Design support for metallic machined ribs using knowledge based engineering techniques

T. van der Laan, F. van Dalen and B. Vermeulen
ton.vanderlaan@fokker.com
Stork Fokker AESP BV
Papendrecht
Holland

ABSTRACT

In recent years increasingly pressure has been applied on aircraft component suppliers to reduce design lead-time and design cost of aircraft components. To achieve this reduction in lead-time and cost, advanced automation tools that automate part of the engineering process can be used. Knowledge based engineering (KBE) tools are one such automation tool type and automate part of the engineering process based on existing knowledge within a company. In this paper the development process and use of a KBE application to design machined ribs in an industrial setting is discussed. The development of the KBE tool has resulted in the standardisation of the design methodology. Furthermore it has reduced machined rib design lead-time in a project to develop a bussines jet empennage by 40%.

1.0 INTRODUCTION

In recent years increasingly pressure has been applied on aircraft component suppliers to reduce design lead-time and design cost of aircraft components. To achieve this reduction in lead-time and cost, advanced automation tools that automate part of the engineering process can be used. Developing these automation tools can be a costly though. Therefore it is imperative that the developed automation tools can be used multiple times within a single project and/or can be used in multiple projects. To achieve this the automation tools should be aimed at often recurring structural components of which the engineering process is highly standardised.

Stork Fokker AESP (SFA) develops and produces advanced complex lightweight structures for the aviation and aerospace industry. It excels in the design and production of box-type structures, e.g. movables, empennages, and shell-type structures, such as fuselage sections. SFA has two engineering and production facilities in the Netherlands, and one engineering office in Bucharest, Romania. It supplies lightweight aircraft components and systems to leading European and American aircraft builders in both the civil and defence sectors, amongst which are Airbus, Boeing, Cessna, Dassault, Eurocopter, Gulfstream, Lockheed Martin, Raytheon and others.

In the box type structures developed by SFA, ribs are often occurring elements. Multiple design concepts exist for these ribs. The design concept is often related to the manufacturing method used to produce the rib and the assembly method used to assemble the rib in the aircraft component. Rib design concepts are also approximately similar between projects. Therefore automating part of the engineering process of such a design concept could be used to achieve the lead-time and cost reductions required by the market.

In this paper the process of developing an automation tool for the design of machined ribs is discussed. Ribs of this design concept are often occurring in the box type structures engineered and manufactured by SFA. One methodology for developing an automation tool is the knowledge based engineering (KBE) methodology. With this methodology knowledge is captured from the experts in the development process and this knowledge is used to create automation tools.

Paper No. 3419. Manuscript received 17 March 2009, revised version received 20 July 2009, accepted 4 August 2010.

https://doi.org/10.1017/S0001924000004164 Published online by Cambridge University Press

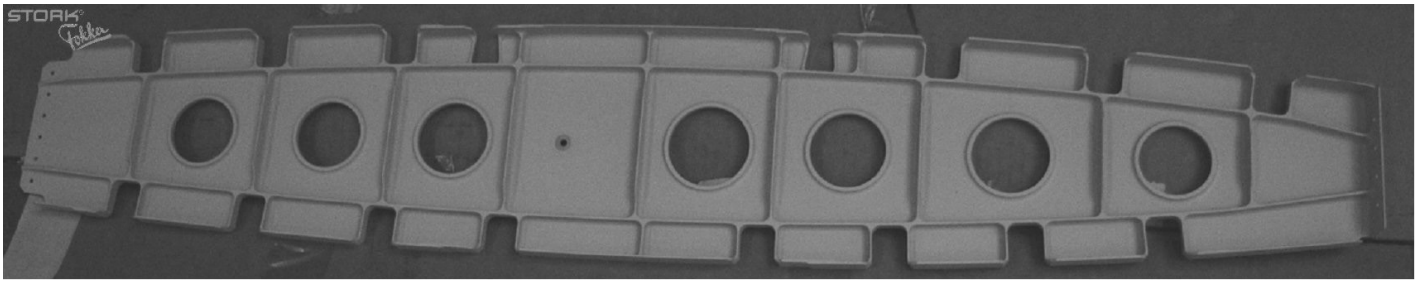


Figure 1. Machined rib example.

The goal of the developed tool is to reduce design lead-time and to standardise the design methodology for the machined rib design concept. The tool was developed within a project of developing the empennage for a large business jet. Therefore this project will be used as an example throughout this paper. The tool development was executed in part by a specialised software development firm in this case Dassault Systems (DS). The tool is used at multiple sites within SFA in fact the first users of the tool were located in the SFA engineering office in Bucharest.

