

## 2.4 MOKA

MOKA was developed to reduce the risks and investment costs associated with the development of KBE systems (Stokes, 2001). The method is a modular system consisting of six steps, as shown in Fig. 6.

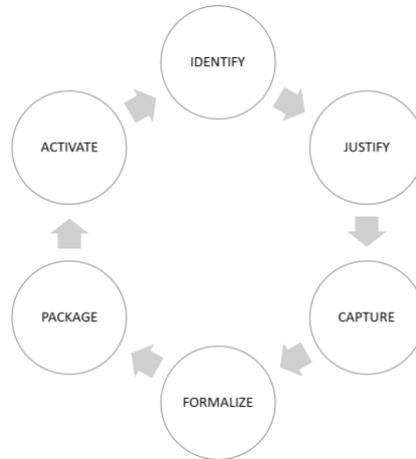


FIG. 6: The KBE lifecycle (Stokes, 2001)

The contents of the individual steps are described in detail by *Stokes*. In this paper the focus lies on “Identify”, “Justify”, “Capture”, “Formalize” and “Package”.

MOKA provides a method for collecting and structuring knowledge from different sources (Capture) and transferring it step by step into a formal representation (Formalize), so that a KBE application can be developed from it (Package), as illustrated in Fig. 7. The methodology also indirectly supports communications between the active participants (e.g. expert, knowledge engineer, software developer). Through its generic approach, the modules of the MOKA process, such as notations or forms of presentation, can be exchanged or supplemented by equivalent elements if necessary.

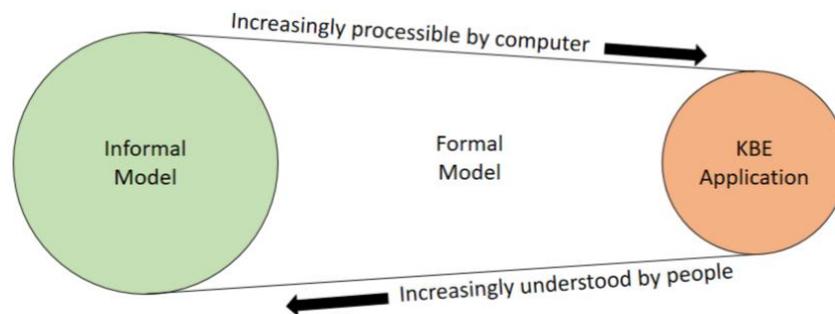


FIG. 7: Assessment of the informal and formal model of human readability and computer processability (Stokes, 2001)

MOKA is used in numerous studies as a method for the development of KBE applications. There are examples in aerospace (Emberey et al., 2007), mechanical engineering (Hunter et al., 2006) and architecture (Montali et al., 2019, 2017), and MOKA is also used in connection with inspection and manufacturing (Álvarez et al., 2010; Barreiro et al., 2009; Martínez et al., 2012). In certain fields of application, MOKA has gaps, which have been identified in various investigations and closed by domain-specific extensions, for example for the automated planning of processes as MOKA understands a product model as a physical object (Ammar-Khodja et al., 2008; Helguson and Kalhori, 2012). *Chan* has extended the method to develop simulation workflows automatically with the help of KBE mechanisms (Chan, 2013). *Skarka* has used Protégé to develop knowledge ontologies based on MOKA, see (Skarka, 2007, 2006). Due to the use of MOKA in aerospace and mechanical engineering, Dassault CATIA is often used as a software package for developing KBE applications using MOKA, see (Lohith et al., 2013; Skarka, 2007; Zheng et al., 2012).

*Sandberg* has examined literature in the field of KBE, MOKA and construction and concluded that while individual MOKA phases are adopted, concrete details on their implementation are usually lacking: “[...] Design automation research focuses too much on the application itself rather than describing the way the application was developed” (*Sandberg*, 2015). Among other things, the definition of acceptance criteria for the successful implementation of a KBE application is criticized. With the help of acceptance criteria the success or failure of the KBE application shall be judged (*Stokes*, 2001).

The literature research revealed that the MOKA method is often used to develop KBE applications, but the examples were predominantly hard-coded “black box” solutions. MOKA aims to minimize the language barriers between Experts, Knowledge Engineers and Software Developers, but MOKA focusses only up to the formal model and describes these steps detailed. The package and activation step is not in focus of MOKA (*Stokes*, 2001).

For the present study, the advantages of the basic principles (collecting and structuring knowledge) of MOKA outweigh the disadvantages (need for domain-specific extensions). The literature research shows that MOKA is generic and extensible, which means that the disadvantages of pure MOKA are manageable. MOKA is therefore elected to be used but instead of developing a “black box” solution, an approach based on a graphical notation is employed, as described in the next section.

## 2.5 BPMN and DMN

A primary objective of the MOKA method is the creation of the formal model, consisting of a product and activity model. *Stokes* recommends the Unified Modeling Language (UML) for the development of product models and the corresponding UML activity diagrams for the design process model (*Stokes*, 2001). The focus of this research lies on the presentation of the design process model. MOKA does not detail the transfer of the formal model into an executable KBE application in any detail and only recommends converting the formal model into a neutral data format such as Extensible Markup Language (XML) to standardize the transformation process. In contrast to the MOKA recommendation, this study instead looks at how BPMN can be used in place of UML activity diagrams.

‘An activity diagram is a UML behavior diagram which shows flow of control or object flow with emphasis on the sequence and conditions of the flow. The actions coordinated by activity models can be initiated by other actions finishing execution, by objects and data becoming available, or by the occurrence of some event external to the flow’ (UML Activity Diagrams, 2020).

Alongside the UML activity diagrams, workflows can also be formally represented with the help of the BPMN. BPMN is an international standard (ISO/IEC 19510) that is maintained by the Object Management Group (ISO/IEC 19510:2013, 2013). *Recker et al.* describe BPMN as a ‘[...] structured, coherent and consistent way of understanding, documenting, modeling, analyzing, simulating, executing, and continuously changing end-to-end business processes and all involved resources in light of their contribution to business performance’ (*Recker et al.*, 2006). ‘BPMN gained popularity because it is a standardized graphical notation, adopted by the Object Management Group (OMG), and it is easy to understand, also for non-IT experts. Since version 2.0, BPMN also includes execution semantics and a standardized XML-based syntax’ (*Chan*, 2013).

*Chan* compares various process modelling languages and evaluates them in the categories ‘graphical notation’, ‘designed for execution’, ‘standardized’, ‘widely adopted’ and ‘easy to use’ (see also Table 1).

TABLE 1: Trade-off of various workflow modelling languages; score: 2 plus points (++), 1 plus point (+), no point (0), 1 minus point (-), 2 minus points (--) (*Chan*, 2013).

Criteria	Weight	UML	BPMN
Graphical notation	1	++	++
Designed for execution	1	-	+
Standardized	1	++	++
Widely adopted	1	++	++
Easy to use	1	+	+
<b>Score</b>	<b>Max: 10</b>	<b>6</b>	<b>8</b>

*Chan's* results show that BPMN is better suited than UML activity diagrams due to the ability to make workflows executable with the help of a corresponding workflow engine. A workflow engine 'performs the process step by step in accordance with the modelled logic and can react to events during runtime as well as trigger events itself. Such an event might be a data input by the user, the calculation of mathematical formulae, a decision resulting from an if-then condition, or the execution of text-based source code. By virtue of its ability to automate the developed processes by means of a workflow engine, BPMN can be classified as belonging to the category of visual programming languages' (Häußler et al., 2021). In practice, however, *Chan* only uses BPMN for visualizing the workflow but does not make it executable.

BPMN can be used in many ways. Recker states that "classical" process management applications such as documentation, redesign, continuous improvement and knowledge management dominate the application areas of BPMN, while more technical application areas such as software development, workflow management or process simulation are not (yet) widespread' (Recker, 2010). In the AEC industry, BPMN is used for the development of IDMs (Aram and Eastman, 2010; Obergriesser and Borrmann, 2012; Weise et al., 2008), which is also recommended by (ISO 29481-1:2016, 2016). *Alreshidi* combined UML diagrams with BPMN workflows to develop a cloud platform (Alreshidi et al., 2016). *Dimyadi et al.* and *Häußler et al.* used BPMN with corresponding workflow engines to verify building data models in the context of code compliance checking (Dimyadi et al., 2016, 2014; Dimyadi and Amor, 2017; Häußler et al., 2021).

It is specifically the executability of BPMN workflows that set them apart from other methods, as *Borrmann et al.* confirm: 'In particular, the option to use BPMN in the context of workflow management systems as an implementation language [...] is an important criterion for use in construction projects' (Borrmann et al., 2017a). 'The most frequently used BPMN elements for representing business processes in the AEC industry were sequence flow, pool, lane, task/activity (exclusive), gateway, and message flow, which are all basic BPMN modelling elements' (Park et al., 2011). A graphical representation of the individual elements is shown in Fig. 8.

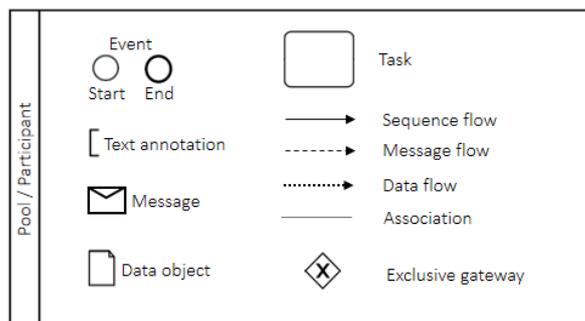


FIG. 8: Proposal for an 'Essential Subset' of BPMN for IDM Development, based on (Park et al., 2011).

As discussed in Section 2.3, KBE applications must have the capacity to support decisions or even make decisions independently. To this end, DMN augments BPMN. 'The primary goal of DMN is to provide a common notation that is readily understandable by all business users, from the business analysts needing to create initial decision requirements and then more detailed decision models, to the technical developers responsible for automating the decisions in processes, and finally, to the business people who will manage and monitor those decisions. DMN creates a standardized bridge for the gap between the business decision design and decision implementation. DMN notation is designed to be usable alongside the standard BPMN business process notation' (Object Management Group, 2019). With the help of Decision Engines (comparable with Workflow Engines) DMN decision tables can be made executable. Through the integration of DMN decision tables in BPMN workflows, decision paths can be incorporated directly into the workflow engine, see also (Häußler et al., 2021).

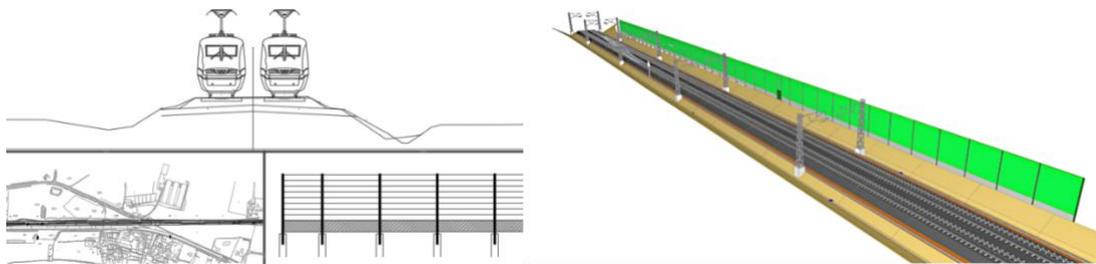
With the help of BPMN and DMN, knowledge, processes and associated decision paths can be represented graphically and thus made more readily understandable than in text-based languages. The special feature of both notations compared to UML is their executability. Compared to MOKA, where the formal model, once complete, has to be manually translated into a specific KBE language by a software engineer, BPMN and DMN do not require any further translation or transformation into text-based code. Instead, the graphical notation can be used directly to develop the informal model in order to visualize process sequences. In the course of developing the formal model, and ultimately also the KBE application itself, the first draft can be developed further without having to switch between different forms of presentation.

## 2.6 Designing in track construction

The aim of this study is to automate design steps in the field of rail infrastructure design with the help of KBE. The focus lies on design of new buildings. According to *Borrmann et al.*, a fundamental distinction must be made between explicit and implicit modelling when creating three-dimensional models. ‘Explicit modelling, [...] describes a volume in terms of its surface [...]. Implicit modelling by contrast employs a sequence of construction steps to describe a volumetric body, and is therefore commonly termed a procedural approach’ (Borrmann et al., 2017b). Of particular relevance for this study is parametric modelling, which *Borrmann et al.* describes as a possible implicit approach. ‘Feature-based parametric CAD is currently the industry standard technology to create geometric models and assemblies, and is widely used across many engineering fields’ (Camba et al., 2016). The importance of parametric modelling is also described by Tang et al. (2020), Brown et al. (2020) and Zou & Feng (2020).

For this research the term “design authoring system” is used as a synonym for software which provides capabilities for the object-oriented and parametric design of geometric models in context of railway infrastructure facilities. The term BIM is used as a synonym for the object-oriented workflow during the building lifecycle, it is not considered as a specific software product. The tools used for this research are presented in Section 3.5.

