REVERSE ENGINEERED DESIGN AUTOMATION: APPLYING KNOWLEDGE BASED ENGINEERING TECHNIQUES TO A CASE OF AUTOMOTIVE FIXTURES DESIGN CONFIGURATION

Johansson, Christian

Blekinge Institute of Technology

ABSTRACT

In the production of automotive body components, fixtures are an important part of the ongoing work on geometrical assurance. The fixture is uniquely defined for each component, and the design and configuration of these are time-consuming and takes a lot of effort. The objective with this paper is to explore the use of a design automation approach and application to semi-automate the configuration process of the fixture product. The paper presents an approach to automate the configuration of the fixtures in a flexible way, by reverse engineering the configuration of the fixture product from a generic blueprint that represents the expected outcome of the process, using a knowledge-based engineering approach applied to a computer aided design (CAD) environment. A reverse-engineered design automation toolbox for a CAD-software is developed. The toolbox is developed to lead a user through the configuration process, in the way that the experts want it done, end-to-end, making use of some unconventional solutions from a design automation perspective.

Keywords: Knowledge management, Case study, Computer Aided Design (CAD), Design Automation, Customization

Contact:
Johansson, Christian
Blekinge Institute of Technology
Mechanical Engineering
Sweden
christian.m.johansson@bth.se

Cite this article: Johansson, C. (2019) ‘Reverse Engineered Design Automation: Applying Knowledge Based Engineering Techniques to a Case of Automotive Fixtures Design Configuration’, in *Proceedings of the 22nd International Conference on Engineering Design (ICED19),* Delft, The Netherlands, 5-8 August 2019. DOI:10.1017/dsi.2019.164

1 INTRODUCTION

The design process paradox (Ullman, 2010), see expanded and adapted version in figure 1, captures the catch-22 that many product developing organizations experience in the early phases of their product development phases. At the beginning of the design project, when the possibility and freedom to affect the design and thus also the committed costs are easiest to influence, the available knowledge is also at its lowest. Conversely, when the developers have more knowledge, it is effectively too expensive and/or too late to have a substantial impact on the design decisions. Similarly, Calkins *et al*. (2000) identified that a substantial part of the cost (as much as 70%) is committed already by the end of the conceptual design stage. Many strands of research into supporting and improving product development focuses on moving the curves in the desired directions, which is illustrated by the dashed curves in figure 1.

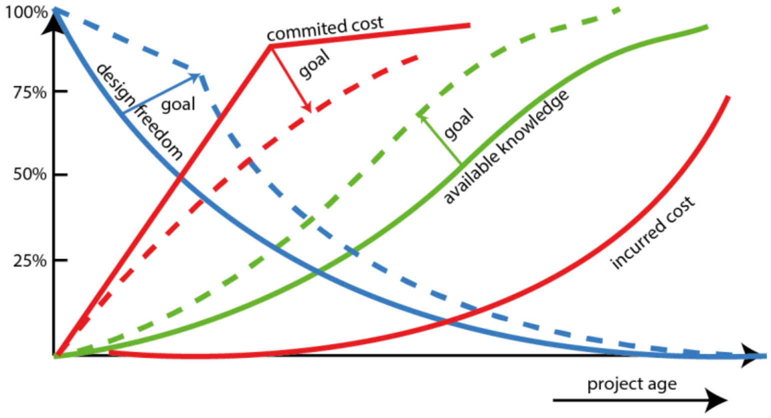

Figure 1. Expanded version of design process paradox; adapted from (Ullman, 2010) and (Verhagen et al., 2012).

This is also the motivation in the case of knowledge enabled engineering (Bertoni, Johansson and Bertoni, 2015), where knowledge-oriented tools and methods are applied to move downstream knowledge more upstream – or to say, to utilize previous experiences and learnings in the current project. One of these approaches is Knowledge-Based Engineering (KBE), which is a technology and method to capture and reuse engineering knowledge to automate CAD-based engineering design activities (Rocca, 2012). The allure of KBE is to shorten lead time of these activities and simultaneously increase quality, by alleviating routine design activities from the workload of the designers (Verhagen *et al*., 2012).

In the production of automotive body components, fixtures are an important part of the ongoing work on geometrical assurance. They are used to control that the manufactured part, often formed through a stamping, adhere to the expected form, within given tolerances. A fixture setup is built-up by several holders connected to a frame (see figure 2). The positioning of the holders is relative to readily defined reference points on the automotive body component, which are positioned so it is possible to quickly measure at a set of defined control points. By using these fixture setups, it is possible to accelerate the number of measures that can be made.

