Extending the Use of Design Automation Within 3D-Modelling of Tool Inserts

A project investigating the possibility of reusing and adapting an existing design automation in a similar situation.

Amanda Svensson
Linköping University, Sweden

Supervisors: Olle Vidner, Joel Johansson
Examinator: Mehdi Tarkian
Author
Amanda Svensson
M.Sc. Student in Mechanical Engineering
Linköping University, Sweden

Supervisors
Joel Johansson
PLM & Digitalization Specialist
Thule Group AB, Sweden

Olle Vidner
Post Graduate Student
Division of Product Realization
Linköping University, Sweden

Examinator
Mehdi Tarkian
Associate Professor
Division of Product Realization
Linköping University, Sweden
Abstract
About eighty percent of all engineering work done today is repetitive, and about ninety percent of all engineering work consist of modelling minor changes. By replacing all the repetitive human engineering work with computers, the engineers could instead focus on creating new products or improving the existing. This could make companies more competitive and increase sales. Using design automation to model small changes would also enable companies to produce small batches at a lower cost.

This project is done in collaboration with Thule Group. One of Thule’s largest product categories contain roof racks. The company manufacture the roof racks at their warehouse in Sweden and all variant modelling of them are performed by their technicians and engineers. For every new car model that is released, a new variant of the attachment for the roof racks needs to be modelled. There are different types of brackets used in the attachments, whereas two of them are called Evo Clamp and Evo Flush. For each new car model, new tool inserts for the manufacturing of the brackets also needs to be modelled. In previous research, a design automation process of the tool inserts for Evo Clamp was created. This project aims to use the outcome from that research to create a new design automation process for tool inserts to create Evo Flush.

To execute the project, the DRM framework was used. To gather information, a literature study and an empirical study were performed. Furthermore, the design automation was created using VB.NET and Solidworks. To evaluate the outcome of the project, three factors were set up to test the process by. The outcome from this project was also compared to the outcome from the Evo Clamp research.

The results showed that it was difficult to reuse and adapt the previous research since the templates for the tool inserts of the two different brackets were modelled in two completely different ways. Therefore, the main conclusion from this project is that; if the intention is to automate a process, then this must be kept in mind when modelling the components and templates. To have concrete modelling guidelines seems to be even more important if the intention is to reuse code from one process when automating another.
Acknowledgement

I want to thank everyone who has guided and helped me through this master thesis project. It has been a challenge doing it on my own. Although, with all the amazing support I’ve gotten, I’ve never felt alone. Thank you.

Also, thank you to Thule for all the support I’ve received during my last few years of university studies. I’m very grateful and happy to be able to finish our “university journey” with this project. I have always felt welcomed here and I hope our paths cross again in the future.
Table of Contents

1. Introduction .......................................................................................................................... 8
   1.1. Background .................................................................................................................. 8
   1.2. Aim ............................................................................................................................... 10
   1.3. Research purpose ....................................................................................................... 11
   1.4. Delimitations .................................................................................................................. 11
   1.5. Deliverable .................................................................................................................... 11

2. Approach ............................................................................................................................. 12
   2.1. Design Research Methodology (DRM) ........................................................................... 12
   2.2. Literature Review ......................................................................................................... 13
   2.3. Interview Methodology ............................................................................................... 13

3. Applying the Approach ....................................................................................................... 14
   3.1. DRM ............................................................................................................................. 14
   3.2. Research Clarification .................................................................................................. 14
   3.3. Descriptive Study I ..................................................................................................... 14
      3.3.1. Interviews ............................................................................................................. 14
      3.3.2. Studying Recorded Material ................................................................................. 14
      3.3.3. Studying the Evo Clamp script ............................................................................. 15
   3.4. Prescriptive Study ......................................................................................................... 15
   3.5. Descriptive Study II ..................................................................................................... 15

4. Theoretical Framework ......................................................................................................... 16
   4.1. Product Development Process (PDP) ........................................................................... 16
   4.2. Computer-Aided Design (CAD) .................................................................................. 16
   4.3. Design Automation (DA) ............................................................................................ 17
   4.4. Knowledge-Based Engineering (KBE) ........................................................................ 17
   4.5. Knowledge Objects ...................................................................................................... 17
   4.6. Visual Basic .NET (VB.NET) ..................................................................................... 18

5. Results .................................................................................................................................. 19
   5.1. Research Clarification ................................................................................................... 19
      5.1.1. Literature Review .................................................................................................. 19
   5.2. Descriptive Study I ...................................................................................................... 20
      5.2.1. Interviews ............................................................................................................. 20
      5.2.2. Templates ............................................................................................................. 20
      5.2.3. Comparison ......................................................................................................... 21
   5.3. Prescriptive Study Iterations ........................................................................................ 21
      5.3.1. First Iteration – Identification .............................................................................. 21
      5.3.2. Second Iteration – Imitate the Manual Process ................................................... 22
      5.3.3. Third Iteration – Identify Knowledge Objects ...................................................... 22
      5.3.4. Fourth Iteration – Write a Functional Script ....................................................... 24
List of Figures