The definition of the parametric 3D-model in software products like Civil 3D (Autodesk Inc., 2021), OpenRail Designer (Bentley Systems Inc., 2021), card\_1 (IB&T Software GmbH, 2021), Vestra Rail (AKG Software Consulting GmbH, 2021), ProVI (ProVI GmbH, 2020) or Ferrovial (CGS Labs, 2021) is managed with the help of a ‘drawing-oriented view – split into site plan, cross-section and elevation [see also Fig. 9]. This type of model is referred to as an implicit geometry description. With implicit models, the governing design parameters become significantly more accessible than with explicit models’ (Häußler and Borrmann, 2020). The description of the design of railway infrastructure in the underlying guidelines is predominantly parameter-oriented (Häußler et al., 2021).



*FIG. 9: Comparison of implicit and volumetric 3D models: while implicit models (drawing-oriented view) are used during the design process (parametric modeling), explicit models are used in the context of BIM-based analysis (Häußler and Borrmann, 2020).*

‘The procedural approach provides the user of the system with the possibility to easily modify an existing model by going back in the construction history and adapting the corresponding parameter [...]’ (Borrmann et al., 2014). This is a particular advantage in the context of KBE. For the automation of design steps, the use of a design authoring system which supports the parametric modeling approach is indispensable, as this offers the greatest degree of freedom compared to other modelling philosophies, especially explicit modelling.

## 3. METHODOLOGY

In this paper, a KBE application is developed based on the BPMN and DMN notations standardized by the OMG. The intention is to minimize the break that arises when developing an informal model into a finished KBE application using the MOKA method. BPMN and DMN are used not just to represent technical knowledge but also process flows graphically, making them transparent for those involved in development as well as for later users. The ability to make BPMN and DMN executable using a workflow engine makes this approach interesting for the development of KBE applications.

This section describes the following steps of the MOKA method: Identify, Justify, Capture, Formalize and Package.

### 3.1 Identify

This paper focuses on the development of a KBE application to support the design of new buildings along railway infrastructure. The suitability of BPMN and DMN for representing guidelines in the railway design section was discussed in earlier work by Häußler *et al.* which concluded that up to 68 % of the examined rules can be digitally mapped using BPMN and DMN. A commercially available Design authoring system was used, and the KBE application aims to support the work steps of an engineer using this specialist software.

The following key sources of knowledge can be identified:

- Expert knowledge (human sources)
- Contents of guidelines (documents)
- Requirements and modes of operation of the computer application (computer files)

The classified objectives for the development of a KBE system are as follows:

- The structure of a KBE system is fulfilled (very important)
- Use of a graphical notation to provide transparency (very important)
- Use of a standardized development language (very important)
- Combination of different knowledge sources is possible (important)
- Combination with commercially available Design authoring systems, the KBE system generates parameters for the Design authoring system which is done usually by users (important)
- The KBE system decreases design time and ensures design quality (important)

### 3.2 Justify

The lack of universal definitions for acceptance criteria has been criticized in the literature, most notably by Sandberg. Stokes recommends the definition of (acceptance) criteria to verify the success or failure of KBE systems. As a result, criteria are compiled and clustered for the successful implementation of KBE applications in the course of literature research. These were then used to evaluate the approach of a BPMN and DMN-based KBE system developed in this study. Taking the objectives of this work into account not all criteria are necessarily equally important, the criteria were additionally weighted. The criteria which meet the very important objectives are weighted three times, the important ones two times and the one which are non-important compared to the objectives are weighted with one.

The criteria, their weighting, the corresponding sources, comments and evaluation result are shown in Table 2. The evaluation was carried out as follows:

- Requirement not fulfilled “--” (2 minus points)
- Requirement is not sufficiently met “-” (1 minus point)
- Requirement is largely met “+” (1 plus point)
- Requirement is fully met “++” (2 plus points)

TABLE 2: Criteria for the successful integration of BPMN and DMN into KBE development

Criteria	Source	Note	Evaluation	Weighting
<b>System configuration</b>				
Separate knowledge base	(Bermell-Garcia et al., 2012; La Rocca, 2012)	BPMN and DMN	+	3
Separate reasoning mechanism/ inference engine	(Bermell-Garcia et al., 2012; La Rocca, 2012; Singer, 2014; Tripathi, 2011)	Workflow engine	+	3
Provides user interface	(Singer, 2014; Tripathi, 2011)	Graphical representation and possibility to interact with webpages via interfaces	++	3
Provides explanation facility	(Singer, 2014; Tripathi, 2011)	Graphical notation and reporting of process steps is possible	++	3
Capability to capture, formalize and provide knowledge	(Bermell-Garcia et al., 2012; Brimble and Sellini, 2000; Singer, 2014; Tripathi, 2011; Van Der Velden et al., 2012)	Process knowledge with BPMN, decision support with DMN	++	3

Criteria	Source	Note	Evaluation	Weighting
<b>Supported Features</b>				
Support of different rules types	(La Rocca, 2012)			
- logic rules	(La Rocca, 2012)	Possible: JavaScript and java available	++	2
- math rules	(La Rocca, 2012)	Possible: JavaScript and java available	++	2
- geometry manipulation rules (parametric rules)	(La Rocca, 2012)	The parametric rules of the Design authoring system cannot be manipulated, since the KBE system is not integrated. The geometric rules have to be developed in the coupled KBE system.	--	1
- configuration selection rules (or topology rules; combination of mathematical and logic rules)	(La Rocca, 2012)		++	2
- communication rules (interfaces)	(La Rocca, 2012)	Possible, but interfaces have to be implemented	+	2
Reconfig. rules and outputs based on new inputs	(Van Der Velden et al., 2012)	Possible, but only manually	-	1
Derive new rules automatically from old rules based on input changes	(Van Der Velden et al., 2012)		--	1
Intelligently control rule sequencing and execution	(Van Der Velden et al., 2012)		++	1
Provides a mediating language between experts, knowledge engineers and software developers, simple yet expressive and preferably graphical	(Brimble and Sellini, 2000)	Graphical notation	++	3
Expert/end user involvement	(Bermell-Garcia et al., 2012)		+	2
User extensible and customizable	(Brimble and Sellini, 2000)		+	1
Formal enough to support code generation	(Brimble and Sellini, 2000)	BPMN and DMN are developed to be executable	++	3
Allow models to be organized according to local style	(Brimble and Sellini, 2000)		+	1
Automate processes in a product development lifecycle	(Van Der Velden et al., 2012)	Is possible using workflow engine	++	2
Verify designs against standards	(Van Der Velden et al., 2012)	Verification is limited to the designed process, no additional checks are executed automatically	-	1
Handle new known and unknown problems	(Van Der Velden et al., 2012)	Unknown problems cannot be solved	-	1
Provide high level commands that invoke a number of sub-processes	(Van Der Velden et al., 2012)	Due to the representation of workflows, it is possible to invoke tasks and sub-processes	++	1
Adequate definition of activities	(Stokes, 2001)		++	3
<b>Benefits of KBE</b>				
Decrease time and costs for new product development	(Bermell-Garcia et al., 2012; Corallo et al., 2009; Stokes, 2001; Van Der Velden et al., 2012; Verhagen et al., 2012)		++	2

Criteria	Source	Note	Evaluation	Weighting
Improve quality of design analysis, decisions and ensure consistent quality of outputs	(Bermell-Garcia et al., 2012; Corallo et al., 2009; Van Der Velden et al., 2012)		+	2
<b>Documentation</b>				
Personalization and codification of knowledge, “learning by doing” (the knowledge must be personalized: it must be geared towards the end user(s), who must be able to retrieve, understand and if necessary, update the knowledge used for design and analysis)	(Bermell-Garcia et al., 2012)	Graphical representation makes it possible to convey knowledge to the end user in an understandable manner. It is also possible to update the knowledge easily. It is not possible for the system to update knowledge automatically.	+	1
Users can inspect the steps in the design or analysis process, and can see the associated knowledge through the related knowledge component(s)	(Bermell-Garcia et al., 2012)		++	3
Consistency of representation	(Stokes, 2001)		++	2
Methodology worthy becoming a standard	(Stokes, 2001)	BPMN and DMN are already standards	++	3
Provision of case reports (input, output and used knowledge is listed)	(Bermell-Garcia et al., 2012; Stokes, 2001)	Standardized reports have to be implemented in the system	+	1
Goes beyond “black box” processes and applications by supporting categorization, accessibility, traceability and subsequent sourcing of knowledge	(Bermell-Garcia et al., 2012; Sandberg, 2003)		++	2

A total of 31 evaluation criteria were identified in the course of literature research and have been applied to evaluate the approach tested in this paper. Comparing the evaluated score of 36 points against the maximum achievable score of 62 points, the approach discussed in this paper fulfils 58% of the requirements that are not weighted against each other. Taking into account the weighting of the criteria applied for this study, the BPMN/DMN approach in this study fulfils 73 % of the requirements, which can be considered a good value due to the large number of different requirements.

### 3.3 Capture

‘The capture step involves the collection of raw knowledge and transforms it into the first level of MOKA representation – the Informal Model’ (Stokes, 2001). After the sources of knowledge have been identified (see Section 3.1), the knowledge they contain must be compiled, analyzed and put into context. According to *Roth et al.* there are different types of knowledge. As described by *Tripathi* (2011) knowledge can be collected using various methods, for example using interview techniques, through text analyses, observation techniques and review techniques.

The types of knowledge were systematically examined with the help of interviews. Table 3 shows the structure of the interviews conducted on the respective knowledge types. Open, unstructured interviews were conducted with domain experts, but also with the software developers of the selected design authoring system. The suitability criteria of the domain experts are described in Section 4.2.

In addition to the results of the interviews, the interviewees provided additional documents that can be considered as further sources of knowledge.

The objective of the interviews with experts was to gather existing technical knowledge for the proposed KBE application. This includes methodological, normative and fact-based knowledge, but also knowledge from experience gained. No pre-existing documents could be provided for methodological knowledge as well as knowledge from experience. In order to make existing knowledge usable for the present study, a scenario for a designing task was developed and the experts were asked to explain the processing and solution path of this step by step. This explanation served as the basis for a first version of a BPMN diagram.

TABLE 3: Structure of interviews and knowledge sources

Type of knowledge			Interview	Additional documents/tools
Practical knowledge ('Know-how')	Specialized knowledge	Methodical	Domain Expert	
			Software developer	Descriptions concerning Design authoring system, Design authoring system
		Normative ('know-why')	Domain expert	Deutsche Bahn AG Guidelines
		Factual ('know-what')	Domain expert	Deutsche Bahn AG Guidelines
	Experience knowledge	Domain expert	–	
Operational knowledge	Software developer	Descriptions concerning Design authoring system, Design authoring system		

The interviews with the software developers served to understand the software used. They provided insights into the methodical procedure during modelling, but also into the interfacing requirements for connecting the design authoring system to the proposed KBE application (operational knowledge).

The documents made available were then screened and analyzed, and the designing tasks to be solved with the help of KBE were extracted and grouped into knowledge types. Most of the guidelines examined describe the “know-why” and “know-what” while the documents for the Design authoring system describe the operational knowledge.

Based on the steps performed, one can state that the design of railway infrastructure and its associated structures is very strongly parameter-oriented and that numerous geometric dependencies (vertical distance, horizontal distance, height, length, etc.) exist between the individual component and/structures, see also (Häußler et al., 2021). The most important basis for the design is the alignment, which serves as the defining element for all associated structures. To determine the dependencies between components and assemblies, DSM were developed in conjunction with the experts.

An essential aspect for structuring the collected knowledge is the so-called ICARE forms (Illustrations, Constraints, Activities, Rules and Entities) in MOKA (see Fig. 10). These can be used to represent the informal model. All the information for proceeding forward towards the KBE application is thus available. In the study, special attention is paid to how the Design authoring system and the KBE system can be connected and which inputs and outputs are understood or will be generated.

Form	Rule	
<b>Name</b>		
<b>Reference</b>		
<b>Objective</b>		
<b>Context, information, validity</b>		
<b>Description</b>		
<b>Related activities</b>		
<b>Linked rules</b>		
<b>Linked constraints</b>		
<b>Related entities</b>		
<b>Related illustrations</b>		
<b>Information origin</b>		
<b>Management</b>	<b>Author</b>	MH
	<b>Date</b>	15.05.2020
	<b>Version number</b>	1.0
	<b>Status</b>	Final

FIG. 10: An (empty) example of a rule form containing, for example, general information (name, reference number), rule description, and links to other ICARE forms

### 3.4 Formalize

MOKA distinguishes in the formal model between “Product Model” and “Design Process Model”. These are derived from the ICARE forms and both are intended to be represented with the help of UML diagrams. This means, however, that in the next “Package” step, the newly created formal model has to be translated into a KBE language because UML is not developed for direct execution. The Design authoring system used is already a domain-specific specialist application. The UML diagrams serves to visualize the underlying product model and as a basis from which to develop the KBE application. This study, however, focuses on automating the design steps, which can be represented by a “Design Process Model”. This is created by the expert in the “Capture” phase and used and developed to support the transformation process from the informal to the formal model as well as to be transparent for the experts involved. To develop the process model, the contents of the ICARE forms are systematically transferred into BPMN and DMN. In a first step, the Activity forms are evaluated in combination with the created workflow diagrams. These contain links to the Rule and Entity forms. This procedure has been corroborated by *Stokes (2001)*.