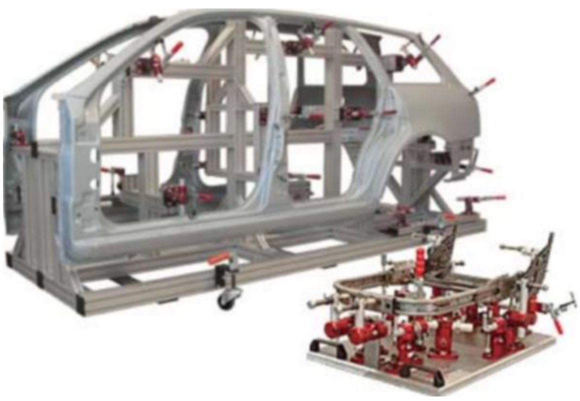

Figure 2. A fixture setup for control measuring.

https://doi.org/10.1017/dsi.2019.164 Published online by Cambridge University Press

Most approaches to KBE and design automation are to do generative design, where there are rules in place and a known quantity of the hardware components or assembly that can be varied or adapted according to some pre-defined design intent. The challenge with the fixtures is that an outsider designer will insert an essentially unknown component into the design assembly and the application should be able to define product concepts around on this component, where the final product is a combination of the customer component and the fixture assembly and components. At first glance this is not a capability that is offered by the CAD-based design automation packages, who generally focus on the former.

This paper reports on an approach to reverse engineer this automotive fixture product concept by means of design automation. Efficiency gains would come both from more automation and from turning design capabilities, which are governed by design automation rules, over to the hands of the customers, who are designers in the customer companies. This implies that the customer should be more in the driving seat and co-create the final product with the (limited) assistance of experts and automated design tools. With this approach, they can explore various fixture configurations for their parts.

From this frame of the challenge, the guiding question for this work has been "how should a reverse engineered design automation application be designed in a contemporary CAE system?". From this an important sub-question has been derived, "what are the specific requirements or limitation on product models?", because some limits became apparent during the development of the solution.

From this, the paper proceeds to review some central concepts for the work, followed by explanation of research approach. Thereafter the industrial case implementation of the reverse engineered design automation (REDA) toolbox is presented. The paper then discusses important topics to consider in comparison to more standard approaches in design automation as well as further developments in future research to really assert that this is a desirable and viable way forward.

2 THEORY

2.1 Design automation and knowledge-based engineering

Design automation deals with automating design tasks that are normally conducted manually by the engineers and designers. From Cederfeldt and Elgh (2005), Design Automation is defined as: "*Engineering support by implementation of information and knowledge in solutions, tools, or systems, that are pre-planned for reuse and support the progress of the design process. The scope of the definition encompasses computerized automation of tasks that directly or indirectly are related to the design process in the range of individual components to complete products.*" (p.2)

Therefore, it deals with - via computer - preparing and automating design tasks that are normally carried out in a manual fashion by engineers and designers.

Knowledge-based engineering (KBE) is defined by La Rocca (2012) as "*... 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 time and costs of product development by means of [...] automation of repetitive and non-creative design tasks [and] support of multidisciplinary design optimization in all the phases of the design process.*"

Verhagen *et al*. (2012) see KBE as a way of working, achieving design automation and knowledge retention. A KBE-system is closely related to Computer Aided Design (CAD), often implemented as modules in the CAD-systems, where the developers write code to manipulate a CAD-representation of a product in the KBE-application. Inputs are used to analyse and redesign a product model in order to generate an output design (Rocca, 2012), see figure 3.

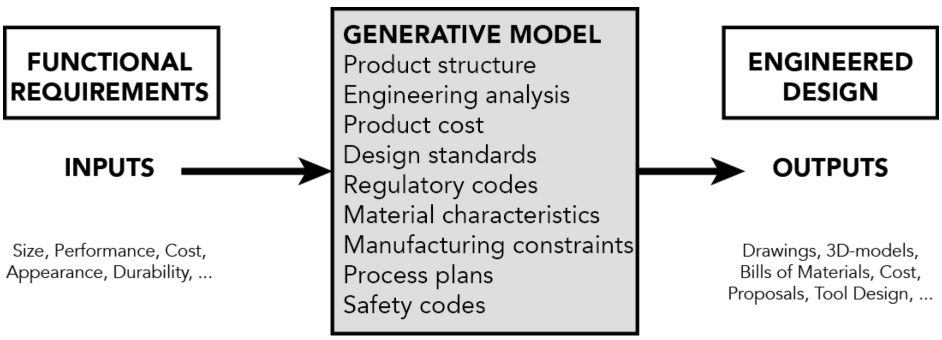

Figure 3. Generative KBE Application, adapted from (Rocca, 2012).

https://doi.org/10.1017/dsi.2019.164 Published online by Cambridge University Press

From a knowledge perspective, KBE is motivated and supported by the application of a knowledge lifecycle (Stokes, 2001), see figure 4. The knowledge lifecycle is devised to support the identification, justification, capture, formalization, packaging, and activation of knowledge-support solutions on specific design activities or entities.