Figure 1 A roof rack attached with Evo Clamp brackets to a car without railings. (Thule WingBar Evo, 2021) .......................................................... 9
Figure 2 A CAD model of an Evo Clamp bracket. .......................................................... 9
Figure 3 Roof racks attached to the railings using Evo Flush brackets inside the foot pads. (Thule WingBar Evo, 2021) .......................................................... 10
Figure 4 CAD models of the two parts of an Evo Flush bracket .................................. 10
Figure 5 The DRM framework (reproduced from Blessing & Chakrabarti, 2009) ............... 12
Figure 6 The overall workflow based on DRM .............................................................. 14
Figure 7 The product development process reproduced from Ulrich & Eppinger (2016) ....... 16
Figure 8 A screen shot from Howtoration Suite showing a Knowledge Object ................. 18
Figure 10 The finished tool inserts and documentation for the Evo Clamp manufacturing tool (reproduced from Heikkinen, 2018) .......................................................... 19
Figure 11 Evo Flush tool insert template, and the bracket template. The top part of the template is called a pad, and the bottom part is called a punch ........................................ 20
Figure 12 Evo Clamp tool insert template, and the bracket template. The templates are all modelled using surface modelling ......................................................... 21
Figure 13 The automation process for creating a folder including the needed templates and brackets. The process begins with creating a folder to store the bracket variant in, and then all necessary files are added to the folder. Lastly, the files are renamed .................................................. 22
Figure 14 The automation process for preparing the bracket, starting with opening the bracket from the variant folder and then preparing it to match the tool inserts template before saving it to the variant folder again .................................................................................. 22
Figure 15 The process for inserting and mating the bracket. The process starts with opening the tool inserts template and then inserting the bracket and placing it correctly. Lastly, the assembly is saved ........................................................................................................... 23
Figure 16 The automation process of the pad and the punch. The processes are executed through the assembly. Therefore, the first step must be to edit the correct part. The parts are then edited to match the bracket and lastly everything is saved .................................................. 23
Figure 17 The updated process for the tool inserts ................................................................ 25
Figure 18 The variant part of an Evo Flush bracket with reference planes ............................ 26
Figure 19 A bracket before and after automation ............................................................... 27
Figure 20 The different buttons used to automate the pad .................................................. 27
Figure 21 The different steps of the pad automation .......................................................... 28
Figure 22 Clarified hole sketch around the green holes ....................................................... 28
Figure 23 The buttons different buttons used to automate the punch ................................. 29
Figure 24 The different steps of the punch automation that occurs when the automation script runs. Every step is connected to a button in Figure 23 .................................................. 29
1. Introduction

In every industry, there are several different companies competing to be the leading manufacturer or retailer in their market segment. The cost efficiency is an aspect usually discussed for a company to be competitive, nevertheless when working with customizable products or components. However, the cost efficiency is usually one of the most challenging aspects when offering small batches of a component or product (Pakkanen et al., 2016).

In some cases when creating customized components, there are only minor changes that need to be made (Encarnacao et al., 2012). Instead of modelling brand new components every time small changes are made, letting a computer do what is repetitive saves a lot of time and allows the engineers to use their creativity to invent new products instead (Tarkian, 2012).

1.1. Background

This project has been conducted in collaboration with Thule Sweden AB (Thule). Thule is a company that manufacture and sell, among other things, car equipment. One of their main product categories include their roof racks. They aim to create roof racks that fit 95% of the cars on the market (Heikkinen et al., 2018), and the roof racks are the main feature used for several of their products to work correctly. For example, products such as cargo carriers, bike racks and roof tents made by Thule, all depend on the roof racks.

The roof racks consist of several different brackets. Since all car models look different, the roof racks, and foremost the brackets, must also look different depending on the car model. This results in Thule, as for now, manufacturing almost six hundred different models of Evo brackets. When a new car model enters the market, chances are that Thule needs to create new Evo brackets for their roof racks to fit the new car model. This does not only mean that they need to create new brackets, but also new tool inserts for the manufacturing tool. Usually there are only minor changes that needs to be done.

To reduce the lead time of creating a new tool insert for Evo brackets, previous research has been made by Heikkinen et al. (2018). During the previous research, a design automation plugin for a CAD program was developed. The plugin made it possible to create the tool inserts autonomous from the 3D model of the requested Evo bracket. Although the plugin program works, it is only a prototype, and it is not yet in use. Furthermore, the plugin currently only works for one model of Evo brackets called Evo Clamp. The Clamp brackets are used in the foot pad of the roof racks to attach them to cars without railings. The foot pads are clamped around the roof by the doors, as seen in Figure 1. A CAD model of an Evo Clamp bracket can be seen in Figure 2.
For car models that have integrated railings, Evo Clamp brackets cannot be used. Instead, a different bracket model called Evo Flush is used. The Flush brackets consist of two parts and are integrated in the foot pad that is attached around the railings instead of directly around the roof. Just like the Evo Clamp brackets, the shape of these varies depending on the car model. The attachment of the Evo Flush brackets can be seen in Figure 3, while Figure 4 shows CAD models of the inner and outer part of an Evo Flush bracket.
Figure 3 Roof racks attached to the railings using Evo Flush brackets inside the foot pads. 
*(Thule WingBar Evo, 2021)*

Figure 4 CAD models of the two parts of an Evo Flush bracket.

The Evo Flush brackets, and the tool inserts, are currently developed manually by a technician at Thule. It does not exist any prototype of a plugin to create this model autonomous, as it does for the Evo Clamp brackets.

1.2. Aim

The aim of this project is to investigate how an existing design automation plugin for tool inserts can be reused to model other tool inserts. The project revolves around a specific problem about how the design automation of tool inserts for an Evo Clamp manufacturing tool can be reused to create autonomous tool inserts for an Evo Flush manufacturing tool. When the project is completed, the goal is to have a script that automatically models the tool inserts for the Evo Flush bracket manufacturing tool by clicking a button. The goal is also to assess the effects of the final design automation through an evaluation where the robustness, flexibility and reduced time is explored.
1.3. Research purpose

The main purpose of this project is to investigate how the knowledge from previous research from a similar situation can be reused and adapted to streamline the work of creating a new design automated process. The purpose is also to investigate the effects of the outcome.