In the course of the expert interviews, numerous references to directives in the railway design sector were communicated. The requirements were documented and recorded in the ICARE forms. For the most part, these guidelines are only available in human-readable form and the requirements are presented in various forms, e.g. continuous text, graphics, tables and formulae, see also (*Häußler et al., 2021; Preidel and Borrmann, 2015*). To translate them into a machine-readable language, the RASE syntax (**R**equirement, **A**pplies, **S**elect, **E**xception) was used. This has been described comprehensively by *Hjelseth & Nisbet (2011)* and *Häußler et al.* also used it to translate the corresponding guidelines. This process has been adapted in the present study using the BPMN elements script task, parallel and exclusive gateway, sequence flows and DMN tasks. Sub-processes are used to visually summarize related activities. As far as multiple but similar activities are included, the sub-processes are designed so that they can be executed several times with the help of loop criteria.

Since the focus is the automation of the design process, it is important to depict not just the technical content of the knowledge sources, but also the methodical approach of the experts and the necessary procedure within the Design authoring system through the BPMN diagrams. An example BPMN diagram is shown in Fig. 11.

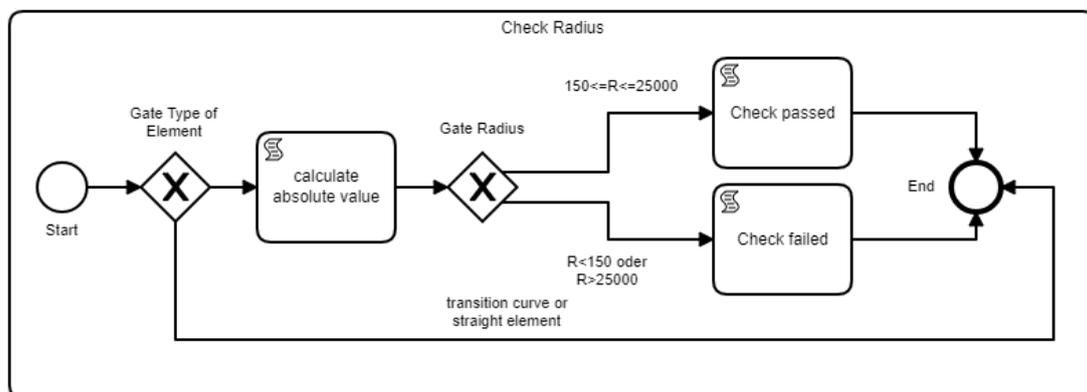


FIG. 11: BPMN-Process in the context of code compliance checking (*Häußler et al., 2021*)

The A (Activities) and R (Rules) forms contain appropriate information. The main objective of the KBE application is to process data from upstream processes in such a way that new data is generated through the execution of the KBE application which can then be made available to subsequent processes (see Fig. 3). As the aim is to support the user, the interface of the KBE application has been kept to a minimum so that only native data of the Design authoring system is used. Although the interfaces are defined during the formalization process, they are not developed. This is the content of the “Package” step.

The resulting “Design Process Model” was then shown to the experts for joint verification of correctness (see Section 4.3).

The advantage here is that no further steps are required to translate the created workflows and decision tables into a neutral data format such as XML, as these are already saved as XML data. As such, this step from MOKA is not required.

### 3.5 Package

In the “Package” step, the formal model is usually translated by a software developer into the programming language of the target platform. Ideally, the formal model saved as XML is used as a basis. Since BPMN and DMN are already designed as executable languages and are stored in XML format, no further translation is required. The package step is therefore used to test the functionality and interaction of the applications used. In particular, the definition of variables is validated and adjusted if necessary.

In addition, a web-based user interface was developed as a direct connection to the KBE system along with the necessary interfaces to the Design authoring system.

The individual components of a KBE application were described earlier in Section 2.3. In our case, the following software components were used (see also Fig. 12):

**Knowledge base:** Camunda Modeler (Camunda Services GmbH, 2018)

**Inference engine:** Camunda Community Platform (workflow engine) (Camunda Services GmbH, 2018)

**Explanation facility:** Web services – webpage and Camunda Community Platform (Camunda Services GmbH, 2018)

**User interface:** ProVI 6.2 (ProVI GmbH, 2020) and webpage (interface to KBE system)

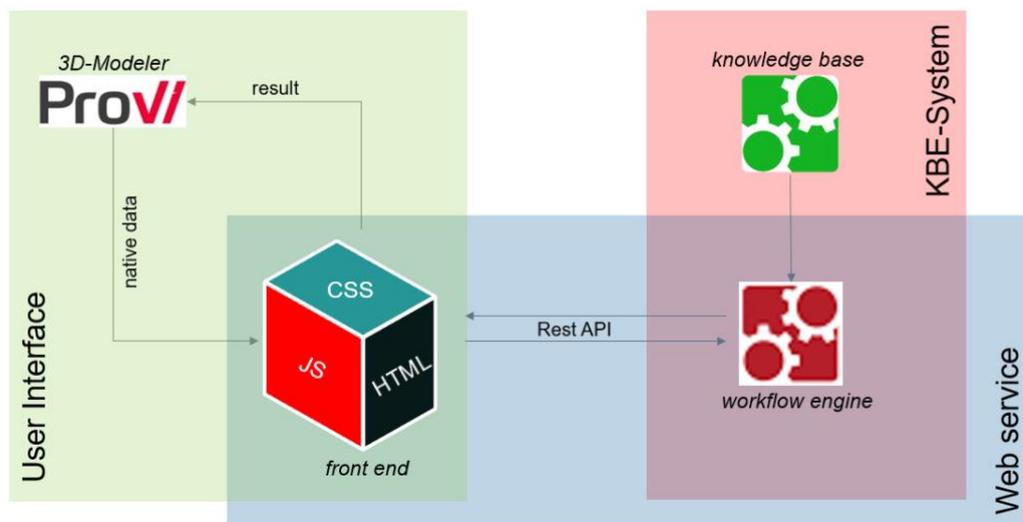


FIG. 12: Software configuration

The communication between the user interface and the KBE system running on a server is facilitated by a Rest API.

## 4. CASE STUDY: THE DESIGN OF NOISE BARRIERS

### 4.1 Identify and Justify

A key aspect in the design of railway facilities is to reduce the noise produced by rail traffic (Beier et al., n.d.; Deutsche Bahn AG, 2020), for example by erecting noise barriers alongside railway lines. The construction and the individual components of noise barriers are highly standardized, which lends itself towards automation of the design process. The KBE application proposed here as a case study is intended to support the geometric design of a noise barrier by automating the steps that are carried out manually. Existing automation steps already available in specialist applications do not need to be replicated. For example, the specialist app already models the noise barrier as a parametric model so that the KBE application only needs to determine the required input parameters. The KBE applications should nevertheless be generic enough so that different variants of the same noise barrier can be generated, but also different noise barriers for different projects and project areas.

The software configuration has already been described in Section 3.5, and the acceptance criteria for this study were evaluated in Section 3.2. The evaluation discussed there applies unchanged for the specific task at hand.

## 4.2 Capture

For knowledge acquisition, five domain experts from different companies were consulted in order to ensure a cross-company approach. The domain experts have more than 5 years of professional experience and are classified as senior engineers in their companies. In addition, a software developer of the Design authoring system employed was interviewed for the capture phase.

The interviews took between two and three hours per domain expert (only one expert per interview). The experts were asked to explain their design workflow without any interposed question first to get an unaffected description. The experts described the constraints in design, used input data and desired outcome. All of the experts use the same Design authoring system and expressed that no comparable system for the design of noise barriers is known to them. The parametric logic of a noise barrier was described as well as the interdependency to other facilities along railway tracks. In a second step, the idea of a KBE system was introduced by the knowledge engineer and the experts were asked which part of the design process is worth to automate. The experts provided different documents concerning the design of noise barriers. For this study the same Design authoring system was used as the one indicated by the domain experts. The interview with the software developer focused on the operation of the Design authoring system, provided interfaces were discussed as well as possible extensions for the software.

Based on the interviews, the relevant guidelines were reviewed.

Various disciplines are involved in the design of a noise barrier, such as sound calculations, structural analysis and geometric design. The interviews showed that the result of noise calculations is a key input variable for the geometric design of noise barriers. The calculation is carried out by environmental engineers using specialized software and essentially provides information on the height and length of a noise barrier. The structural analysis is a downstream process for the geometric design of the noise barrier and is carried out by structural engineers. The aim of the test case outlined here is to support the design engineers in the routine tasks of geometric design.

The main results of the knowledge acquisition are summarized below. A noise barrier consists of four main components: post (1), wall element (2), base (3) and foundation (4), as shown in Fig. 13. In addition, the noise barrier is dependent on various input variables. The distance from the front edge of the noise barrier relates to the alignment of the track as set out in Deutsche Bahn guideline 800.0130 (DB Netz AG, 2018). An explanation is given in Section 4.3.2.

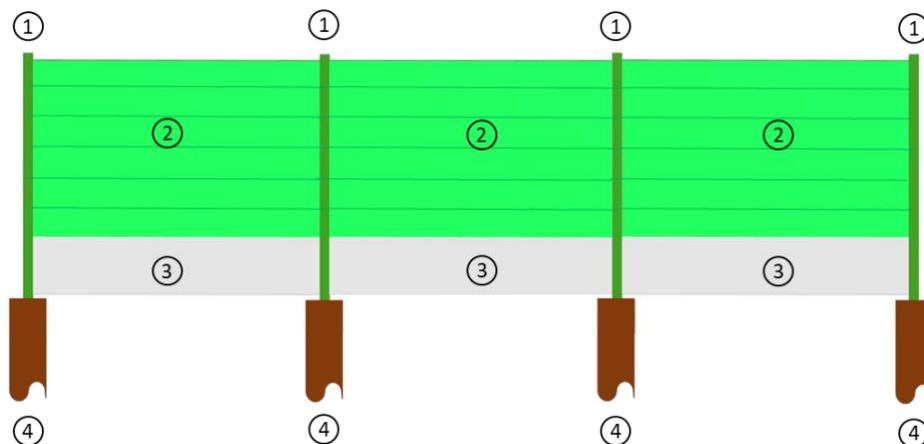


FIG. 13: Design of a noise barrier

Various structures exist along railway tracks that influence the design of noise barriers. These include drainage, underground cable conduits, overhead line systems and components for signaling and safety technology, as shown in Fig. 14 situation 1. These can obstruct the continuous construction of a noise barrier parallel to the track (Fig. 14 situation 2). Table 4 shows the interdependencies at the level of the installation in the form of a DSM. For all the subsections considered, the alignment therefore serves as an input variable, and for the noise barrier in particular, the subsections listed are likewise input variables. The design of a noise barrier must therefore be able to respond flexibly to other subsections by means of so-called by-passes that avoid collisions with installations from other subsections. Fig. 14 situation 3 shows how that noise barrier steps back to avoid colliding with a catenary mast and manhole.

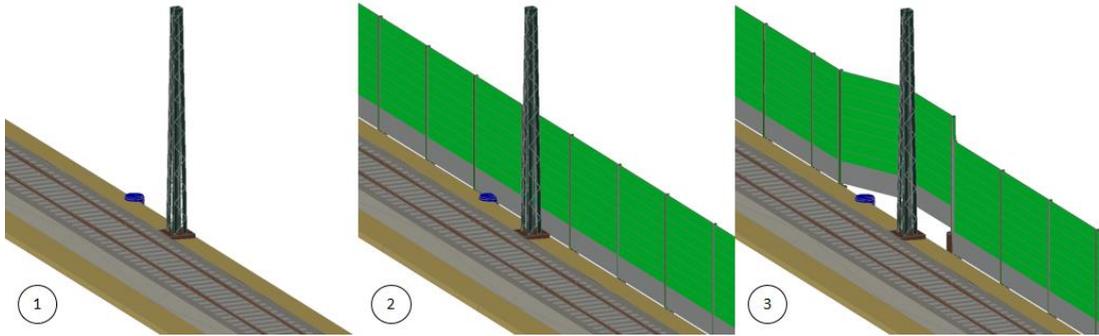


FIG. 14: Railway track model; situation 1 – railway with associated installations; situation 2 – railway with installations and a parallel noise barrier; situation 3 – with installations and noise barrier by-pass.

TABLE 4: DSM for noise barriers

		Output					
		alignment	noise barrier	catenary	signaling	drainage	cable system
Input	alignment		x	x	x	x	x
	noise barrier						
	catenary		x		x	x	x
	signaling		x	x		x	x
	drainage		x				x
	cable system		x			x	

The Design authoring system can work with parametric model data in software-specific formats for alignment as well as for the installation of overhead line and drainage systems. For formalizing the informal model, the approach discussed here is limited to the consideration of catenary masts and manholes.

A methodical description of the procedure was developed in cooperation with the experts as a draft BPMN workflow shown in Fig. 15. The design engineer begins by evaluating the route alignment, checking the constraint points and the distance of the noise barrier from the parallel track lines. In a second step, the engineer checks the structural data of the influencing subsections (here: overhead line and drainage), determines the positions of the overhead line masts and manholes, and plans any necessary by-passes at these points. Since these work steps have to be carried out repeatedly for numerous instances of these objects, the tasks must be conceived as a loop.