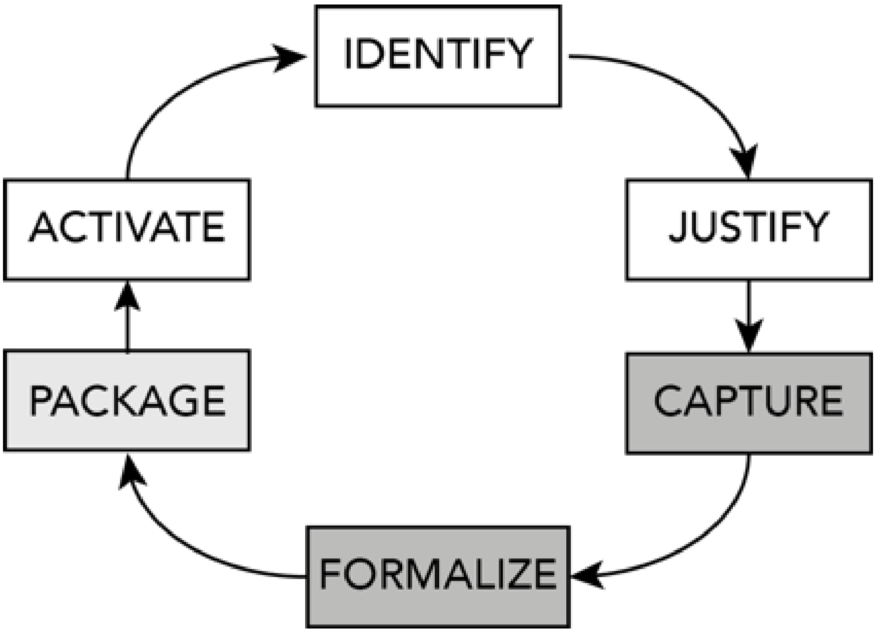

Figure 4. KBE lifecycle, adapted from (Stokes, 2001) – accentuating its focus on primarily knowledge capture and formalization.

One approach to facilitate methodologically structuring knowledge and conceptually creating KBE application is MOKA (Stokes, 2001), which stands for Methodology for Knowledge-Based Engineering Applications. Although supporting the whole KBE lifecycle, the focus of MOKA is on the knowledge-processes of the capture and formalization steps to support the KBE-implementation. With MOKA, different stakeholders (e.g., experts, knowledge engineers, developers, managers, and end-users) who are affected by, and has interest and decision power in the KBE-application will be considered and supported in its conception, development, and deployment.

At the centre of a design automation application are various types of rules (i.e., logic rules, math rules, geometry manipulation rules – see (Rocca, 2012)), which control the flow of selections and decisions being made by the system, and thus allows more advanced capabilities to be captured, than what a parametric CAD application can capture otherwise.

Arguments for adopting KBE (Verhagen *et al.*, 2012) relate to rationalisation and automation of conceptual and preliminary design, because here a lot of the costs are committed. With capabilities, offered by KBE, to reduce the amount of time-consuming routine work in the early phases, there is an opportunity to spend this gained time on more creative exploration of alternatives. Knowledge re-use in a guided KBE framework can help save time, and thus either reach markets faster or to allow the engineers to do more iterations of the same design tasks in the same amount of time, or both (Verhagen *et al.*, 2012). Similarly, KBE can be used also to support mass-customization (Vadoudi, 2012), where the use of knowledge sources to drive variation of product features allow designers to create variant designs with little additional work effort.

2.2 Reverse engineering

Traditionally, design is the development of original solutions; starting from an idea that is transformed into a final geometry. Herein, the role of CAD is essentially to digitize those geometries. If there is already a part in place, there is a good reason to apply a reverse engineering approach to this digitization (Motavalli, 1998) and to capture the shape of the existing part. By use of measuring machines or laser scanners, the surface of the component can be captured and digitized. Rekoff (1985) defined reverse engineering as "*...the process of developing a set of specifications for a complex hardware system by an orderly examination of specimens of that system." (p.244)*. Thereafter, designers can either completely replicate the component, or make enhancements to it.

https://doi.org/10.1017/dsi.2019.164 Published online by Cambridge University Press

Generally, there are three main steps in the approach to reverse engineering (Motavalli, 1998); part digitization, data segmentation, and part modelling.
Chikofsky and Cross (1990) break the notion of reverse engineering down further - into the concepts of forward engineering, reverse engineering, redocumentation, design recovery, restructuring, and reengineering - to depict the back-and-forth the moving between requirement, design, and implemented form and maturity of the product.
In this paper, reverse engineering is viewed as a way of achieving a digital representation of a product that is already known.