The project was based on the suggestion that computers should do all that is repetitive and iterative, leaving the engineers to focus on the development that is creative and demands the use of human intuition. This could save both time and money, making a company more competitive on the market as they could focus on developing new products and improving their existing ones, instead of repeatedly modelling variations of a component, or in this case, tool inserts.

1.4. Delimitations

Since the original automation plugin is made to be compatible with Solidworks, other CAD programs will not be considered during this project. The research which this project is based on (Heikkinen et al., 2018) uses the code language VB.NET, therefore the same language will be used in this project. Also, the previous research uses a program called Howtoration Suite (Johansson, 2015) to run the script. The program is based on Knowledge Objects (Johansson & Poorkiany, 2019) and is partially commercialized in the mean that it is used daily by a company that produces polymer tools. However, since the program does not affect the functionality of the automation created in this project, the script to run the automation will only be executed through Visual Studio.

The bracket manufacturing tool consist of several components. However, the only components that depend on the bracket design are the tool inserts. The inserts consist of a punch and a pad used for sheet metal bending. In consequence of this, the manufacturing tool model will be simplified to only include the tool inserts.

Furthermore, the Flush bracket consist of an inner and an outer part. However, because of the timeframe of the project, only the inner part will be considered. There are also Evo Flush brackets that need several bending stations in the manufacturing tool to obtain the right shape. These brackets will not be considered since deciding if several stations are necessary requires human intuition.

1.5. Deliverable

When the project is concluded, a design automation script that automatically creates a CAD model of tool inserts for an Evo Flush bracket manufacturing tool will have been delivered. If possible, the code from research regarding tool inserts by Heikkinen et al. (2018), will have been adapted and reused. The design automation will have been evaluated and conclusions regarding the effects of it will have been drawn.
2. Approach

2.1. Design Research Methodology (DRM)

DRM is a general research framework meant to help structuring design research (Blessing & Chakrabarti, 2009). The different stages of DRM can be seen in Figure 5.

![Diagram of DRM framework](image)

**Figure 5** The DRM framework (reproduced from Blessing & Chakrabarti, 2009)

During the first stage, *Research Clarification*, the focus lies on understanding and describing previous research to support the goal of the study. The second stage, *Descriptive Study I*, includes performing an empirical study to further investigate the outcomes of the first stage. The goal with this stage is to clarify and understand the current situation further. In the third stage, *Prescriptive Study*, the actual support is developed. The support is also evaluated to ensure that it meets the set requirements.

The final stage, *Descriptive Study II*, holds the final evaluation. The final evaluation can be performed in three different steps depending on the desired outcome. The first one is called *Method/Tool Evaluation*. It is used continuously during the development to ensure that the method can be tested. The second one is called *Application Evaluation* and is used to see that the applicability and usability reach the goal of the desired key factors. The third one is called *Success Evaluation* and tests the usefulness of the support based on the desired aims. (Blessing & Chakrabarti, 2002)
2.2. Literature Review

The literature review is an important tool when conducting research. To gather information about previous research, and to understand different perspectives of it is the foundation of a well-structured research process. Much like the interview methodology, there are different approaches to take when conducting a literature review. There are among others; the systematic, the semi-systematic and the integrative approach. What approach is appropriate varies depending on the desired outcome. (Snyder, 2019)

Literature reviews can be used when the researcher either wants to evaluate theory in a certain area, or to validate the accuracy of theories (Tranfield et al., 2003). The semi-systematic approach is usually used to overview the research area, although not unusually over time. The integrative approach, which is also called the narrative approach (Rother, 2007), is used to synthesize and critique (Snyder, 2019).

2.3. Interview Methodology

There are three types of interviews that can be conducted. There are the structured, the semi-structured and the non-structured interview. The structured interview includes predefined questions that typically only requires a straight answer. There is no space for the interviewee to sway of topic or add irrelevant information to their answer. The interviewer presents the exact same questions, in the exact same order, to all recipients. (Frances et al., 2009)

The semi-structured interviews on the other hand, are more flexible and the interviewees can present longer answers in open-ended questions. The semi-structured interviews, unlike the structured, allows the interviewer to clarify questions and different levels of language can be used when the questions are answered. (Lune & Berg, 2017)

The unstructured interviews do not include any predefined questions. The interviewing form is more often used to allow answers regarding a specific topic rather than specific questions. The interviewer asks broad open questions and holds a conversation with the interviewee, which take its turn depending on the answers from the interviewee. Unstructured interviews are usually used when there is little knowledge known about the subject beforehand. (Frances et al., 2009)
3. Applying the Approach

3.1. DRM

The approach applied during this project is based on the DRM framework and can be seen in Figure 6. The different stages are further described under the subheadings in this chapter.

![Diagram of DRM framework]

Figure 6 The overall workflow based on DRM.

3.2. Research Clarification

The research clarification was about understanding the problem and gaining knowledge on how to solve it. This was done by executing a literature study. The studied literature included both ways to conduct a design automation project and how a similar problem had been solved previously. The outcome from the literature study for the project can be found under chapter 4. Theoretical Framework, while the case specific results can be found under chapter 5.1.1 Literature Review.

3.3. Descriptive Study I

The first descriptive study was divided into two main parts. The first one was interviews, while the other one was studying the former DA. Studying the former DA included studying videos where the manual process was explained and studying the DA script. The interviews were performed after a study visit at Thule where the manufacturing tools were shown, and the problem was explained.