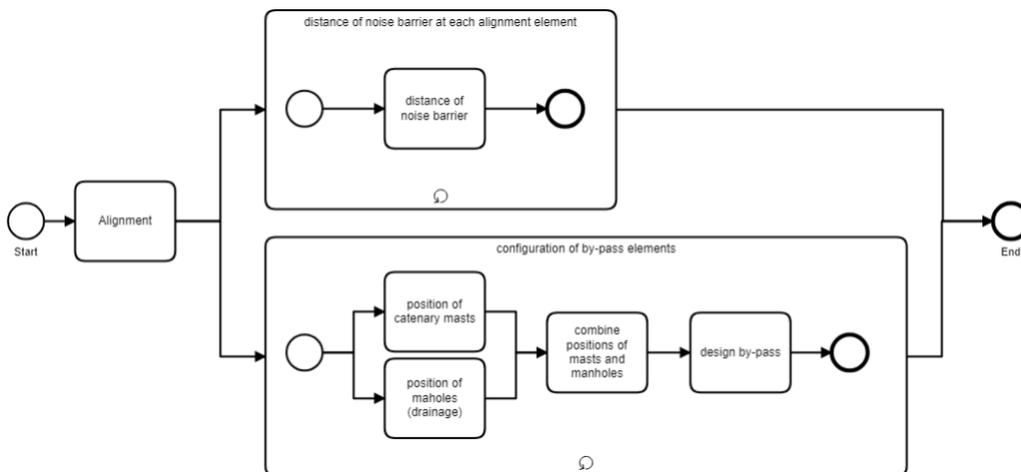


FIG. 15: BPMN workflow of the design process of a noise barrier, created by an expert

This workflow serves as a basis for the Activity forms, of which the “distance of a noise barrier” is shown in Table 5 by way of example. In this study, the combination of standardized forms and the process-related representation using BPMN proved to be helpful as a means of facilitating communication between the participants.

TABLE 5: Example of MOKA activity form, definition of noise barrier distance

Form	Activity	
<b>Name</b>	Definition of noise barrier distance, import data	
<b>Reference</b>	A_D_1	
<b>Trigger</b>	Completion of alignment	
<b>Input</b>	Alignment data	
<b>Output</b>	List of parameters with noise barrier distance at each alignment element	
<b>Potential failure modes</b>	Needed data is incomplete	
<b>Objective</b>	Designing the distance of a noise barrier	
<b>Input requirements</b>	Input data from design authoring system must include information about: station, radius, cant, pace, number of tracks, side of noise barrier	
<b>Context, information, validity</b>		
<b>Description</b>	This activity aims to describe the import data for alignment.	
<b>Related Activities</b>	A_D_2	
<b>Rules involved</b>	R_D_1	
<b>Entities involved</b>	ES_NB_1	
<b>Related Illustrations</b>		
<b>Information Origin</b>	Guideline 800.130 A07, description of design authoring system	
<b>Management</b>	<b>Author</b>	MH
	<b>Date</b>	15.05.2020
	<b>Version Number</b>	1.0
	<b>Status</b>	Final

### 4.3 Formalize

The ICARE forms created above, along with the outline workflow are developed in greater detail in the MOKA “Formalize” step. First, the product model is designed as a UML diagram. Two sub-processes are derived from the workflow, both of which are based on the alignment data of the project, but can also be viewed one after the other, as per the methodical procedure of the experts as well as within the Design authoring system. The sub-processes “Determining the distance” and “Defining the by-passes” are described in the following sections.

#### 4.3.1 Product model

The product model of a noise barrier was developed as a UML diagram, shown in Fig. 16. The UML diagram shows the structural design of a noise barrier and its individual components (as per Fig. 13) and assigns attributes to the classes that need to be considered when designing a noise barrier. To clarify the origin of the data, the attributes are given a color. In this paper, the focus lies on the components relevant for the geometric design and their parameters, which are shown in red.

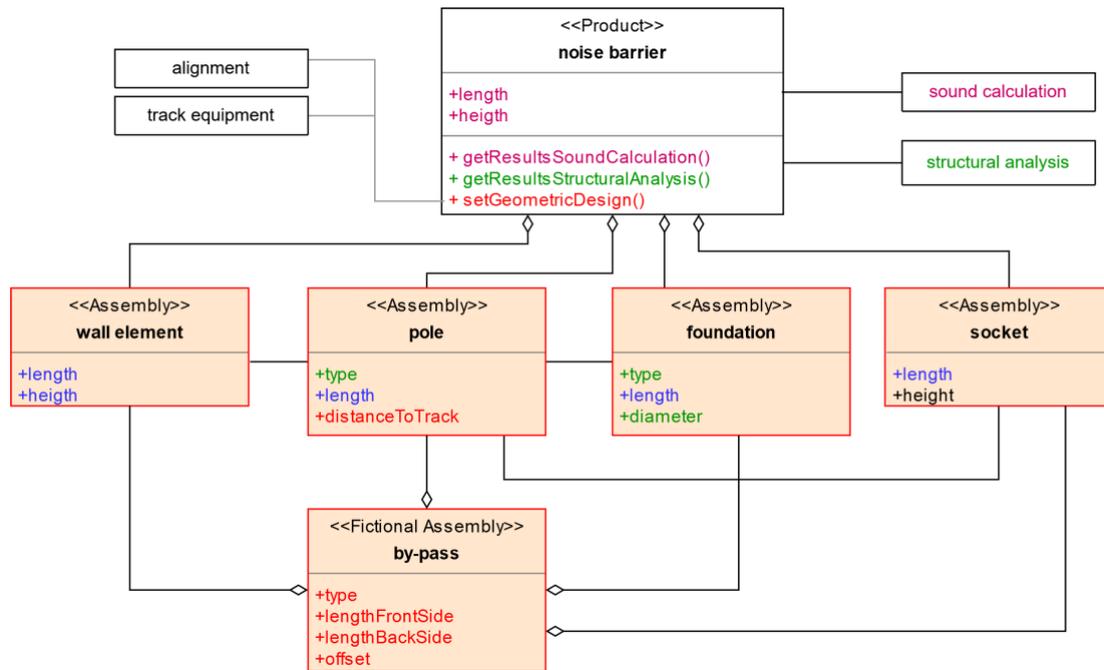


FIG. 16: UML diagram of a noise barrier. Sound calculation (purple) and structural analysis (green) are inputs for the geometric design (red). The blue attributes are calculated by the Design authoring system automatically, and the red attributes by the KBE application.

#### 4.3.2 Design process model: Definition of distance to track

The most important input variable for determining the distance of a noise barrier to the track is the track alignment. The relevant reference object for determining the position is the axis which is comprised of straight line, circular arc and transition curve segments as shown in Fig. 17.

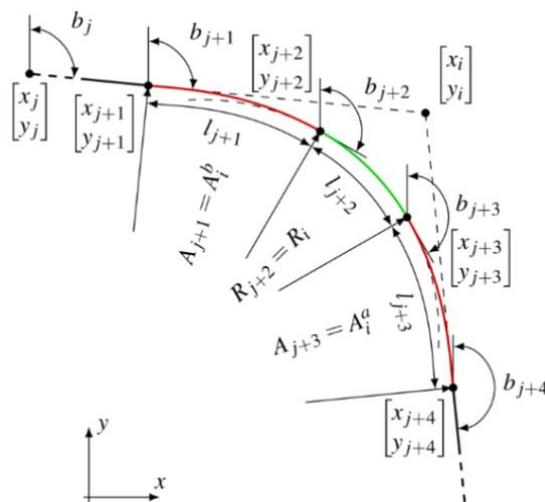


FIG. 17: Different types of segments along a horizontal alignment: straight elements in black, transition curves in red, circular arc in green (after an original in (Markic et al., 2018))

Directive 800.0130 Annex 07 defines the necessary distance between noise barriers and track alignment axes, and this has already been recorded as a Rule form in the informal model. The table contained in the directive is evaluated as a decision table based on its structure and the clear definition of input and output variables, which makes it possible to formalize it as a DMN decision table shown in Table 6.

TABLE 6: Decision table for the distance between a noise barrier and alignment axis

Input				Output
Cant [mm]	Number of tracks [-]	Pace [km/h]	Side of arc [-]	Distance [m]
[0 .. 160]	1	$\leq 160$	-	3.3
[0 .. 160]	1	$> 160$	-	3.8
[0 .. 160]	2	$\leq 160$	inside	3.3
[0 .. 20]	2	$\leq 160$	outside	3.3
[25 .. 50]	2	$\leq 160$	outside	3.4
[55 .. 100]	2	$\leq 160$	outside	3.55
[105 .. 160]	2	$\leq 160$	outside	3.7
[0 .. 160]	2	$> 160$	inside	3.8
[0 .. 20]	2	$> 160$	outside	3.8
[25 .. 50]	2	$> 160$	outside	3.9
[55 .. 100]	2	$> 160$	outside	4.0
[105 .. 160]	2	$> 160$	outside	4.2

This was used as basis for checking the design input data, e.g. to determine if the necessary input variables have been provided. The alignment data includes information on the cant and speed of the axis elements, but the number of tracks can only be specified by the user as there are no other data sources for this. A corresponding input interface was developed as part of the subsequent “Package” step. By using the Activity and Rule forms as well as the decision table, it became clear that a further key variable for automating the execution of the decision table was the position of the noise barrier in relation to the axis, e.g. on which side of the arc the noise barrier lies (left or right of the track). This, too, must be defined by the user. Fig. 18 shows the possible situations. A special case in this context are straight sections of track: these have no curvature and are thus not explicitly detailed in the guidelines. The interviews with experts revealed that these are typically treated as “inside arcs”. As the route alignment data also contains information on the direction of curvature, the following distinction can be made:

- Radius  $< 0$  → left curved
- Radius  $> 0$  → right curved
- Radius = 0 → not curved

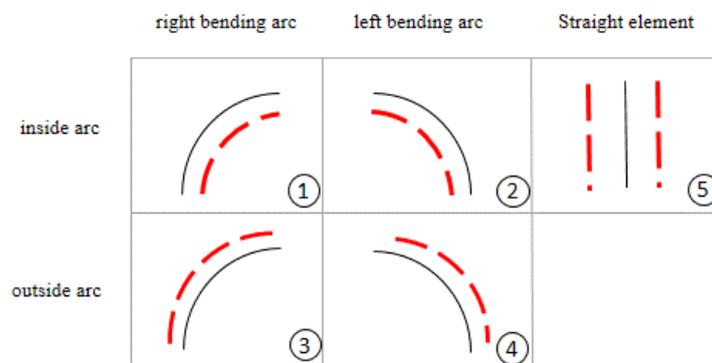


FIG. 18: Relation of noise barrier (red, broken line) to arc element (black, continuous line)

These decision processes can be mapped using BPMN as shown in Fig. 19, where the numbers correspond to the different situations shown in Fig. 18. The DMN task “Define necessary distance” represents the decision table shown in Table 6 as a BPMN node.

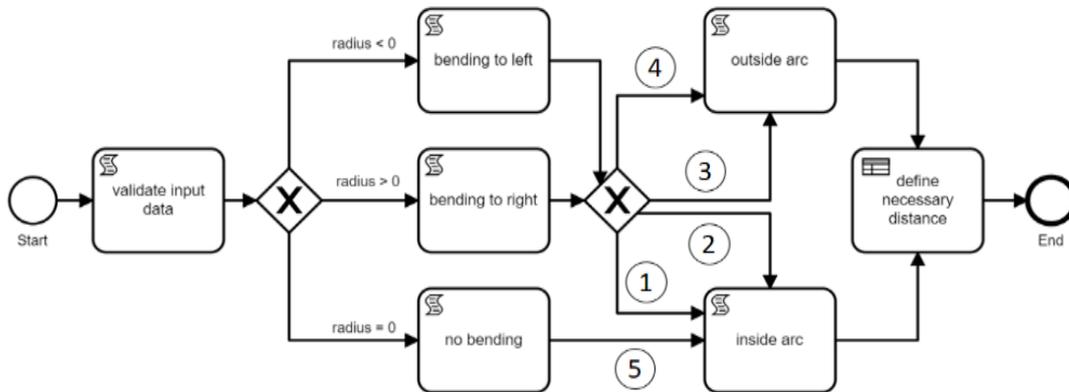


FIG. 19: Workflow for defining the necessary distance at each alignment element

### 4.3.3 Design process model: Definition of by-pass parameters

Alongside determining the distance between the noise barrier and the track alignment axis, the “Capture” phase and the DSM also made it clear that design of noise barriers is subordinate to other subsections such as the design of overhead lines or drainage. These must therefore also serve as input variables for the design of the noise barrier. The next step in the design of noise barriers is therefore to ensure it does not collide with existing installations along the railway line.

The logic of drainage systems means that the data provided by the Design authoring system includes the entire sewer network. Manholes do not necessarily have to collide with the noise barrier. The same also applies to catenary masts for overhead line design. The masts and drainage elements can be positioned on either side of the track and in some cases may not clash with the noise barrier at all. The first step is, therefore, to filter out only those objects that may influence the course of the noise barrier, i.e. those elements located on the same side of the tracks as the noise barrier (see Fig. 20).