2.0 MACHINED RIB DESCRIPTION

A machine rib can be found in most box-type structures. These box-type structures can be aircraft wings, parts of the empennage or aircraft movables. In all these structures the shape and function of a machined rib is approximately the same. The machined rib is used to support the box skins in order to ensure that the box retains its outer shape when loaded. Furthermore the ribs can be used to introduce loads in the box structure. These loads can originate from actuators, hinges or other elements that are part of the box construction.

Figure 1 shows an example of a machined rib, this example shows the often-occurring elements of a machined rib. One element is the interface of the rib to the skin. For this interface flanges are incorporated in the rib with which it is fastened to the skin. Another element is the interface of the rib to its adjacent spars. The solution used for this interface is dependent on the loads transferred between rib and spar and also on the shape and design concept of the spar. What can also be seen in Fig. 1 are the stiffening and supporting elements perpendicular to the rib plane. These elements provide stiffness to the rib and are also used as load paths to transfer the loads within the rib. Finally cut-outs are made in the rib. In Fig. 1 two types of cut-outs can be recognised; stringer mouse holes and system or lightening holes. The stringer mouse holes allow the stringer to pass through the rib. The system or lightening holes allow systems such as hydraulic lines or control cables to pass through the rib or are used to reduce the weight of the rib.

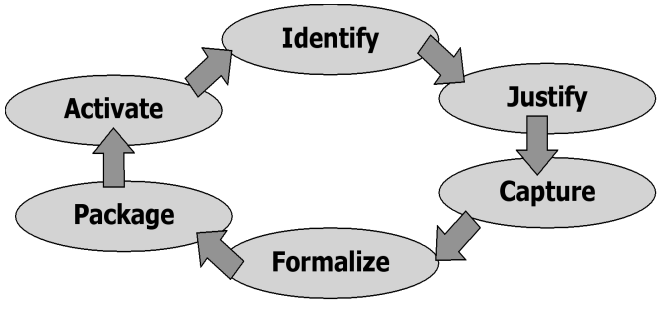


Figure 2. The MOKA cycle for the development of a KBE tool.

The machined rib design tool must be able to model all the features mentioned above that are generic throughout machined ribs. Some elements that are not very generic, such as a rib spar connection, will have to be designed manually by the designer.

3.0 DESCRIPTION OF THE MOKA METHODOLOGY USED FOR DEVELOPING KBE TOOLS

The development process of KBE tools described in this paper will follow the MOKA methodology[1]. This methodology describes the whole life cycle of a KBE tool as shown in Fig. 2. The KBE lifecycle is highly iterative, which means that the steps are not always executed in the sequence shown. For example 'proof-of-concept' tools are sometimes activated during the capture phase to provide valuable feedback.

Identify: Using an existing business opportunity, the potential KBE applications and the resources needed to build them are identified.

Justify: For each potential KBE application, a go-ahead decision is made on the basis of expected benefits, development cost and available resources.

Capture: The knowledge needed for the KBE applications is gathered and structured in order to get a knowledge base for the KBE application. This knowledge base uses simple models, understandable by engineering domain experts.

Formalise: The captured knowledge is remodelled in the so-called formal format, which is easily understandable by the software engineers developing the KBE application.

Package: The software that makes up the actual KBE application is written based on the formal model created in the previous phase.

Activate: The software tools developed in the previous phase are put to work in daily practice.

In the sections below the contents of these steps will be described for the development of KBE tool for the design of machined ribs.

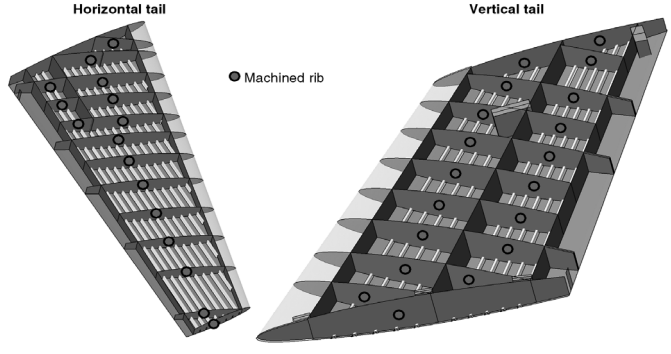


Figure 3. A schematic model of a horizontal and vertical tail of a large bussines jet indicating the locations of machined ribs.