3.3.1. Interviews

Two non-structured interviews were held with a technician who worked with modelling tool inserts. During the first interview, the technician was asked to explain how manufacturing tools work, and what he wished for to improve his work. CAD models of both the Evo Flush and Clamp manufacturing tools were shown and explained. The second non-structured interview was held remotely over Microsoft Teams. The technician was then asked to show and explain how he modelled a tool insert for the Evo Flush manufacturing tool. This interview was recorded to easily be able to follow the manual modelling process later in the project. A transcript of the manual modelling process can be found in Appendix I. The outcomes of the interviews are further shown in chapter 5.2.1 Interviews.

3.3.2. Studying Recorded Material

The design automation process for the Evo Clamp tool inserts had previously been recorded. Studying the videos was highly important to understand the automation process that had been
previously implemented. The videos helped to understand the process in a way the literature did not.

3.3.3. Studying the Evo Clamp script

Since a working plugin program for the Evo Clamp brackets already existed when this thesis started, researching, and understanding the program was important. This was done by watching the recorded videos and inspecting each step in the VB.NET script. By using Howtomation Suite (Johansson, 2015), each sub in the script could be executed schematically. Seeing each Knowledge Object and how they were connected visually made it easier to understand the process, rather than just going through the VB.NET script in Visual Studio.

3.4. Prescriptive Study

The development and testing of the DA was executed continuously and iteratively. The same bracket that was shown during the interviews was used as a reference throughout the project. The motivation for this was to easily compare the manual and automated process. The iterative results from the prescriptive study can be found under chapter 5.3 Prescriptive Study.

3.5. Descriptive Study II

Evaluating that the script worked was done iteratively in both the fourth and fifth iteration. However, after the fifth iteration when the script was seen to work for one bracket, a final evaluation was made.

To evaluate the outcomes of this project, three different factors were set up. The factors for the new script were set to reduced time, robustness, and flexibility. The time was compared to how long it usually takes for a technician at the company to perform the entire process. The flexibility was measured by testing different Evo Flush brackets to see if they worked in the automated tool inserts. The robustness was set by investigating the impact of parameters and component names in Solidworks.
4. Theoretical Framework

This chapter contains the outcome of the general literature review used to execute the project.

4.1. Product Development Process (PDP)

According to Ulrich & Eppinger (2016), the generic product development process can be described in six different stages. It starts off with planning the project, then the concept development process begins followed by how to integrate the product in a system, and then the details are set and designed. The last two steps are the testing and refinement, followed by the production ramp-up. The process can be used by different functions in an organization. However, the process considered here is used in the design function. All six stages of the PDP can be seen in Figure 7.

Figure 7 The product development process reproduced from Ulrich & Eppinger (2016).

Ulrich & Eppinger describes the different stages of the design process as following:

- **Planning** – Plan the product structure and consider different technologies and methods.
- **Concept Development** – Develop different design concepts and assess their feasibility. Also, create prototypes.
- **System-Level Design** – Create the product structure and define sub-systems. Also, refine the previous concepts.
- **Detail Design** – Go into detail of the design by choosing materials, defining the geometry, setting tolerances, and creating design control documents.
- **Testing and Refinement** – Test the product by checking the reliability, performance, and durability. Implement the necessary design changes found after the testing.
- **Production Ramp-Up** – Evaluate the created production material.

4.2. Computer-Aided Design (CAD)

Ivan Sutherland is usually considered “the father of CAD” as he developed Sketchpad in 1963. By then, most of the modelling was done in 2D. However, during the 70’s, the first programs for 3D modelling were invented, and in the late 80’s, parametric modelling entered the market as Parametric Technology Corporation released Pro/Engineer. From that, the CAD
modelling has continuously taken steps forward to eventually become the CAD modelling known today. (Tornincasa & Monaco, 2010)

CAD is widely used for design, analysis, simulations, and production preparations around the industry.

4.3. Design Automation (DA)

Design Automation is the automatic completion of design work through software (Spacey, 2017). It can mean a various of things such as autonomous technology design, infrastructure design or visual design. In this case, DA is seen as automatic CAD modelling.

Although CAD has indefinitely helped companies be more cost-efficient and to speed up the development process, approximately eighty percent of all CAD modelling is repetitive (Stokes, 2001). It has also been shown that about ninety percent of all CAD activities are variant modelling with small changes (Encarnacca et al., 2012). If some of all that is repetitive could be automated, the creativity connected to designing new products or improving old ones could increase (Cederfeldt, 2007). According to Tarkian (2012), machines should be used to perform repetitive, routine-like and non-creative tasks, leaving the engineers to perform the tasks that require intuition and creativity.

4.4. Knowledge-Based Engineering (KBE)

Knowledge-based engineering (KBE) has been shown to be a promising way of using design automation (Amadori et al., 2012). It can be described as an engineering method used to merge techniques for AI (artificial intelligence), OOP (object orientated programming), CAD and DA (Chapman & Pinfold, 2001). It is a research field focused on how methods and techniques can be used to reuse previous work when creating a new product or process. The idea of KBE is to enhance the product development process by reducing the time and cost it takes to create a new product (Verhagen et al., 2012).

4.5. Knowledge Objects

Johansson & Elgh (2019) define Knowledge Objects as “...bundles of human comprehensible knowledge representations and computer routines for the automated application of the represented knowledge.”. A Knowledge Object consists of two parts, an automation section, and a description section. The content in the automation section is mainly focused on being machine readable, while the content in the description section is set to be read and understood by humans. An example of content in the automation section is computer code, while an example of content in the description section is descriptions of product knowledge.