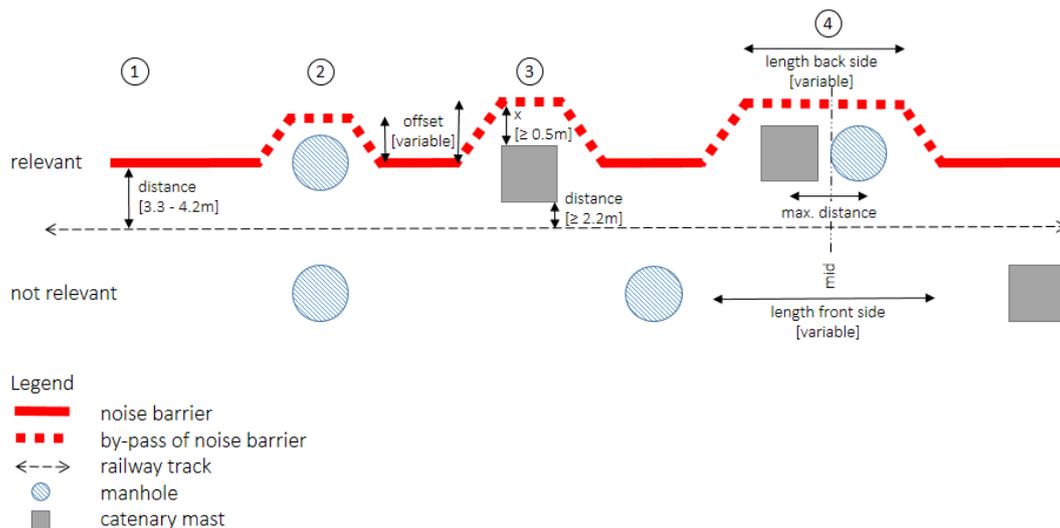


FIG. 20: Sketch of a noise barrier with by-passes. Four situations are shown: 1) without obstacle, 2) collision with a manhole, 3) collision with a mast, 4) by-pass around several obstructions.

The next step is to check which of the filtered objects actually influence the design of the noise barrier and to what extent. The Design authoring system can provide the following parameters on manholes and catenary mast objects:

- Object type
- Station on the track
- Distance of the center of the object to the track
- Dimensions of the object (height, length, width, diameter etc.)

The objects are initially considered independently of each other and it is assumed that each of the objects will result in a single by-pass segment in the noise barrier. The aim of the automation is to determine the length of the respective by-pass around the objects encountered. Catenary masts restrict the space available next to the tracks. The interviews with experts revealed that space must be left for the safe passage of maintenance personnel next to the tracks. The process must therefore determine whether the space between the mast and tracks is sufficient or whether service staff will need to pass behind the mast (see Fig. 20 situation 3). This can be determined with the help of a simple decision table (see Table 7) and requires only the “distance between track and the front edge of the catenary mast” as an input value. If this value is less than 3.3 m, the distance between the rear edge of the mast and the noise barrier must be at least 0.8 m, otherwise 0.5 m (the dimension “x” in Fig. 20 situation 3). In the case of the manhole, this is not relevant for the staff maintenance route as the manhole cover is at ground level and does not obstruct the path passing over it (see Fig. 20 situation 2). Section 4.3.2 describes how the distance of the noise barrier to the track axis is determined (Fig. 20 situation 1). This is in relation to the main elements of the underlying track alignment. In addition to the determined position of the individual relevant objects and the regulatory information on the space to be kept free, the distances from the axis to the front edge of the noise barrier is calculated at each catenary mast or manhole using the same logic as described in Section 4.3.2. With this information, the necessary offset of the noise barrier around manholes and catenary masts can be determined. The corresponding section of the BPMN workflow is shown in Fig. 21.

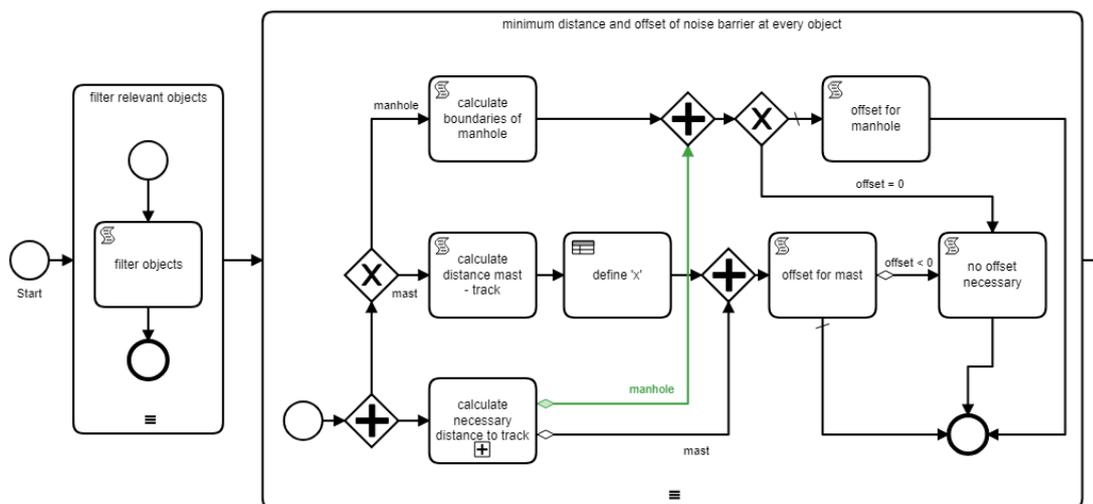


FIG. 21: BPMN diagram for the design of by-passes part I: Filter relevant objects, minimum distance and offset of noise barrier at every object

TABLE 7: Decision table for the definition of ‘x’ as necessary distance between foundation and wall

Input	Output
Distance track to foundation	Distance foundation to wall ‘x’
[m]	[m]
< 3.3	0.8
≥ 3.3	0.5

Using the steps described, the data available from the implicit model was used to generate information and knowledge for the course of the noise barrier. The assumption up to now has been that each manhole or mast will result in an independent bypass. The next step is to determine whether several small by-passes can be combined to form a single larger by-pass (Fig. 20 situation 4), for example because geometrically expedient or for cost reasons. To this end, an additional script task was integrated into the process to identify the spatial proximity of different objects. This script task checks the distance between objects based on a maximum distance specified by the user. By varying this input variable, different variants of the same noise barrier can be generated. The requisite operators for this were combined in a single task to avoid a complex and incomprehensible BPMN diagram. Preidel calls this form of representation the “atomic method” (Preidel, 2020). Once close-by objects – i.e. objects that can be combined in a single by-pass – have been identified, the actual parameters of the by-pass can be determined in an independent sub-process. First, the mid-point of the by-pass is determined from the minimum and maximum station of the objects to be considered. Based on whether closed or open by-passes are to be designed (a user-specified input), additional parameters are determined for “length front” and “length rear”. This enables the user to consider further variants. The corresponding section of the BPMN workflow is shown in Fig. 22.

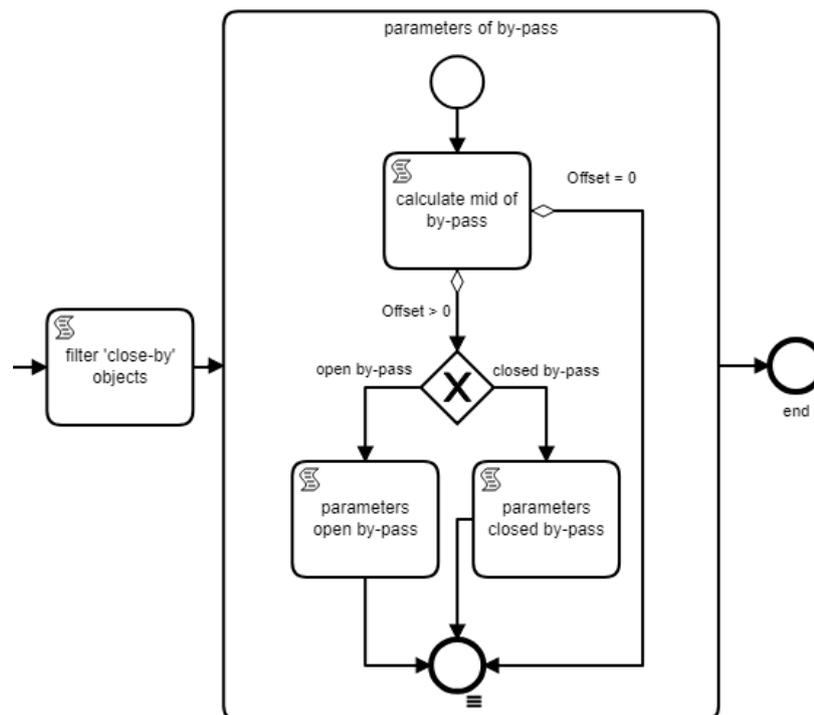


FIG. 22: BPMN diagram for by-pass design Part II: Filter “close-by” objects, parameters of by-pass

The result of this process provides the parameters necessary for the design of by-passes. For this purpose, input data was processed, and new data, information and knowledge was generated. Data required for the Design authoring system can also be passed to it via corresponding interfaces.

Although in MOKA, the “Capture” and “Formalize” phases follow each other, the authors found that questions arising in the course of formalization can make it necessary to repeat individual steps of the capture phase. The authors therefore see these phases as being iterative.

#### 4.4 Package

With the help of BPMN and DMN, the knowledge identified in the “Capture” phase could be digitally mapped in the “Formalize” phase. In this final step, automated systems need to be developed to extract relevant data from the underlying design input data, which is available as tabular data in text files. Since both the Design authoring system and the KBE system have a web interface, the programming language JavaScript can be used for data extraction. As the workflow engine can also interpret JavaScript, the entire KBE application has a consistent programming language. The web-based user interface is shown in Fig. 23.

The user must first select the data sets to be evaluated and then define boundary conditions applicable to the process, such as the number of tracks, and on which side of the tracks the noise barrier is needed, as described earlier. The website allows the user to choose using simple select dropdowns. The BPMN process is also shown.

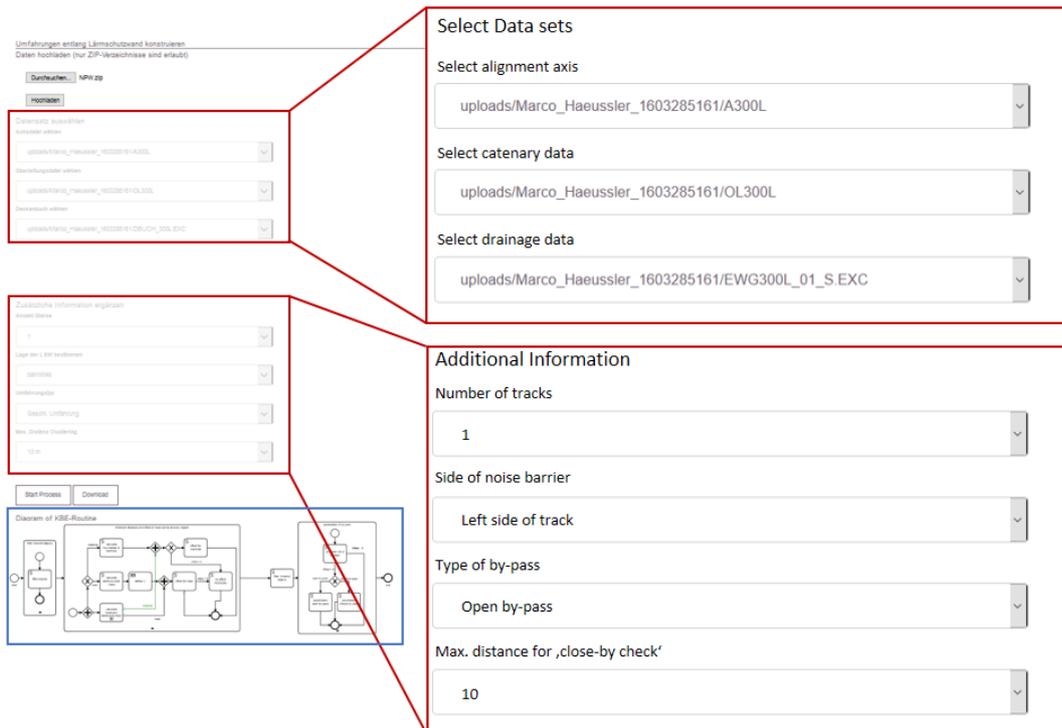


FIG. 23: Web-based user interface for choosing the data to submit to the KBE system (server-side) and setting additional user inputs via dropdowns (red). The BPMN process diagram is also displayed to the user (blue).

In addition, the generated process diagrams are extended in the “Package” step so that they can be executed by the workflow engine used. The variable definitions made need to be checked and adapted if necessary, and the generated data for passing back to the Design authoring system must be converted into the required Comma-separated values format (CSV). For each station (one line per axis element or by-pass) the corresponding parameters are arranged in a pre-defined column sequence. This data can be downloaded from the server by the user and imported back into the Design authoring system. Due to the lack of an API, it is not possible to connect the Design authoring system directly to the KBE application.

Over the course of developing the executable KBE application, one can also see that the “Formalize” and “Package” phases are also closely related and influence each other. This is primarily because using BPMN and DMN makes it possible to produce a consistent logic with an executable language so that the conditions for successful execution can already be determined during the “Formalize” phase. The interaction between the different phases is a positive characteristic in the authors’ view. Fig. 24 shows the interrelationships of the “Capture”, “Formalize” and “Package” phases of the MOKA method in the case study project.