https://doi.org/10.1017/S0001924000004164 Published online by Cambridge University Press

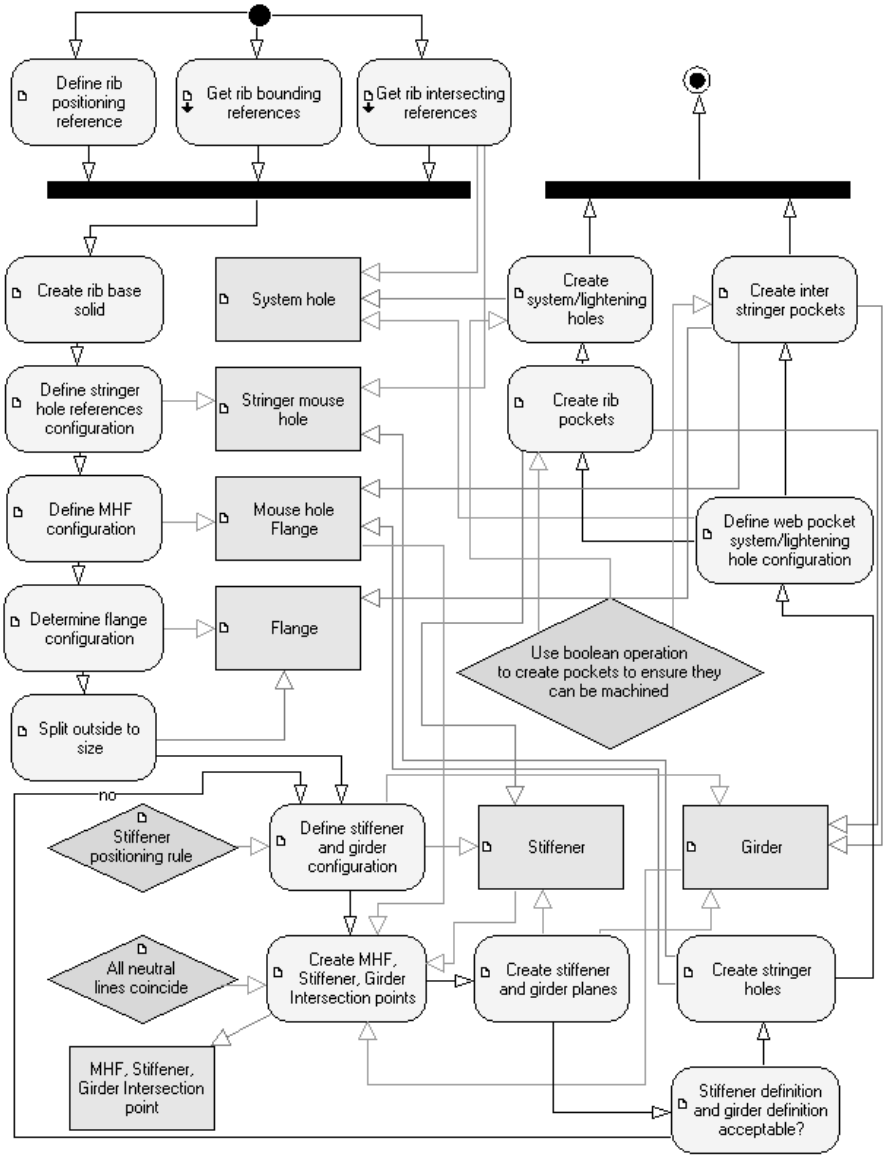


Figure 4. The captured machined rib design process knowledge.

4.0 IDENTIFYING THE OPPORTUNITIES FOR DEVELOPING KBE TOOLS AND JUSTIFYING THIS DEVELOPMENT

At the start of any KBE project the elements that will be incorporated in the project need to be identified and the use of the resources required for such a project need to be justified. Before deciding to automate the design of machined ribs such an identification effort was undertaken for a project in which the empennage, meaning the horizontal and vertical tails including movables, of a large business jet are developed.

A KBE application can automate certain steps in the development process. However knowledge needs to be available about these steps. Developing a KBE application also costs a considerable amount of resources. For the KBE application to become a success it has to be used frequently. Therefore the concept designs of the horizontal and vertical tails were evaluated for often occurring parts. As can be seen in Fig. 3 the vertical and horizontal tail design concepts contain many machined ribs. In total approximately 35 machined ribs are encountered. Therefore, automating part of the development process of these ribs was identified as an opportunity for automation using the KBE methodology.