Knowledge Objects are meant to be used to update and change information autonomous over time easily and flexibly. By using Knowledge Objects and an object-oriented perspective of the product structure, a program called Howtomatication Suite was developed to enable just that, while also giving a clear view of the whole automation process. (Johansson, 2015)

An example of a Knowledge Object in Howtomatication Suite can be seen in Figure 8. In the figure, CreateProjectFolder is the Knowledge Object. It is connected to a VB.NET script that holds machine readable code, while the visible object can be expanded and holds information that can be read by humans.
4.6. Visual Basic .NET (VB.NET)

Visual Basic is an object-orientated programming language developed by Microsoft. .NET is a development platform created by the same company to enable development of different kinds of applications. .NET can be used with either C#, F# or Visual Basic (Microsoft, 2021b). When used together with Visual Basic, it is usually referred to as VB.NET. VB.NET is not to be confused with VBA, which is an event-driven programming language used to extend other applications, for example Office applications (Microsoft, 2021a).
5. Results

5.1. Research Clarification

As mentioned in chapter 3. Applying the Approach, this phase was about understanding the problem. Here, the result of the literature review that was performed to gain knowledge about the problem is shown.

5.1.1. Literature Review

The results from the general literature review for the project can be found in chapter 4. Theoretical Framework. Here, the results from the literature review of the previously executed research regarding tool inserts performed in collaboration with Thule is presented.

To see how similar tool inserts had been automated using DA, a licentiate thesis by Heikkinen (2018) was reviewed. Heikkinen investigated how product model extensions could be created using DA. He developed an automated Solidworks plugin to automate the modelling of tool inserts for the creation of Evo Clamp brackets as an example throughout the thesis. The outcome from the example in the thesis were an automated model of tool inserts for Evo Clamp manufacturing tools, and documentation. The outcome can be seen in Figure 9.

Figure 9 The finished tool inserts and documentation for the Evo Clamp manufacturing tool (reproduced from Heikkinen, 2018)

The thesis included three research questions, one about the state of the art regarding product model extension, one about the industrial requirements, and one about how product model extensions could be used to support multidisciplinary DA.

For the current project, the most relevant question was what the industrial requirements were. Heikkinen highlighted some challenges he saw throughout his thesis work. Most of them were connected to modelling flexible and robust CAD models and managing software updates.
5.2. Descriptive Study I

During this phase, the outcomes of the first phase was further investigate through interviews and by studying the former DA on a deeper level.

5.2.1. Interviews

A first non-structured interview was held with two technicians at Thule. During the interview, they showed how the tool inserts were connected to a manufacturing tool and how they had approximately six hundred variations of the Evo brackets. They stated that it took a lot of time to model each new bracket and tool insert, while it was only minor changes that needed to be made. They conveyed their wishes as wanting to reduce this manual work to be able to focus on less repetitive tasks.

A second interview with one of the technicians at Thule was held. The interview was non-structured, and the technician was asked to describe how he manually created the tool inserts for the Evo Flush manufacturing tool from the time that he received the bracket to the time that he had saved the tool inserts. The interview was recorded. The described process was then summarized and can be seen in Appendix I. Describing the process in detail took approximately 13 minutes. However, the manual modelling usually takes 5-10 minutes, according to the technician.

5.2.2. Templates

When modelling the tool inserts manually, it is never done from scratch. The technician modelling the inserts has an assembly template consisting of a pad, a punch, and a bracket. This template was used when automating the process in this project as well. The Evo Flush template is shown in Figure 10.

![Figure 10 Evo Flush tool insert template, and the bracket template. The top part of the template is called a pad, and the bottom part is called a punch.](image)

The bracket, punch, and pad templates are imports without almost any parametrical entities. The reason behind this is that they were modelled in a different CAD program before the company used Solidworks. Only a few entities were added after the import had been made.

The Evo Clamp tool insert had a similar template, as can be seen in Figure 11. However, these template components were all parametrical.
5.2.3. Comparison

Having access to both videos of the manual modelling of the tool inserts from Heikkinen’s thesis, and his automation script and plugin as well as the recorded process for the Evo Flush tool inserts made it possible to do a comparison of the different tool inserts to investigate their similarities and differences.

The videos, models and script showed major differences in the modelling of the Clamp and Flush brackets and tool inserts. In Table 1 some of the greatest and most important differences are highlighted.

Table 1. The differences between the Evo Clamp and Evo Flush brackets and tool inserts.

<table>
<thead>
<tr>
<th>Evo Clamp</th>
<th>Evo Flush</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bracket created using surface modelling</td>
<td>Bracket created using solid modelling</td>
</tr>
<tr>
<td>Tool insert templates are parametric</td>
<td>Tool insert templates are imports with no parameters</td>
</tr>
<tr>
<td>Bracket template is parametric</td>
<td>Bracket template is an import with no parameters</td>
</tr>
<tr>
<td>E-mailing is used for receiving brackets</td>
<td>A PLM system is used for retrieving brackets</td>
</tr>
</tbody>
</table>

5.3. Prescriptive Study Iterations

Here, each iteration from the prescriptive study is described, starting with the identification of what should be automated.

5.3.1. First Iteration – Identification

The first iteration was about identifying what should be automated from the manual process. Therefore, summarizing the interview and writing down the entire manual process with pictures helped to understand each manual step. The summary can be seen in Appendix I. This
iteration was performed already during the first descriptive phase since it included the material from the interviews.

5.3.2. Second Iteration – Imitate the Manual Process
The next step to understand the process included following the transcript and think how the steps could be automated. This iteration was iterative itself. The first time, the entire transcript was followed. Later, some steps were replaced by other functions to identify the best way to automate the process. During this phase, macros were also recorded to identify automation functions in VB.NET.