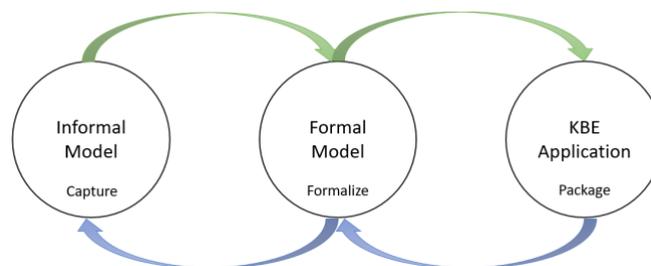


FIG. 24: Relationships between the MOKA phases in the case study

## 4.5 Results

In this study, the aim was to use a uniform presentation form for the activity diagrams from the “Capture” to “Package” phases. As BPMN workflows in combination with DMN decision tables can be made executable by means of a workflow engine, this approach could be used for these phases of this study. The acceptance criteria outlined for this study in Section 3.2 were evaluated for the method used. One of the most important challenges in connection with KBE is that the automation of routine tasks should reduce the processing time and, in turn, production costs. To quantify these results, the processing times for the semi-automated (by using functionalities of the Design authoring system) design of a noise barrier were compared with the KBE-based approach. To determine the time required for semi-automated design, we assumed the user would use the same Design authoring system used in the KBE-based approach. The initial steps for generating input data are the same in both approaches and were therefore not compared. The following data was used as a basis for the design:

- Route axis with a length of 2,700 m
- Sewer system with 102 manholes
- Catenary data with 21 catenary masts

The noise barrier should extend the length of the railway track, i.e. also have a length of 2,700 m.

Table 8 compares the times required to design a noise barrier semi-automated and with the help of the automated KBE approach. The processing steps compared correspond to the procedure described above and the semi-automated processing time are empirical values estimated by five domain experts. The experts came up with close-by estimates independently of each other. According to their estimates, the semi-automated design of the noise barrier takes about 145 min on average. These times are based on routine tasks only; special situations such as designing noise barrier that pass over a bridge structure, are not considered here. By comparison, the KBE-based approach is significantly faster, taking only 3 minutes 40 seconds, which corresponds to a ratio of 2.5 % of the time.

TABLE 8: Comparison of the semi-automated and automated design of a noise barrier

Phase	Description	Time needed for semi-automated design	Time needed for automated design	Ratio
1	Analysis of underlying data	17 min	Included in 2-4	-
2	Import to KBE system	-	1 min	-
3	Definition of noise barrier distance	20 min	20 sec	1.5%
4	Design of by-passes	108 min	2 min	1.9%
5	Export to Design authoring system	-	20 sec	-
	<b>Total</b>	<b>145 min</b>	<b>3 min 40 sec</b>	<b>2.5%</b>

Variants are correspondingly quicker to produce: the user can, for example, decide at the beginning of the process whether the noise barrier should be designed with open or closed by-passes. With the KBE-based approach, the “type of by-pass” selection can be adjusted accordingly, and the system determines the configuration within 3 minutes 20 seconds. A design engineer would require a further 40 minutes to adapt the previously created design. This corresponds to an approx. 92 % improvement in efficiency when developing variants or adjustments to the design.

The entire process of designing the executable KBE application from the “Capture” to the “Package” phase took about 4 weeks in the described example.

## 5. CONCLUSION

This paper has presented a study for a KBE system based on the graphical notation BPMN in combination with DMN. The development of the application followed the general principles of MOKA which is already well established in practice. MOKA aims to collect knowledge in different ways in successive phases and to transfer it from an initially unstructured form to a KBE system. A disadvantage of MOKA in the opinion of the authors is that there is no uniform notation. The results of each individual phase always need to be converted into a different data format for the next, i.e. from ICARE forms to UML diagrams and from UML diagrams to a KBE language.

The aim of this study was to use a consistently uniform presentation form for the activity diagrams. To this end, BPMN and DMN were used as two notations that can be used for the graphical representation of knowledge and are also executable with the help of an associated workflow and decision engine. The case study examined an application in the field of railway infrastructure design for new buildings. As part of the “Justify” phase, 31 acceptance criteria were compiled from the literature and the approach presented here was evaluated against them: the results show that the approach fulfils 58 % (unweighted) and 73 % (weighted) of the criteria. During the “Capture” phase, the available sources of knowledge (expert knowledge, guidelines, functionality of the applications used) were collected and structured with the help of ICARE forms, DSM and BPMN diagrams. In the next step, the informal model was formalized using methods described in earlier work by Häußler *et al.* to formalize the knowledge contained in guidelines with the help of BPMN and DMN. The methodical knowledge of the experts and the requirements of the specialized applications can likewise be formalized with the help of BPMN and DMN.

The case study examined the automation of the geometric design of a new noise barrier along railway tracks and demonstrates that BPMN and DMN can be applied consistently from the “Capture” phase to the actual KBE application. By automating the processing steps, the time required for the design of noise barriers under the given boundary conditions can be reduced many times over: in the case study, the KBE approach requires only 2.5 % of the time required for semi-automated design. The creation of different variants is also significantly faster when using automated processes. In the example shown, 8 % of the semi-automated processing time was required to develop an additional variant of an existing noise barrier design.

In addition to significantly shortening the required generation time, the approach developed here also demonstrates the graphical representation of different types of knowledge in connection with the process-based approach to the design of railway infrastructure facilities. The process is transparent, meaning that the KBE system is not a “black box” solution, which can in turn benefit user acceptance. In addition, the system can be adapted to meet different requirements.

Apart from these advantages, however, there are also some aspects that the approach presented here cannot solve, or only insufficiently address. The KBE application is developed as a system decoupled from the Design authoring system. This makes it necessary to develop appropriate interfaces for importing and exporting data, which may need to be updated as the Design authoring system is developed. Currently, the data must be actively transferred to the KBE solution by the user, which introduces a potential point of error. With the help of an API, the KBE system could in future be directly linked to the Design authoring system. This would also eliminate a further disadvantage: the user would not have to leave the Design authoring system and the result would be displayed in the Design authoring system without further action. In the current prototype, the approach is limited to the knowledge provided by the engineer and the routines devised by the developer. The system is not able to learn independently from the tasks set and find new solutions to problems. For highly standardized structures – such as the example of the noise barrier – this approach is, in the authors’ opinion, quite sufficient. But if more complex, highly specific solutions need to be developed, e.g. escape routes and emergency services access in dense inner-city environments, or the fitting of railway platforms, this approach may currently be too limited.

The case study presented here is limited to the geometric design of new noise barriers. Future investigations should aim to develop cross-domain solutions. Using the noise barrier as an example, this would entail linking together sound calculations, structural analysis and geometric design. This could be achieved, for example, by triggering sophisticated calculation and simulation applications from a workflow or by modelling the calculations and decision paths using BPMN and DMN.

As the field of railway infrastructure design is particularly well defined by standards and regulations, the approach shown here offers very good and valuable support for routine tasks. Further fields of application could be the automated design of superstructures, underground cable routing, platforms and engineering structures such as bridges and tunnels. Applications are also conceivable in road construction, e.g. in the design of road cross sections (road superstructure, road widths). BPMN and DMN methods could also be used for the design of road junctions, though here stronger interaction with Design authoring system would be necessary.

## REFERENCES

- AKG Software Consulting GmbH (2021), vestra civil, Retrieved from <https://www.akgsoftware.com/solutions/railway/>
- Alreshidi E., Mourshed M. and Rezgui Y. (2016), Cloud-Based BIM Governance Platform Requirements and Specifications: Software Engineering Approach Using BPMN and UML, *Journal of Computing in Civil Engineering*, Vol. 30, 1–41, Retrieved from <https://ascelibrary.org/doi/full/10.1061/%28ASCE%29CP.1943-5487.0000539>
- Álvarez B.J., Cuesta E., Martínez S., Barreiro J. and Fernández P. (2010), Implementation of decision rules for CMM sampling in a KBE system, *Proceedings of the 36th International MATADOR Conference*, Vol. 2010-Janua, 335–338, doi: [https://doi.org/10.1007/978-1-84996-432-6\\_76](https://doi.org/10.1007/978-1-84996-432-6_76)
- Amadori K., Tarkian M., Ölvander J. and Krus P. (2012), Flexible and robust CAD models for design automation, *Advanced Engineering Informatics*, Vol. 26, 180–195, doi: 10.1016/j.aei.2012.01.004
- Ammar-Khodja S., Perry N. and Bernard A. (2008), Processing Knowledge to Support Knowledge-based Engineering Systems Specification, *Concurrent Engineering Research and Applications*, Vol. 16, 89–101, doi: 10.1177/1063293x07084642
- An S., Martinez P., Al-Hussein M. and Ahmad R. (2020), BIM-based decision support system for automated manufacturability check of wood frame assemblies, *Automation in Construction*, Vol. 111, 1–15, doi: 10.1016/j.autcon.2019.103065
- Aram S.V. and Eastman C. (2010), Introducing a new methodology to develop the Information Delivery Manual for AEC projects, *Proceedings of the CIB W78 2010: 27th International Conference –Cairo, Egypt*, 1–10, Retrieved from <https://itc.scix.net/pdfs/w78-2010-49.pdf>
- Autodesk Inc. (2021), autodesk civil 3d, Retrieved from <https://www.autodesk.eu/products/civil-3d/rail-design?plc=CIV3D&term=1-YEAR&support=ADVANCED&quantity=1>
- Barreiro J., Martinez S., Cuesta E., Alvarez B. and Fernandez P. (2009), high level diagrams for identification of knowledge as a basis for a kbe implementation of inspection planning process, *III Manufacturing Engineering Society International Congress*, Retrieved from [https://www.researchgate.net/profile/J\\_Barreiro2/publication/235792948\\_High\\_level\\_diagrams\\_for\\_identification\\_of\\_knowledge\\_as\\_a\\_basis\\_for\\_a\\_KBE\\_implementation\\_of\\_inspection\\_planning\\_process/links/0912f5138df4477bb500000/High-level-diagrams-for-identifica](https://www.researchgate.net/profile/J_Barreiro2/publication/235792948_High_level_diagrams_for_identification_of_knowledge_as_a_basis_for_a_KBE_implementation_of_inspection_planning_process/links/0912f5138df4477bb500000/High-level-diagrams-for-identifica)
- Beier M., Grütz H., Jäger K., Kock G., Onnich J. and Strube R. (n.d.), headed for the low-noise railway: the db noise reduction research programme, Retrieved from <http://www.railway-research.org/IMG/pdf/022.pdf>
- Bellinger G., Castro D. and Mills A. (2004), Data, Information, Knowledge, and Wisdom, 1–3, Retrieved from <https://www.systems-thinking.org/dikw/dikw.htm>
- Bentley Systems Inc. (2021), bentley openrail designer, Retrieved from <https://www.bentley.com/en/products/product-line/civil-design-software/openrail-designer>
- Bermell-García P. and Fan I.-S. (2002), A KBE System for the design of wind tunnel models using reusable knowledge components, *Ponencia presentada en el VI Congreso Internacional de Ingeniería de Proyectos celebrada en Barcelona en el año 2002*, 594–604, Retrieved from <https://www.semanticscholar.org/paper/A-KBE-System-for-the-design-of-wind-tunnel-models-García-Ip-Shing/17198e5244e421981afe16834cc2ea868a508a12>
- Bermell-García P., Verhagen W.J.C., Astwood S., Krishnamurthy K., Johnson J.L., Ruiz D., Scott G. and Curran R. (2012), A framework for management of Knowledge-Based Engineering applications as software services: Enabling personalization and codification, *Advanced Engineering Informatics*, Vol. 26, 219–230,