3 RESEARCH APPROACH

The approach in this paper has been to develop a demonstrator, that incorporates design automation of CAD geometry, using a Knowledge-Based Engineering (KBE) approach. Oriented around the design research methodology (Blessing and Chakrabarti, 2009), this paper deals with primarily the prescriptive part of the project, where a demonstrator is developed to address the challenge that is focal point of the research. In developing KBE applications, there are specific development methodologies to assert that the applications are developed according to strategies and plans. One of the more prevalent methodologies is MOKA (Stokes, 2001), as presented earlier. MOKA essentially centres on the Knowledge Lifecycle (see Figure 4 in the previous section), with a special emphasis on the knowledge processes, that is, the Capture and Formalize steps, before allowing the developers to package the application in their system of choice. The needs for the demonstrator was elicited via a series of reoccurring semi-structured interviews with two key experts in the partner company. New interviews were conducted on and off as new challenges arose with the development and rationale for choices needed to be explained. Similarly reporting of progress happened periodically. In this approach, they explained the rationale for the design activity as well as the steps and activities that they go through to configure the fixture products today. This part was also aided by observing a video recording of the process, which the researcher could return to when needed.
The development process of this demonstrator has been iterative, which was illustrated in situations where the developer tried to replicate the manual steps of the design activity without relying on the manual and human cognitive actions, which are continuously performed in the manual design activity, but challenging to replicate and generalise by use of coded rules. In these instances, the researcher sometimes had to either go back to the experts for explanation or to find alternative approaches to solve the issue by circumventing this manual action. Further, sometimes the solution had to rely on restrictions and conventions in the part modelling on behalf of the customer, which reflected the sub-research question presented earlier. As the work progressed, important knowledge elements were identified, which were considered for implementation and as guidelines for the user.

4 INDUSTRIAL CASE OF REVERSE ENGINEERED DESIGN AUTOMATION

This section will present the results being a CAD-based tool for reverse engineering a product configuration using design automation and KBE. First, the approach to elicit and model the knowledge is presented, followed by a presentation of the implemented demonstrator.
As part of the knowledge elicitation process, it soon became apparent that the user in mind for this tool would be a customer, likely - but not necessarily - being a designer at the customer firm. In this scenario, the idea is to allow the customers the freedom to explore their own configurations using a tool, in a similar way as is common with configurators for cars or kitchens. It could then free up time and effort for the provider, but it would also require a full capture and codification of the configuration process in the design automation tool, because it would not be feasible to assume that every customer will be skilled enough about the what is and is not allowed in the fixture configuration process. This scenario (see figure 5) begins with a user designing their parts. This is then uploaded to the configurator environment in either a proprietary or neutral file format. The application is then started, and a new project is launched. The first thing the user will do is then to load the part in the configurator environment. Then they will search for their connector points. As they are found, the user will be presented with a library of different holders, which can be selected and imported to the environment. As the holders are imported, they will also be constrained. This will be done for each connector point the user wants to utilise. Thereafter, the system needs to analyse the position and configuration of these holders. Some support structures will be added and then a base frame made

https://doi.org/10.1017/dsi.2019.164 Published online by Cambridge University Press

from aluminium profiles will be imported, based on the configuration of the holders. The frame will then be constrained with the holders, and the final step is to generate a bill of materials (BoM) as well as reports containing drafting and required configuration information to be able to replicate the virtual definition in real life. An important step at the end of the configuration process is to validate that the configuration follows the rules and requirements, which is done by the producer.

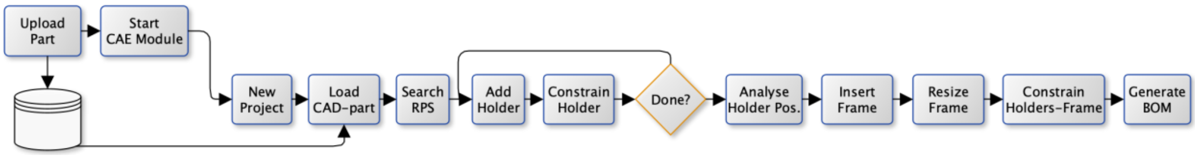

Figure 5. Workflow of configuration activity captured in design automation toolbox.

Figure 6 depicts the breakdown of the product, as it will look in generic terms when the configuration of the fixture is completed. The challenge with this case - and what makes it particularly interesting from a development point of view - is that the exact configuration of customer component is unknown, and fluid as far as is concerned in the design automation application. This means that there is little control of the geometrical structure/layout of the part supplied. In the manual configuration scenario, this poses little problems as an experienced designer is flexible enough and knows where to place the connectors. But with the potential true novice customers behind the keyboard, the application must cater to this lack of experience. As the process of configuring the assembled product starts with this part supplied by the customer, it must essentially be flexible from the beginning of the configuration process. Therefore, there needs to be an approach to this that maintains the standardisation element of a design automation configurator, while also allowing for flexibility.