5.3.3. Third Iteration – Identify Knowledge Objects
After following the manual process, the Knowledge Objects could easily be identified. Therefore, one process for the administrative parts of the automation, one process for inserting the bracket, and three processes for editing the different components were written down. The processes can be seen in Figure 12, Figure 13, Figure 14 and Figure 15. Each box in the processes illustrates a Knowledge Object. These were then meant to be one sub each in the VB.NET script.

![Diagram](image1.png)

Figure 12 The automation process for creating a folder including the needed templates and brackets. The process begins with creating a folder to store the bracket variant in, and then all necessary files are added to the folder. Lastly, the files are renamed.

![Diagram](image2.png)

Figure 13 The automation process for preparing the bracket, starting with opening the bracket from the variant folder and then preparing it to match the tool inserts template before saving it to the variant folder again.
Figure 14 The process for inserting and mating the bracket. The process starts with opening the tool inserts template and then inserting the bracket and placing it correctly. Lastly, the assembly is saved.

Figure 15 The automation process of the pad and the punch. The processes are executed through the assembly. Therefore, the first step must be to edit the correct part. The parts are then edited to match the bracket and lastly everything is saved.
5.3.4. Fourth Iteration – Write a Functional Script

During this iteration, the processes from the third iteration should be converted to code. The goal of this phase was to create a functional script that performed the automation correctly. To test the separate functions, buttons in a Windows Form through Visual Studio were used. This phase included plenty of testing. Each button was connected to a Knowledge Object which executed a function in Solidworks. The functionality of the buttons is further explained in chapter 5.3.8. The Pad and 5.3.9. The Punch. Everything in this phase was iterative and the processes from the third iteration phase were updated along the way. The updated process for the punch and pad can be seen in Figure 16. These were now the same as the different buttons. The processes for saving documents and preparing the bracket remained the same throughout this iteration.
Figure 16 The updated process for the tool inserts.
5.3.5. Fifth Iteration – Clean up the Code
During the fifth iteration, the code was cleaned up and reduced. In the fourth iteration, there was a lot of code that was repeated in each sub, this was no longer the case after the fifth iteration.

5.3.6. The Outcome
The results seen in this chapter is the outcome of the fourth and fifth iteration phases. Because of the major modelling differences seen in the case study, reusing the code from the previous research was very difficult. The entire structure for the Evo Clamp tool insert was built on having parametric models created using surface modelling. Since the structure of the Evo Flush bracket and tool insert templates were completely different, an almost entirely new script was developed. The only parts that could be reused or adapted from the Evo Clamp script was how and where to save the files.

5.3.7. Evo Flush Bracket
To be able to mate the Evo Flush bracket to the template bracket, reference planes were added manually to both the desired bracket, and the template bracket. To mate the component, three planes were added to each of them. Each Evo Flush bracket consist of two parts, a fixed and a variant. The fixed part is never changed when a new variant of the bracket is modelled. Therefore, the new reference planes were added to the fixed part as seen in Figure 17. The template bracket consists only of the fixed part.

![Figure 17 The variant part of an Evo Flush bracket with reference planes.](image)

Furthermore, the process from Figure 13 was followed. The value of the angle was accessed through the parameter name of the angle. The same method was applied to suppress the corners. The bracket before and after the automation was performed can be seen in Figure 18.
5.3.8. The Pad

Since the pad was an imported feature, a reference plane was added to the template file. The plane was used to be able to create a sketch on one side of the pad. Other than that, the pad contained a few parametric entities, which could be used to perform the DA. These entities would always have the same names. Therefore, the names were used when accessing the entities through the VB.NET script.

The automation of the pad was made in six different steps. Each step was connected to a button which triggered a sub in the VB.NET script. The buttons can be seen in Figure 19. Each button was named to match a step in the process of the pad presented in Figure 16. There was also a button that triggered all the above.

In Figure 20, a visualization of what happens in the CAD model when pressing the different buttons is shown.
5.3.9. The Punch

The automatic modelling of the punch was done very similarly to the pad. Here, a sketch plane had to be created in the same way as with the pad, and some entities could be used even if the punch was an imported feature without any parametrical entities.

However, when extruding the sketch in the same way for the punch, the extruded punch sketch covers the holes used when attaching the inserts to the tool. Therefore, a sketch called “Holes” were added to the template, see Figure 21. This sketch was then accessed to clear the holes after the extrusion had been made.

Figure 21 Clarified hole sketch around the green holes.

Much like for the pad, nine buttons were created based on each step for the punch in the process from Figure 16. Furthermore, a button to execute all the above was created. The buttons are shown in Figure 22. Figure 23 shows the different steps executed in the CAD assembly.
To create a new folder that included all necessary files for the project, code could be reused from the research regarding tool inserts for the Evo Clamp manufacturing tool. Only minor changes had to be made to fit the new DA since the complexity of the one for Evo Clamp was much higher. The changes mainly contained reducing the code. The process from Figure 12 was followed. The function *Pack and Go* in Solidworks was necessary to use when saving the
assembly files since this function broke all the links to the original template. When the files were copied, their names were also changed to fit the bracket number. Since the bracket was accessed through the PLM system used at Thule, this was not automated. The final DA required that the user manually retrieved the bracket and opened it in Solidworks before starting the process.

5.4. Descriptive Study II

5.4.1. Final Evaluation

The final evaluation was done based on three previously presented factors: reduced time, robustness, and flexibility. The evaluation based on the factors are here presented on at a time.