doi: 10.1016/j.aei.2012.01.006

- Bhaskara S. (2010), DSM based approach for managing requirements, rules and design parameters in knowledge based design process, *Managing Complexity by Modelling Dependencies - Proceedings of the 12th International DSM Conference*, 237–248, Retrieved from <https://www.designsociety.org/publication/30370/DSM+Based+Approach+for+Managing+Requirements+Rules+and+Design+Parameters+in+Knowledge+Based+Design+Process>
- Bolisani E. and Bratianu C. (2018), The Elusive Definition of Knowledge, in: *Bolisani, E., Bratianu, C. (Eds.), Emergent Knowledge Strategies: Strategic Thinking in Knowledge Management*. Springer International Publishing, 1–22, doi: 10.1007/978-3-319-60657-6\_1
- Borrmann A., Kolbe T.H., Donaubaer A., Steuer H., Jubierre J.R. and Flurl M. (2014), Multi-Scale Geometric-Semantic Modeling of Shield Tunnels for GIS and BIM Applications, *Computer-Aided Civil and Infrastructure Engineering*, Vol. 30, 263–281, doi: 10.1111/mice.12090
- Borrmann A., König M., Koch C. and Beetz J. (2017a), 4.3 - Process modeling, in: *Building Information Modeling - Technology Foundation and Industry Practice*. 1–597
- Borrmann A., König M., Koch C. and Beetz J. (2017b), 2.2 - Solid modeling, in: *Building Information Modeling - Technology Foundation and Industry Practice*. 1–597
- Brimble R. and Sellini F. (2000), The MOKA Modelling Language, in: *Knowledge Engineering and Knowledge Management Methods, Models, and Tools*. 49–56
- Brown N.C., Jusiega V. and Mueller C.T. (2020), Implementing data-driven parametric building design with a flexible toolbox, *Automation in Construction*, Vol. 118, 103252, doi: 10.1016/j.autcon.2020.103252
- Calkins D.E., Egging N. and Scholz C. (2000), Knowledge-Based Engineering (KBE) Design Methodology at the Undergraduate and Graduate Levels, *International Journal of Engineering Education*, Vol. 16, 21–38, Retrieved from <https://www.ijee.ie/articles/Vol16-1/Ijee1118.pdf>
- Camba J.D., Contero M. and Company P. (2016), Parametric CAD modeling: An analysis of strategies for design reusability, *CAD Computer Aided Design*, Vol. 74, 18–31, doi: 10.1016/j.cad.2016.01.003
- Camunda Services GmbH (2018), camunda, Retrieved from <https://camunda.com/>
- Cavieres A., Gentry R. and Al-Haddad T. (2011), Knowledge-based parametric tools for concrete masonry walls: Conceptual design and preliminary structural analysis, *Automation in Construction*, Vol. 20, 716–728, doi: 10.1016/j.autcon.2011.01.003
- CGS Labs (2021), ferrovia, Retrieved from <https://cgs-labs.com/ferrovia/>
- Chan P.K.M. (2013), A New Methodology for the Development of Simulation Workflows: Moving Beyond MOKA, 1–204, Retrieved from <http://resolver.tudelft.nl/uuid:51041c5c-66c2-4d47-bf20-91459a538f7d>
- Chassiakos A.P., Vagiotas P. and Theodorakopoulos D.D. (2005), A knowledge-based system for maintenance planning of highway concrete bridges, *Advances in Engineering Software*, Vol. 36, 740–749, doi: 10.1016/j.advengsoft.2005.03.020
- Chen Y., Okudan G.E. and Riley D.R. (2010), Decision support for construction method selection in concrete buildings: Prefabrication adoption and optimization, *Automation in Construction*, Vol. 19, 665–675, doi: 10.1016/j.autcon.2010.02.011
- Cheng M.Y., Peng H.S., Huang C.M. and Chen C.H. (2012), KM-oriented business process reengineering for construction firms, *Automation in Construction*, Vol. 21, 32–45, doi: 10.1016/j.autcon.2011.05.010
- Chiciudean T.G., La Rocca G. and van Tooren M.J.L. (2008), a knowledge based engineering approach to support



- automatic design of wind turbine blades, *CIRP Design Conference 2008*, Retrieved from <http://resolver.tudelft.nl/uuid:de72e000-1c1e-4dbe-83ec-a1aef880d60e>
- Corallo A., Laubacher R., Margherita A. and Turrisi G. (2009), Enhancing product development through knowledge-based engineering (KBE)-A case study in the aerospace industry, *Journal of Manufacturing Technology Management*, Vol. 20, 1070–1083, doi: 10.1108/17410380910997218
- DB Netz AG (2018), richtlinie 800.0130 - streckenquerschnitte auf erdkörpern, (*Guideline 800.0130 - Cross sections at earthwork*), Retrieved from <https://mediendienste.extranet.deutschebahn.com/DIBS/>
- Deutsche Bahn AG (2020), noise barriers - deutsche bahn, Retrieved from <https://gruen.deutschebahn.com/en/measures/noise-barriers>
- Dibley M.J., Li H., Miles J.C. and Rezgui Y. (2011), Towards intelligent agent based software for building related decision support, *Advanced Engineering Informatics*, Vol. 25, 311–329, doi: 10.1016/j.aei.2010.11.002
- Dimyadi J. and Amor R. (2017), automating conventional compliance audit processes, *J. Ríos et al (eds) Product Lifecycle Management and the Industry of the Future, Vol 517. Proceedings of the 14th IFIP WG 5.1 International Conference (PLM 2017), Seville, Spain, 10-12 July 2017*, Retrieved from [https://www.researchgate.net/profile/Johannes\\_Dimyadi/publication/321691957\\_Automating\\_Conventional\\_Compliance\\_Audit\\_Processes/links/5b263743aca272277fb5d260/Automating-Conventional-Compliance-Audit-Processes.pdf](https://www.researchgate.net/profile/Johannes_Dimyadi/publication/321691957_Automating_Conventional_Compliance_Audit_Processes/links/5b263743aca272277fb5d260/Automating-Conventional-Compliance-Audit-Processes.pdf)
- Dimyadi J., Clifton C., Spearpoint M. and Amor R. (2016), Computerizing Regulatory Knowledge for Building Engineering Design, *Journal of Computing in Civil Engineering*, Vol. 30, 1–13, doi: 10.1061/(asce)cp.1943-5487.0000572
- Dimyadi J., Clifton C., Spearpoint M. and Amor R. (2014), Regulatory Knowledge Encoding Guidelines for Automated Compliance Audit of Building Engineering Design, in: 2014 International Conference on Computing in Civil and Building Engineering. 1–8, doi: 10.1061/9780784413616.067
- Egemen M. and Mohamed A. (2008), SCBMD: A knowledge-based system software for strategically correct bid/no bid and mark-up size decisions, *Automation in Construction*, Vol. 17, 864–872, doi: 10.1016/j.autcon.2008.02.013
- Emberey C.L., Milton N.R., Berends J.P.T.J., Van Tooren M.J.L., Van Der Elst S.W.G. and Vermeulen B. (2007), Application of Knowledge Engineering Methodologies to Support Engineering Design Application Development in Aerospace, *7th AIAA Aviation Technology, Integration, and Operations Conference*, Vol. 1, 83–95, doi: 10.2514/6.2007-7708
- Ertas A. (2018), *Transdisciplinary Engineering Design Process*, John Wiley & Sons, Inc., Retrieved from <https://learning.oreilly.com/library/view/transdisciplinary-engineering-design/9781119474753/?ar>
- Flurl M., Singer D., Mundani R.-P., Rank E. and Borrmann A. (2015), A Rule-based Collaborative Modelling System for Infrastructure Design, 1–10, Retrieved from [https://publications.cms.bgu.tum.de/2015\\_Flurl\\_Collaboration.pdf](https://publications.cms.bgu.tum.de/2015_Flurl_Collaboration.pdf)
- Girodon J., Monticolo D., Bonjour E. and Perrier M. (2015), An organizational approach to designing an intelligent knowledge-based system: Application to the decision-making process in design projects, *Advanced Engineering Informatics*, Vol. 29, 696–713, doi: 10.1016/j.aei.2015.07.001
- Haller H. (2011), *User Interfaces for Personal Knowledge Management with Semantic Technologies*, 1–203, Retrieved from <http://www.heikohaller.de/literatur>
- Häußler M. and Borrmann A. (2020), Model-based quality assurance in railway infrastructure planning, *Automation in Construction*, Vol. 109, 1–15, doi: <https://doi.org/10.1016/j.autcon.2019.102971>

- Häußler M., Esser S. and Borrmann A. (2021), Code compliance checking of railway designs by integrating BIM, BPMN and DMN, *Automation in Construction*, Vol. 121, 1–24, doi: 10.1016/j.autcon.2020.103427
- Helguson M. and Kalhori V. (2012), A Conceptual Model for Knowledge Integration in Process Planning, *Procedia CIRP*, Vol. 3, 573–578, doi: 10.1016/j.procir.2012.07.098
- Hjelseth E. and Nisbet N. (2011), capturing normative constraints by use of the semantic mark-up rase methodology, *CIB W78-W102 2011: International Conference*, Retrieved from <https://itc.scix.net/pdfs/w78-2011-Paper-45.pdf>
- Hunter R., Rios J., Perez J.M. and Vizan A. (2006), A functional approach for the formalization of the fixture design process, *International Journal of Machine Tools and Manufacture*, Vol. 46, 683–697, doi: 10.1016/j.ijmachtools.2005.04.018
- IB&T Software GmbH (2021), card\_1, Retrieved from <https://www.card-1.com/en/cross-section-modules/>
- İlal S.M. and Günaydın H.M. (2017), Computer representation of building codes for automated compliance checking, *Automation in Construction*, Vol. 82, 43–58, doi: 10.1016/j.autcon.2017.06.018
- ISO/IEC 19510:2013 (2013), Information technology — Object Management Group Business Process Model and Notation, Vol. 2013, 1–534, Retrieved from <https://www.omg.org/spec/BPMN/ISO/19510/PDF>
- ISO 29481-1:2016 (2016), Building information models — Information delivery manual — Part 1: Methodology and format, 1–40
- ISO 30401:2018 (2018), Knowledge management systems - Requirements, Vol. 11, 1–28
- Jaud S., Esser S., Muhić S. and Borrmann A. (2020), Development of IFC schema for infrastructure, *Proceedings of the 6th International Conference siBIM*, 1–9, Retrieved from [https://publications.cms.bgu.tum.de/2020\\_Jaud\\_siBIM.pdf](https://publications.cms.bgu.tum.de/2020_Jaud_siBIM.pdf)
- Johansson J., Contero M., Company P. and Elgh F. (2018), Supporting connectivism in knowledge based engineering with graph theory, filtering techniques and model quality assurance, *Advanced Engineering Informatics*, Vol. 38, 252–263, doi: 10.1016/j.aei.2018.07.005
- Kim T.W., Rajagopal R., Fischer M. and Kam C. (2013), A knowledge-based framework for automated space-use analysis, *Automation in Construction*, Vol. 32, 165–176, doi: 10.1016/j.autcon.2012.08.002
- König M., Dirnbek J. and Stankovski V. (2013), Architecture of an open knowledge base for sustainable buildings based on Linked Data technologies, *Automation in Construction*, Vol. 35, 542–550, doi: 10.1016/j.autcon.2013.07.002
- La Rocca G. (2012), Knowledge based engineering: Between AI and CAD. Review of a language based technology to support engineering design, *Advanced Engineering Informatics*, Vol. 26, 159–179, doi: 10.1016/j.aei.2012.02.002
- Liew A. (2013), DIKIW: Data, information, knowledge, intelligence, wisdom and their interrelationships, *Business Management Dynamics*, Vol. 2, 49–62, Retrieved from [https://www.researchgate.net/publication/236870996\\_DIKIW\\_Data\\_Information\\_Knowledge\\_Intelligence\\_Wisdom\\_and\\_their\\_Interrelationships](https://www.researchgate.net/publication/236870996_DIKIW_Data_Information_Knowledge_Intelligence_Wisdom_and_their_Interrelationships)
- Liu F., Jallow A.K., Anumba C.J. and Wu D. (2013), building knowledge modeling: integrating knowledge in bim, *Cib W78*, Retrieved from <https://www.semanticscholar.org/paper/BUILDING-KNOWLEDGE-MODELING%3A-INTEGRATING-KNOWLEDGE-Liu-Jallow/26e436be170558ac75112afe1edd4ce2352bb5b0>
- Lohith M., Prasanna L. and Vaderahobli D.H. (2013), Translating MOKA Based Knowledge Models into a

- Generative CAD Model in CATIA V5 Using Knowledgeware, *International Conference on Modeling, Simulation and Visualization Methods(MSV). The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp)*, 1–6, Retrieved from <https://www.semanticscholar.org/paper/Translating-MOKA-based-Knowledge-models-into-a-CAD-KnowledgewareLohith-Prasanna/58f5d9ba8829603a0e817d4e83198cf1a14ab2b1>
- Markic S., Schlenger J. and Bratoev I. (2018), tangible alignment design, *30. Forum Bauinformatik, Weimar, Germany*, Retrieved from [https://publications.cms.bgu.tum.de/2018\\_Markic\\_FBI.pdf](https://publications.cms.bgu.tum.de/2018_Markic_FBI.pdf)
- Martínez S., Barreiro J., Cuesta E., Álvarez B.J. and González D. (2012), Methodology for identifying and representing knowledge in the scope of CMM inspection resource selection, *AIP Conference Proceedings*, Vol. 1431, 250–257, doi: 10.1063/1.4707572
- Montali J., Overend M., Pelken P.M. and Sauchelli M. (2017), Knowledge-based engineering applications for supporting the design of precast concrete facade panels, in: *Proceedings of the International Conference on Engineering Design, ICED*. 131–139
- Montali J., Sauchelli M., Jin Q. and Overend M. (2019), Knowledge-rich optimisation of prefabricated façades to support conceptual design, *Automation in Construction*, Vol. 97, 192–204, doi: 10.1016/j.autcon.2018.11.002
- Motamedi A., Hammad A. and Asen Y. (2014), Knowledge-assisted BIM-based visual analytics for failure root cause detection in facilities management, *Automation in Construction*, Vol. 43, 73–83, doi: 10.1016/j.autcon.2014.03.012
- Motawa I. and Almarshad A. (2013), A knowledge-based BIM system for building maintenance, *Automation in Construction*, Vol. 29, 173–182, doi: 10.1016/j.autcon.2012.09.008
- Obergriesser M. and Borrmann A. (2012), Infrastructural BIM Standards - Development of an Information Delivery Manual for the geotechnical infrastructural design and analysis process, 1–7, Retrieved from [https://www.researchgate.net/publication/290353617\\_Infrastructural\\_BIM\\_standards\\_-\\_Development\\_of\\_an\\_Information\\_Delivery\\_Manual\\_for\\_the\\_geotechnical\\_infrastructural\\_design\\_and\\_analysis\\_process](https://www.researchgate.net/publication/290353617_Infrastructural_BIM_standards_-_Development_of_an_Information_Delivery_Manual_for_the_geotechnical_infrastructural_design_and_analysis_process)
- Object Management Group (2019), Decision Model and Notation Version 1.3, 1–261, Retrieved from <https://www.omg.org/spec/DMN/1.3/PDF>
- Park Y.H., Cho C.Y. and Lee G. (2011), Identifying a subset of BPMN for IDM development, *Proceedings of the CIB W78-W102, International Conference, Sophia Antipolis, France, 26-28 October*, 1–6, Retrieved from <https://www.semanticscholar.org/paper/IDENTIFYING-A-SUBSET-OF-BPMN-FOR-IDM-DEVELOPMENT-Lee/c7e6ec2a662007ac80601475d6917a776f27ae62>
- Preidel C. (2020), Automatisierte Konformitätsprüfung digitaler Bauwerksmodelle hinsichtlich geltender Normen und Richtlinien mit Hilfe einer visuellen Programmiersprache (Automated code compliance checking of digital building models with regard to applicable standards and, 1–266, Retrieved from <https://mediatum.ub.tum.de/doc/1534486/1534486.pdf>
- Preidel C. and Borrmann A. (2015), Automated Code Compliance Checking Based on a Visual Language and Building Information Modeling, *Proceedings of the 32nd International Symposium of Automation and Robotics in Construction, 15-18 June*, 256–263, doi: 10.13140/rg.2.1.1542.2805
- Premkumar V., Krishnamurthy S., Wileden J.C. and Grosse I.R. (2014), A semantic knowledge management system for laminated composites, *Advanced Engineering Informatics*, Vol. 28, 91–101, doi: 10.1016/j.aei.2013.12.004
- ProVI GmbH (2020), provi 6.2, Retrieved from <https://www.provi-cad.de>