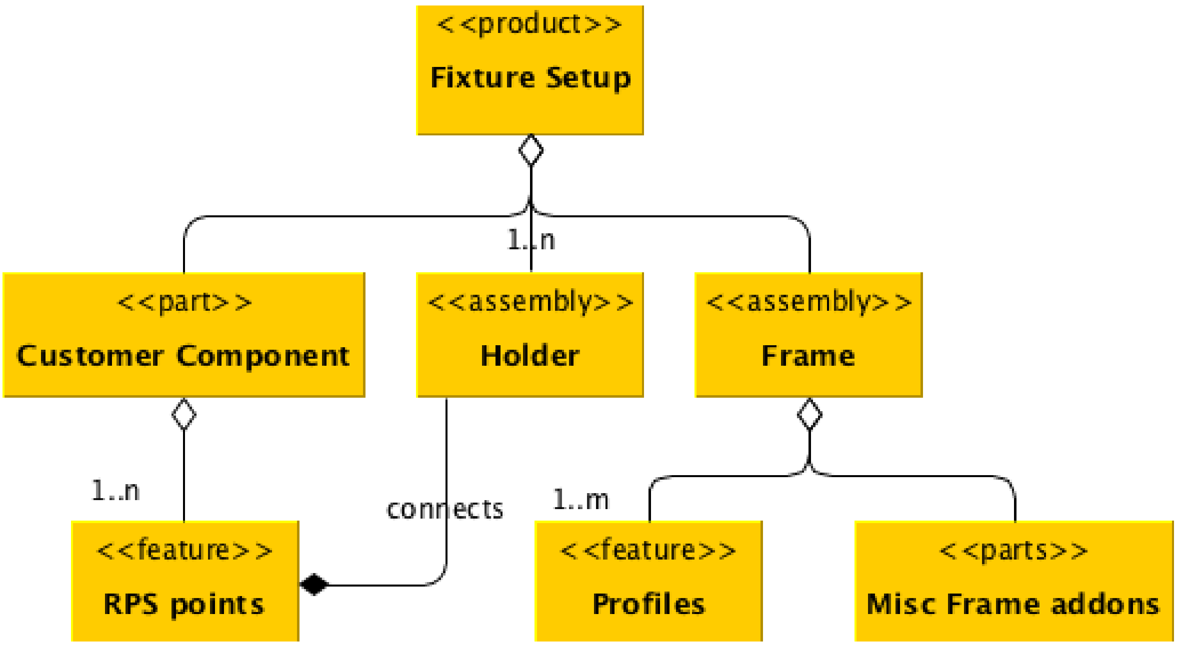

Figure 6. Product breakdown, following a Moka MML/UML-style structure notation.

A reverse engineering approach was elected, with starting from a generic outline of the finished product, and elect to mandate some convention (c.f. convention over configuration (Chen, 2006), as made popular with the Ruby On Rails programming language) - type and naming of required reference geometry - in how the designers at the customers will define their geometry, specifically how they define the interfaces that will connect with the support structure. Thereafter, since - as aforementioned - specific functionalities for configuring unknown geometries did not exist, the strategy was to make use of the built-in search and measurement functions as well as keeping a close check with the application's object model definition. Knowing the hierarchy of the object model and searching for geometries that are defined according to conventions, it was possible to construct an approach by which to follow that allowed to automate configuration decisions of a generic body component.

The core of the reverse engineered design automation (REDA) application is an addon toolbox (see figure 7), developed as a stand-alone application coded in visual basic, which taps into the APIs and object models of the CATIA CAE software as well as Excel that is used both as a database for accessing pre-existing knowledge and information - and as the main reporting tool. The toolbox is oriented according to the suggested steps to take, as defined in the knowledge capture step. It starts by creating a new assembly in the CAE-system and importing the supplied fil from a central storage. Thereafter the user is prompted to identify the supplied reference geometry by naming convention (e.g., points Z1, Z2, etc.). The application utilizes the built-in search function and loads up the points as variables. From a library that is coded into the application, the user can then select the holder that they believe they want, with guidance from information in the toolbox. As all holders are imported, the next tab of the interface, and part of the process, helps the user to select the frame of choice. This is imported, analysed, and constrained with the holders. Here orientation and configuration can be adjusted. Finally, the system helps the user create reports and visualisations in an Excel spreadsheet (key data is exported to predefined ranges in the spreadsheet), which are submitted to a control function.