The first evaluation factor was the reduced time. Modelling the tool inserts manually, including retrieving the bracket and saving all files in the correct place takes an experienced technician 5-10 minutes depending on the complexity of the bracket. Today, approximately one Evo Flush bracket is created manually every month. The automated modelling did not include retrieving the bracket. However, without that it took less than one minute for the entire process to be executed.

The robustness of the DA was proven to be quite low. The reason behind this was the extended use of parameter and component names. In most functions, parameter names had to be used to access points and sketch segments. Since a few of these names were connected to the variant modelling of the bracket, these could change depending on what bracket was being used. However, when choosing lines from the bracket to create the sketch on the punch and pad for example, these chosen lines were chosen through coordinates instead. Although this was proven successful to improve the robustness, the method could not be used in all cases. Another factor that played a part in the low robustness was the fact that the reference bracket was one of the less complex brackets, which resulted in the automation process being quite simple. The low robustness was also connected to the fact that the instance name of the bracket changed every time the script was run. However, this would not be a problem if the DA was used since the instantiation should only be done once.

As a result of the low robustness, the flexibility was proven to be even lower. The DA only function flawlessly for one bracket. However, there was only minor changes that needed to be done for the DA to work on at least one more bracket. The only required changes were changing the parameter names connected to the variant modelling of the bracket.
6. Discussion

6.1. The Approach
Finding an appropriate way to conclude this project was shown to be a challenge. Several
iterations of the approach were needed until the final one was set. In the end, the DRM
framework best described the approach. Even though the DRM framework was developed to
be used in research and not necessarily in a project like this, it was shown to be highly
suitable. The PDP was an option to describe the process for a long time. However, since the
PDP is more suitable for designing a product rather than a process, it had to be modified a lot
to fit the desired approach. Although, even if the PDP was not necessarily used outspoken, it
influenced the iterations that were performed during the prescriptive study phase. Thereby, the
DRM framework could be seen as the overall approach for the project, while the PDP was a
part of the script development process.

Furthermore, this master thesis did not include any research questions, as can be seen in the
report. It did in the beginning, however there was so much focus on finding a sufficient way
to describe the aim through research questions, that it took over the entire project for a while.
Therefore, removing the research questions and only stating the aim and goal of the project
was a way of moving forward. It ended up giving a clearer view on what the project was
about, and therefore this master thesis project does not have any research questions.

6.2. The Results

6.2.1. Simplifications
As seen when studying the processes from the manual to the automated process, some
simplifications were made. The reason behind the choice to simplify the process was that the
goal of the project was to create a 3D model of the tool inserts using DA. This could be done
without all the manual steps. Because of the simplifications, the process could need further
development to be fully functionable in the organization. However, this does not affect the
results when it comes to concluding the thesis.

6.2.2. Reusing Code
As seen in the results, the structure of the different brackets and tool insert templates were
modelled in two completely different ways. After noticing this during the first descriptive
study, reusing the code from the Evo Clamp DA was not an option when creating a DA of the
Evo Flush tool inserts. An attempt of using the code for inserting and mating the bracket to
the tool inserts template was made. However, this was unsuccessful. Therefore, writing a new
function to execute this rather than finding a way to adapt the old code was seen as more time
effective and thereby a better choice. A lot of the problems regarding reusing the code were
not only because the different modelling strategies, but also the fact that the tool insert
templates were imports without any parametric entities. It was very clear that the templates
and the brackets were not modelled with DA in mind. Having templates more suitable for DA
could possibly have changed the outcome. However, for the reuse and adaption of code to
have the best presumptions, the modelling strategy for both brackets and templates should be
more consistent and follow the same pattern. If that would have been the case during this
project, the outcome would have had the prerequisites to have been entirely different.
6.2.3. Challenges
This project did not come without challenges. The struggle with the templates has previously been highlighted. Anyhow, another challenge was automating something that had previously been done manually with human intuition and creativity. During the interview when the manual process was explained, the technician showed how he, for example, dragged the sketch point together to form a closed sketch. He also said that the corner radius of the punch and the pad varied, and he changed the fillet size from time to time depending on the outcome. As for now, the finished automated process does not leave any room for human intuition and creativity. This could clearly be a problem when using the process for different Evo Flush brackets. Therefore, automating the entire process from start to finish might not be the best way even if it saves time. One could inevitably argue that the process should stay manual, or at least include pop-up questions that require human answers for some of the functions.

Furthermore, a problem appeared when looking at different brackets than the one that was demonstrated. Most of the brackets looked very different and more complex than the one used as a reference. The modelling strategy for both the brackets and tool inserts were not consistent even if the input was just different variations of the same bracket. Since this problem appeared quite late, it was not possible to create a DA that could be used for as many brackets as first wanted. Therefore, it was decided to focus on creating a DA that worked for only the reference bracket, with good presumptions to work for others with just minor changes to the script.

The challenges could most likely have been greatly reduced if there would have been modelling guidelines for the components and templates.

6.2.4. The Evaluation
The evaluation did not show a great success of the DA. However, it did highlight the importance of consistent modelling. With more time, the process could most likely have been refined and the outcome could have been even better. Anyhow, the evaluation was performed quite late in the project and if redoing the project, more brackets should have been tested much earlier in the project. However, since the complexity of most of the brackets was not known until later in the project, this was not considered to be important for the aim of the project.

The evaluation did however show that automating a process would reduce the time drastically. Although, since only about five new tool inserts were modelled manually during the master thesis, one could question if creating a DA process for this case would save any greater time for the technician who performs the manual work. However, DA could be good for other causes, such as eliminating human errors and thereby reducing the lead time in a project even more. It would also give a greater freedom in the organization so that other functions in the team would not have to rely on one person to be available to do the manual work, when anyone could press a button.