The development process of a machined rib contains three major elements: designing the rib itself, structurally analyzing the rib to ensure it remains intact during operation, the so-called stress process, and developing the NC program needed to manufacture the rib. These three elements of the development process were identified as candidates for automation. This paper will focus on the rib design elements of the development process.

To justify the development of KBE applications the return on investment of these tools needs to be estimated. In this project the applications had to become profitable within the project of developing the empennage for a large business jet. To determine the return on investment the resources needed to develop the KBE application and the resources saved by such an application need to be estimated. Such an estimation can be based on previous development projects of both the ribs themselves and of other KBE tools. In Table 1 the normalised estimates used in the justification phase are shown. Assumptions for this justification are that 35 ribs will be modelled using the KBE tool and that the design lead-time reduction achieved by using the KBE tool is 60%.

https://doi.org/10.1017/S0001924000004164 Published online by Cambridge University Press

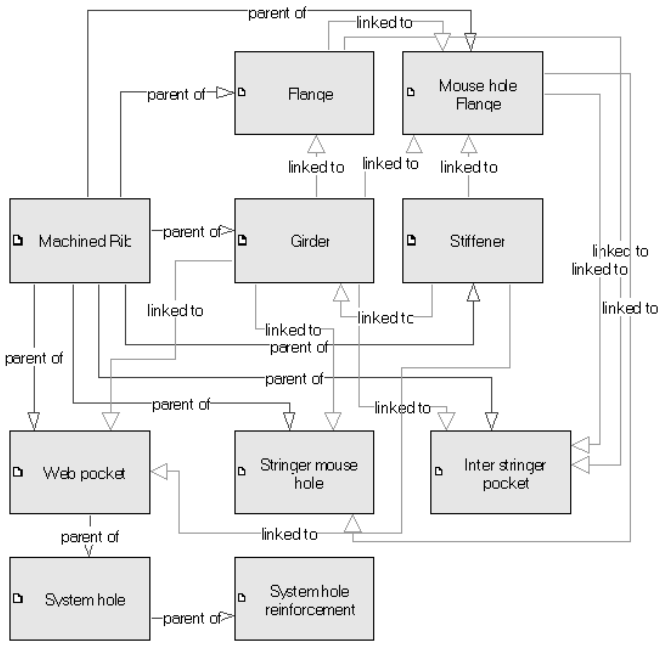


Figure 5. Machined rib structure diagram.

Table 1
Normalised KBE tool justification table with normalised values for the estimated hours

	Rib design
Regular design 1 unit	1
Total time	35
Automated design 1 unit	0·4
Total time	14
KBE application development	9
Return on investment	1·52

As can be seen the expected return on investment for rib design is greater than one justifying the development of a KBE tool for these activities.

5.0 CAPTURING THE DESIGN KNOWLEDGE AND FORMALISING THIS KNOWLEDGE FOR TRANSFER TO OTHER PARTIES

In the capture phase the knowledge needed for developing the KBE tool is captured by, for instance, interviews with experts. To use this knowledge in a KBE tool it has to be formalised. This means representing the knowledge rules in a manner, which is understandable for the software engineer developing the KBE tool. The capture and formalise steps are highly iterative and also interact with the next step in the KBE life cycle the package step. What this means in practice and how the capturing and formalisation phases for the machined rib design tool were executed will be discussed in this section.

To capture the knowledge needed for a KBE tool one first has to determine if this knowledge is available. Part of this assessment has already been done in the Justification phase when an inventory is made of the knowledge sources available. In case of the machined rib design tool the knowledge sources available consisted of previous designs of similar ribs and of experts experienced in designing these ribs. However, the design philosophy used for the machined rib in the development project of the empennage for a large business jet differed somewhat from previous machined ribs. Therefore some of the design knowledge had to be developed from scratch.

For the machined rib design tool previous rib designs were analyzed and design experts were consulted to determine the knowledge rules governing the design of a rib. This knowledge was stored in a knowledge base. Using specialised software e.g. PCPack[2], knowledge diagrams were created. In addition to these diagrams test designs were created. These test designs proved especially helpful to identify areas where there was a lack of knowledge, to develop new knowledge and also to identify where the captured knowledge rules contradicted each other. Using the test designs and the knowledge diagrams the knowledge sources; in this case usually experts could be consulted to fill the identified knowledge voids. A diagram of the rib design process used for consulting the knowledge experts is shown in Fig. 4. In this diagram the yellow boxes represent tasks performed in the rib design process. The green diamonds represent rules that must be applied in these tasks. Finally the blue boxes represent the elements that are output of, or changed by, the tasks.