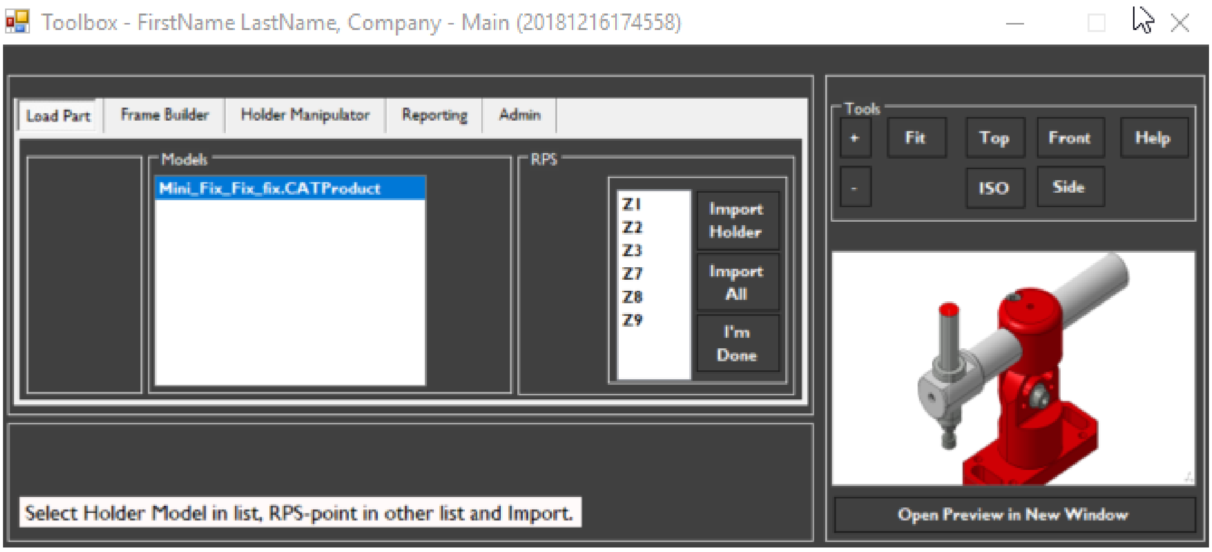

Figure 7. Demonstrator application Toolbox, ready to import holders to RPS-points on customer part.

Because there may be some decisions - relating to either personal taste or conventions - that the user wants to have control over, the decision was to implement the toolbox in a semi-automated way. Some composite activities are automated and periodically the users will make their decisions (e.g., about how many holders to insert, which holder to insert, the layout of the support profile, and more), as permitted by the application and supported with guidelines built into the user interface. All these engineering choices and commands are implemented as unique buttons with support guidelines that are shown and hidden based on the current position of the configuration process. Figure 8 shows the configurator mid-action of configuring a fixture for an automotive body component.

https://doi.org/10.1017/dsi.2019.164 Published online by Cambridge University Press