This evaluation was an application evaluation, as described in the DRM framework. This means it was based on key factors. Method/tool evaluation was thought of continuously through the project so that the application evaluation could be performed. However, it could possibly have been executed in a better way since testing different brackets were not done.
earlier on in the project for example, and thereby there was a very low chance of the flexibility and robustness factors reaching any greater success.

6.2.5. In a Broader Perspective
The struggles with DA that occurred during this project was not only connected to Thule as a company. To be able to create a functioning DA, having CAD models that are modelled with a DA perspective in mind is important. As seen here, automating models that are barely parametric is not easy and requires some changes to be made before starting. Also, using different modelling strategies for components that could potentially use the same DA script is not optimal. Therefore, having a consistent modelling strategy that is used around the company should be required if the goal is to automate a modelling process. Furthermore, using DA could potentially reduce the lead time, and it would enable different functions in the organization to model, for example, a tool insert. DA would not only let engineers focus on the creative work instead of the repetitive, but also reduce human errors and let anyone press a button instead of having to rely on a small group of people to be available to do the manual modelling.
7. Conclusions

The goal of this project was to investigate how the research regarding Evo Clamp tool inserts could be adapted and reused to work for Evo Flush tool inserts. Going in to the first descriptive study, it was quickly seen that the two different tools were manually modelled in two entirely different ways. Connecting this with the DA script of the Evo Clamp tool inserts, the conclusion was clear. Using or adapting the DA script from the Evo Clamp tool inserts was not an option when it came to the parts that were geometry related. However, saving the files and creating folders was still the same and could be reused. Therefore, the conclusion from this project is that, if the intention is to use DA to automate different processes, this should be kept in mind when modelling the templates. If the templates follow the same pattern, reusing the script from one process to another could be possible with only minor changes. However, reusing and adapting a script developed for one process without having another process in mind is not appropriate. Developing a new script for the second process is more time effective. To have a consistent modelling framework could potentially increase the robustness and flexibility of the DA as well as reducing necessary script modifications to get the process to work. This leads to the conclusion of the second aim in this project, to assess the effect of the DA.

The conclusion regarding the effects from the outcome is that seen to anything but the reduced time, the effect is low. The cause of this is as stated before the lack of consistent modelling and parameter names. Since the DA does not have to be backwards compatible, modifying the modelling strategies to better fit an automated process could in theory be done. This would increase the potential of creating a more effective DA. However, as stated in the discussion, all the human intuition and creativity is removed when creating an automated process and this is something that needs to be considered if continuing to implement an automated process in this specific case. When creating modelling guidelines, the creativity would also be reduced. Therefore, it is important to consider if it is worth it or not. One could argue that a completely automated process that removes all creativity would not be beneficial in this case.
8. Further Research

To further investigate the theory and results from this project, a way to go would be to start with building a product platform with processes for similar components or tool inserts that were manually modelled following the same pattern, focusing on modularity. Creating guidelines for the modelling would also be beneficial.

Another approach, which is more focused towards the company, would be to integrate the results from this project in a PLM (Product-Lifecycle Management) system to be able to easily access and maintain the tool inserts. Another appropriate step for the company to take would be to run the script through Howtovation Suite for a more visual representation.
9. Bibliography


d24-be7f-41c5-af66-12c18fc631d9%7CPolo%2520%7CHatchback%2520%28M


I. Appendix

Appendix I
A summary of the manual process for creating tool inserts for a Evo Flush manufacturing tool as described during an interview with a technician at Thule.

PREPARATIONS

- Go to v-arkiv → 1100-1199→V1188→VARIANTER
- Copy tool template (VAR-XXX MALL)
- Name it the desired variant number (VAR-XXX) in a folder in the VAR-fold
- Copy link to "Detalj" where all files from YaPlm will be saved
- Retrieve bracket from YaPlm
- Right-click the file
- "Export files..."
- "Export kit" (PDF drawings part)
- Change "export to folder" to the copied link to "Detalj"-folder

- Go to the folder in "Detalj"
- Copy the CAD-file of the bracket and rename it to end with "FORM YYYYMMDD"
- Open the CAD-file
In CAD

- Variant folder → Sketch 30 → Edit
- Change the dimensions from this...

To this
overbend by 2° (-2°)

Make driven and add a arcline to the radius.

- Set offset to 0.01mm and delete the relations between the old line and the one with an offset.
- Set the offseted line as "Fix" and make it a construction line (This is done to know which one is the original line).
- Suppress the corner radius.
• Save

• Now the bracket - file is ready to model a tool from

• Open the tool - template through SolidWorks (from the "VAR-XXX" folder created in the beginning)

• The file is named something with "Backstation"

• Check the references when opening, they should be to the "VAR-XXX" folder, if not - change them to be

• Since only the A-station will be used, delete the rest

• Rename the assembly in the tree to correct "VAR-number" and delete "mill" and "B", do the same for all parts in the assembly
• Insert component and choose the edited bracket, place it on the bracket template (mate them together by three mates)

• Now we can modify the tool.

• Start with the pad (right-click -> edit part)

• Sketch -> choose the bottom lines of the bracket and the top lines of the tool

• Push the lines together to make a sketch which could be closed and extruded to fit the gap between the tool and the bracket
• Extrude boss up to surface to the other side.

• Pad is DONE

• Move on to the punch

• Edit part → Sketch on one side

• Choose all top lines of the bracket and pick convert, also choose two lines of the punch
- Connect the lines of the punch and the bracket by moving the entities.

- Isolate punch, the new extrude covers the hole/s.

- Make sketch & cut to clear.

- Add radius.