- Recker J. (2010), Opportunities and constraints : The current struggle with BPMN, *Business Process Management Journal*, Vol. 16, 181–201, doi: 10.1108/14637151011018001
- Recker J., Indulska M., Rosemann M. and Green P. (2006), How good is BPMN really? Insights from Theory and Practice, *14th European Conference on Information Systems*, 1–12, Retrieved from [https://aisel.aisnet.org/ecis2006/135/?utm\\_source=aisel.aisnet.org%2Fecis2006%2F135&utm\\_medium=PDF&utm\\_campaign=PDFCoverPages](https://aisel.aisnet.org/ecis2006/135/?utm_source=aisel.aisnet.org%2Fecis2006%2F135&utm_medium=PDF&utm_campaign=PDFCoverPages)
- Reifenhäuser M., Klebermass R. and Mautz R. (2018), bim ifc-alignment: fachliche anforderungen rail der db, öbb und sbb, (*Technical Requirements Rail of DB, ÖBB, SBB*), Retrieved from <https://mediendienste.extranet.deutschebahn.com/DIBS/>
- Rempling R., Mathern A., Tarazona Ramos D. and Luis Fernández S. (2019), Automatic structural design by a set-based parametric design method, *Automation in Construction*, Vol. 108, 102936, doi: 10.1016/j.autcon.2019.102936
- Rezgui Y., Hopfe C.J. and Vorakulpipat C. (2010), Generations of knowledge management in the architecture, engineering and construction industry: An evolutionary perspective, *Advanced Engineering Informatics*, Vol. 24, 219–228, doi: 10.1016/j.aei.2009.12.001
- Roth D., Binz H. and Watty R. (2010), Generic structure of knowledge within the product development process, *11th International Design Conference, DESIGN 2010*, 1681–1690, Retrieved from <https://www.designsociety.org/publication/29514/GENERIC+STRUCTURE+OF+KNOWLEDGE+WIT HIN+THE+PRODUCT+DEVELOPMENT+PROCESS>
- Rowley J. (2007), The wisdom hierarchy: Representations of the DIKW hierarchy, *Journal of Information Science*, Vol. 33, 163–180, doi: 10.1177/0165551506070706
- Sainter Phillip, Oldham K. and Larkin A. (2000), Achieving benefits from knowledge-based engineering systems in the longer term as well as in the short term, *Proceedings of the 6th International Conference on Concurrent Enterprising*, 143–150, Retrieved from [https://www.researchgate.net/publication/229044325\\_Achieving\\_benefits\\_from\\_knowledge-based\\_engineering\\_systems\\_in\\_the\\_longer\\_term\\_as\\_well\\_as\\_in\\_the\\_short\\_term](https://www.researchgate.net/publication/229044325_Achieving_benefits_from_knowledge-based_engineering_systems_in_the_longer_term_as_well_as_in_the_short_term)
- Sainter P, Oldham K., Larkin A., Murton A. and Brimble R. (2000), Product Knowledge Management Within Knowledge-Based Engineering Systems, *Proceedings of DETC'00 ASME 2000 Design Engineering Technical Conference And Computers and Information in Engineering Conference*, 1–8
- Sandberg M. (2015), Towards a Knowledge-Based Engineering Methodology for Construction, *ICCREM 2015 - Environment and the Sustainable Building - Proceedings of the 2015 International Conference on Construction and Real Estate Management*, 1–8, doi: 10.1061/9780784479377.001
- Sandberg M. (2003), Knowledge Based Engineering - In Product Development, 1–16, Retrieved from <https://www.diva-portal.org/smash/get/diva2:995005/FULLTEXT01.pdf>
- Sandberg M., Jensen P., Ramic Z. and Simonsson P. (2016), knowledge-based bridge design, *IABSE Congress Stockholm, 2016: Challenges in Design and Construction of an Innovative and Sustainable Built Environment*, Retrieved from <https://www.diva-portal.org/smash/get/diva2:1000812/FULLTEXT01.pdf>
- Schreiber G., Akkermans H., Anjewierden A., de Hoog R., Shadbolt N.R., Van de Velde W. and Wielinga B.J. (2018), Knowledge Engineering and Management: The CommonKADS Methodology, *Knowledge Engineering and Management*, doi: 10.7551/mitpress/4073.001.0001
- Singer D. (2014), Entwicklung eines Prototyps für den Einsatz von Knowledge-based Engineering in frühen Phasen des Brückenentwurfs (Development of a KBE-prototype for early bridge-design stages), *Master's Thesis, TU München*, 1–121, Retrieved from [https://publications.cms.bgu.tum.de/theses/singer\\_2014\\_amann.pdf](https://publications.cms.bgu.tum.de/theses/singer_2014_amann.pdf)

- Singer D. and Borrmann A. (2015), A novel knowledge-based engineering approach for infrastructure design, *Proceedings of the Fourth International Conference on Soft Computing Technology in Civil, Structural and Environmental Engineering*, Civil-Comp Press, Stirlingshire, UK, Vol. 109, 1–16, doi: 10.4203/ccp.109.17
- Singer D., Bügler M. and Borrmann A. (2016), Knowledge based Bridge Engineering - Artificial Intelligence meets Building Information Modeling, *23rd International Workshop of the European Group for Intelligent Computing in Engineering, EG-ICE 2016*, 1–10, Retrieved from [https://publications.cms.bgu.tum.de/2016\\_Singer\\_AIB.pdf](https://publications.cms.bgu.tum.de/2016_Singer_AIB.pdf)
- Singhaputtangkul N., Low S.P., Teo A.L. and Hwang B.G. (2013), Knowledge-based decision support system quality function deployment (KBDSS-QFD) tool for assessment of building envelopes, *Automation in Construction*, Vol. 35, 314–328, doi: 10.1016/j.autcon.2013.05.017
- Skarka W. (2007), Application of MOKA methodology in generative model creation using CATIA, *Engineering Applications of Artificial Intelligence*, Vol. 20, 677–690, doi: 10.1016/j.engappai.2006.11.019
- Skarka W. (2006), Constructing model of knowledge base according to MOKA methodology, *Diagnostyka*, Vol. 2, 37–40, Retrieved from <http://yadda.icm.edu.pl/baztech/element/bwmeta1.element.baztech-article-BAR0-0019-0043>
- Steenhuizen D. and Van Tooren M. (2012), The implementation of a knowledge-based framework for the aerodynamic optimization of a morphing wing device, *Advanced Engineering Informatics*, Vol. 26, 207–218, doi: 10.1016/j.aei.2012.02.004
- Stokes M. (2001), *Managing Engineering Knowledge; MOKA: Methodology for Knowledge Based Engineering Applications*, Professional Engineering Publishing Limited
- Tang D., Zhu R., Tang J., Xu R. and He R. (2010), Product design knowledge management based on design structure matrix, *Advanced Engineering Informatics*, Vol. 24, 159–166, doi: 10.1016/j.aei.2009.08.005
- Tang F., Ma T., Guan Y. and Zhang Z. (2020), Parametric modeling and structure verification of asphalt pavement based on BIM-ABAQUS, *Automation in Construction*, Vol. 111, 103066, doi: 10.1016/j.autcon.2019.103066
- Tripathi K.P. (2011), a review on knowledge-based expert system: concept and architecture, *Artificial Intelligence Techniques - Novel Approaches & Practical Applications*, Retrieved from <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.206.3352&rep=rep1&type=pdf>
- UML Activity Diagrams (2020), UML Activity Diagrams, Retrieved from <https://www.uml-diagrams.org/activity-diagrams.html>
- Van Der Velden C., Bil C. and Xu X. (2012), Adaptable methodology for automation application development, *Advanced Engineering Informatics*, Vol. 26, 231–250, doi: 10.1016/j.aei.2012.02.007
- Verein Deutscher Ingenieure e.V. (2017), Wissensmanagement im Ingenieurwesen - Wissensbasierte Konstruktion (Knowledge management for engineering - Knowledge-based engineering), *VDI 5610 (2)*, 1–67, Retrieved from <https://www.vdi.de/richtlinien/details/vdi-5610-blatt-2-wissensmanagement-im-ingenieurwesen-wissensbasierte-konstruktion-kbe>
- Verhagen W.J.C., Bermell-Garcia P., Van Dijk R.E.C. and Curran R. (2012), A critical review of Knowledge-Based Engineering: An identification of research challenges, *Advanced Engineering Informatics*, Vol. 26, 5–15, doi: 10.1016/j.aei.2011.06.004
- Vilgertshofer S. and Borrmann A. (2017), Using graph rewriting methods for the semi-automatic generation of parametric infrastructure models, *Advanced Engineering Informatics*, Vol. 33, 502–515, doi: 10.1016/j.aei.2017.07.003

- Wang L. and Leite F. (2016), Formalized knowledge representation for spatial conflict coordination of mechanical, electrical and plumbing (MEP) systems in new building projects, *Automation in Construction*, Vol. 64, 20–26, doi: 10.1016/j.autcon.2015.12.020
- Wang M., Deng Y., Won J. and Cheng J.C.P. (2019), An integrated underground utility management and decision support based on BIM and GIS, *Automation in Construction*, Vol. 107, 1–22, doi: 10.1016/j.autcon.2019.102931
- Weise M., Liebich T. and Wix J. (2008), Integrating use case definitions for IFC developments, 1–9, Retrieved from <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.467.9870&rep=rep1&type=pdf>
- Wu J.W., Tseng J.C.R., Yu W. Der, Yang J. Bin, Lee S.M. and Tsai W.N. (2012), An integrated proactive knowledge management model for enhancing engineering services, *Automation in Construction*, Vol. 24, 81–88, doi: 10.1016/j.autcon.2012.02.006
- Yildiz A.E., Dikmen I., Birgonul M.T., Ercoskun K. and Alten S. (2014), A knowledge-based risk mapping tool for cost estimation of international construction projects, *Automation in Construction*, Vol. 43, 144–155, doi: 10.1016/j.autcon.2014.03.010
- Zhang S., Boukamp F. and Teizer J. (2015), Ontology-based semantic modeling of construction safety knowledge: Towards automated safety planning for job hazard analysis (JHA), *Automation in Construction*, Vol. 52, 29–41, doi: 10.1016/j.autcon.2015.02.005
- Zhao X., Verhagen W.J.C. and Curran R. (2015), Estimation of aircraft component production cost using knowledge based engineering techniques, *Advanced Engineering Informatics*, Vol. 29, 616–632, doi: 10.1016/j.aei.2015.05.004
- Zheng P., Zhao G., Torres V.H. and Rios J. (2012), A knowledge-based approach to integrate Fuzzy Conceptual Design Tools and MOKA into a CAD system, *Proceedings - 2012 7th International Conference on Computing and Convergence Technology (ICCT, ICEI and ICACT), ICCCT 2012*, 1285–1291, Retrieved from [https://www.researchgate.net/publication/259828884\\_A\\_Knowledge-based\\_Approach\\_to\\_Integrate\\_Fuzzy\\_Conceptual\\_Design\\_Tools\\_and\\_MOKA\\_into\\_a\\_CAD\\_System](https://www.researchgate.net/publication/259828884_A_Knowledge-based_Approach_to_Integrate_Fuzzy_Conceptual_Design_Tools_and_MOKA_into_a_CAD_System)
- Zou Q. and Feng H.Y. (2020), A decision-support method for information inconsistency resolution in direct modeling of CAD models, *Advanced Engineering Informatics*, Vol. 44, 101087, doi: 10.1016/j.aei.2020.101087