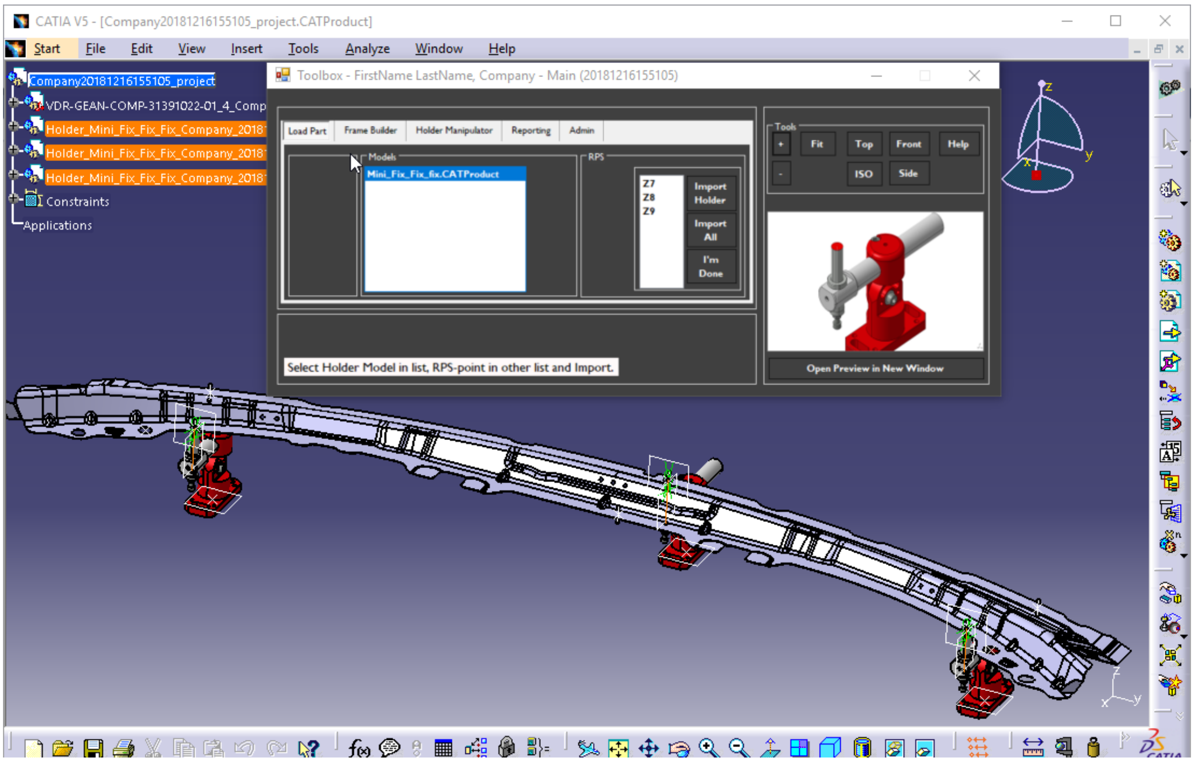

Figure 8. Screenshot of CAE application and toolbox in a CATIA environment.

Figure 9 shows a finished version of the CAD-definition of the fixture.

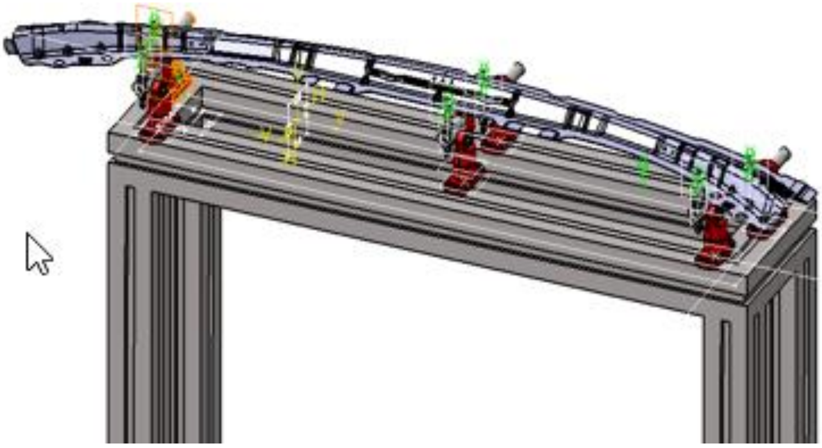

Figure 9. A final configured fixture.

5 DISCUSSION

This paper has presented an approach to creating an increased level of automation for the process and activity of configuring these automotive fixtures that have been in focus. As this is a time-consuming and routine activity, albeit, with some level of complexity, this work has focused on exploring and finding an approach for how to achieve this level of automation.

The challenge of this work has been to push through with the reverse engineered design automation (REDA) approach. With generative design automation (Rocca, 2012), as touched upon before, there is most often a known base definition of the solution that is explored in different configurations based on defined rules and parametrised CAD-models. With the REDA-approach, we know beforehand that we do not know enough beforehand about the final solution to generate the solution in this way. Traditionally, reverse engineering is about scanning or measuring physical geometry. With the REDA-approach, the reverse engineering approach is a little bit different, where the focus lies on automating the build and adaption of an assembly of various existing parts towards a configuration that is almost known. To go from almost to a fully known and controlled endpoint, the choice has been to adopt a

https://doi.org/10.1017/dsi.2019.164 Published online by Cambridge University Press

convention over configuration (Chen, 2006) approach. Because the APIs of most CAD-software is primarily developed to allow coding in variations in the build-up and configuration of known elements, essentially extended parametrization, all the functions depend on varying things with pre-defined references to their objects. We need to work our way backward from the desired situation in principle and work with replicating user behaviour in using search and measure functions. The application must be generic so that it is applicable in most configuration scenarios, rendering the approach substantially more cumbersome than generative design automation, but still viable if there are good insights into the design principles and conventions of the parts that are used by the toolbox.

As the motive for design automation is to improve the efficiency of the design process (Rocca, 2012), the question is; is this achieved with this approach and toolbox? At this stage the work has focused on how, and even what is needed, to go from beginning to the end of the configuration process. With the approach, it is possible to go from beginning to end, if the users adhere to naming and configuration instructions, and thereby can relatively swiftly have their first concept in place. Also, a benefit is that the application can embed instructions for the user, which can be provided at the right time of the process to configure. The approach achieves automation on the micro-level, in each of the steps and even connecting some of the steps. Still, it remains to be seen how to reap benefits from expanding on decision making automation capabilities that more real artificial intelligence might offer.

In an extended implementation, this toolbox capability is provided as an online remote application, via an application sharing platform built into e-commerce enabled web environment. This way the users/customers do not need to bother about installing and maintaining the toolbox as well as not needing to worry about licence fees, where all these considerations fall on the responsibility of the provider. The customers will log in, upload their model geometries and configure their fixture of choice, and then place an order that the provider will verify, and then produce and deliver to the customer.

6 FUTURE WORK

As this paper mainly covers the prescriptive part of the Design Research Methodology, the next important step for the specific research is to evaluate and measure if it achieves the desired benefits of reduced transaction costs and more customers for the industrial partner, based on a more competitive design offering. As this is a prototype, there are still more developments and testing needed to be able to compare the before and after cases.

Another approach that could be interesting to explore is to, by used of discrete event simulation, model the scenarios being represented by the current and the envisioned approach.

More replication is also needed to see what other cases can be applicable and be better served by this type of design automation approach.

Finally, it is also interesting to further explore includes how to incorporate the business model closer with the design automation configurator and specifically how different choices in configuration could also extend to whether a total offer (Alonso-Rasgado, Thompson and Elfström, 2004) would be a viable way of purchasing a solution, possibly in conjunction with the e-commerce enabled web environment..

7 CONCLUDING REMARKS

This paper has explored the use of knowledge-based engineering techniques in a reverse engineering case of the configuration of automotive fixtures.

The guiding questions centred on how to implement a reverse engineered design automation approach in a contemporary CAE-solution, as well as on limitations to CAD-models that are necessary to achieve implementation.

The work started with capturing expert knowledge from the engineers in the partner company that provides the fixtures. Based on the elicited knowledge and iterative design approach of a reverse engineered design automation toolbox for a CAE package was developed. The toolbox was developed to lead a user through the configuration process, in the way that the experts want it done, end-to-end making use of some unconventional solutions from a design automation perspective, tapping into functions relating to search and measure in a way that is not usually used for generative design automation. In addition, the approach requires that instructions are in place for how the user defines their geometries, especially with using supporting reference geometries and how these are named to enable the system to easily identify them.

https://doi.org/10.1017/dsi.2019.164 Published online by Cambridge University Press

REFERENCES

Alonso-Rasgado, T., Thompson, G. and Elfström, B.-O. (2004), “The design of functional (total care) products”, *Journal of Engineering Design*, Vol. 15 No. 6, pp. 515–540. https://dx.doi.org/10.1080/09544820412331271176.

Bertoni, M., Johansson, C. and Bertoni, A. (2015), “Knowledge enabled engineering”.

Blessing, L. T. M. and Chakrabarti, A. (2009), “DRM, a design research methodology”, *DRM, a Design Research Methodology*. https://dx.doi.org/10.1007/978-1-84882-587-1.

Calkins Egging N.B. and Scholz C.B D. E. (2000), “Knowledge-Based Engineering (KBE) Design Methodology at the Undergraduate and Graduate Levels”, *International Journal of Engineering Education*, Vol. 16 No. 1, pp. 21–38. Available at: http://www.scopus.com/inward/record.url?eid=2-s2.0-0034399807&partnerID=40&md5=defac65000c96652c084c72408e394e5.

Cederfeldt, M. and Elgh, F. (2005), “Design Automation in SMEs-Current State, Potential, Need and Requirements”, in *Inte*, pp. 1–15. Available at: http://search.informit.com.au/documentSummary;dn=390095181685561;res=IELENG.

Chen, N. (2006), *Convention over Configuration*.

Chikofsky, E. J. and Cross II, J. H. (1990), “Reverse Engineering and Design Recovery: A Taxonomy”, *IEEE Software*, pp. 13–17. https://dx.doi.org/10.1109/52.43044.

Motavalli, S. (1998), “Review of reverse engineering approaches”, *Computers & Industrial Engineering*, Vol. 35 No. 1–2, pp. 25–28. https://dx.doi.org/10.1016/S0360-8352(98)00011-4.

Rekoff Jr., M. G. (1985), “On Reverse Engineering”, *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-15 No. 2, pp. 244–252.

Rocca, G. La (2012), “Knowledge based engineering: Between AI and CAD. Review of a language based technology to support engineering design”, *Advanced Engineering Informatics*. Elsevier Ltd, Vol. 26 No. 2, pp. 159–179. https://dx.doi.org/10.1016/j.aei.2012.02.002.

Stokes, M. (2001), “Managing engineering knowledge: MOKA: methodology for knowledge based engineering applications”, Professional Engineering Publishing Limited, London, UK.

Ullman, D. G. (2010), *The Mechanical Design Process*, Mechanics of Materials.

Vadoudi, K. (2012), “Proposal of a knowledge-based engineering methodology for mass customization”, (July 2013), pp. 287–294.

Verhagen, W. J. C. C., et al. (2012), “A critical review of Knowledge-Based Engineering: An identification of research challenges”, *Advanced Engineering Informatics*. Elsevier Ltd, Vol. 26 No. 1, pp. 5–15. https://dx.doi.org/10.1016/j.aei.2011.06.004.

ACKNOWLEDGMENTS

The research leading to these results has received financial support by the Swedish Knowledge and Competence Development Foundation (Stiftelsen för kunskaps- och kompetensutveckling) through the Model Driven Development and Decision Support research profile at Blekinge Institute of Technology.