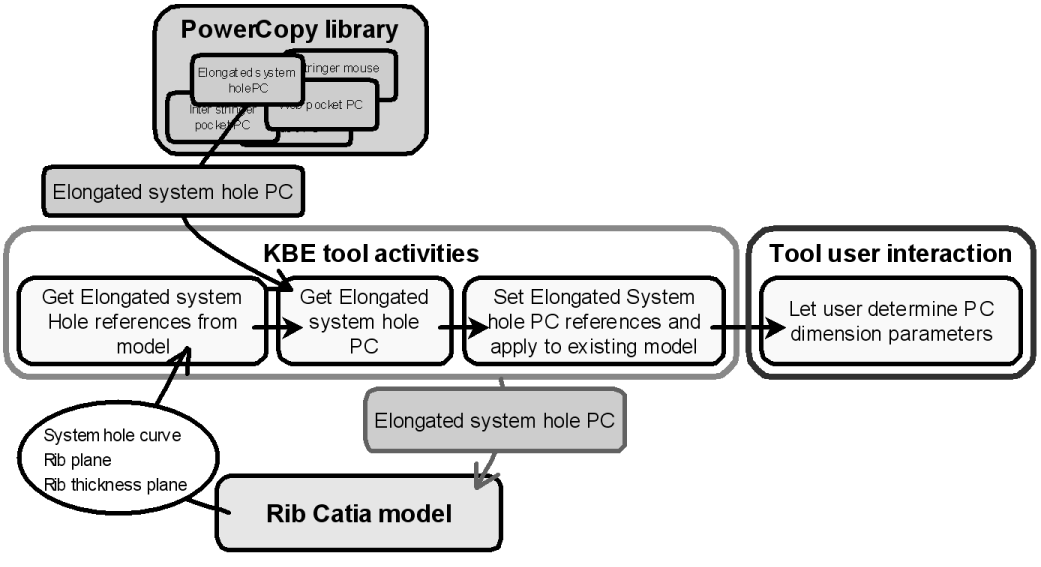


Figure 6. The process of applying a PowerCopy to an existing Catia model of a partially built machined rib part.

https://doi.org/10.1017/S0001924000004164 Published online by Cambridge University Press

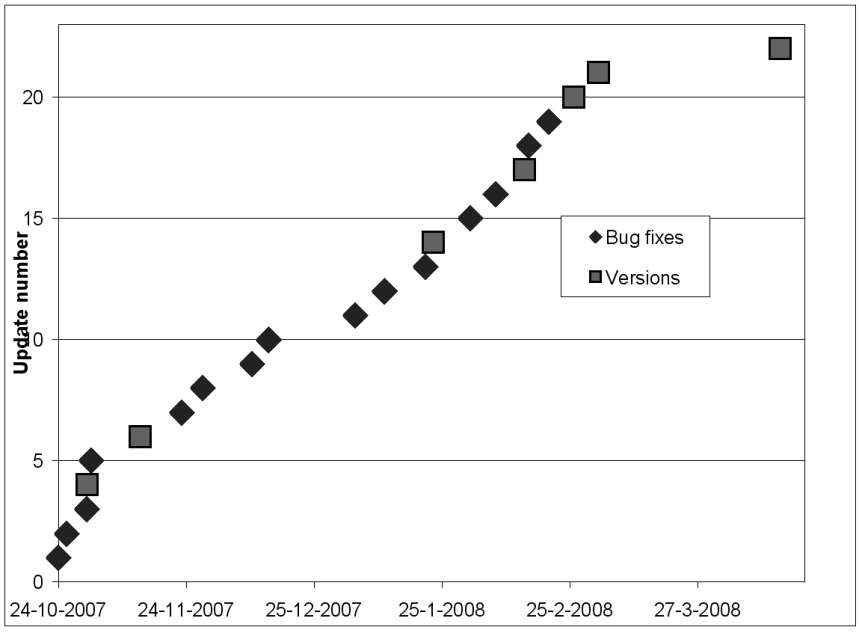


Figure 7. History of updates of the tool after first release, squares are version updates with changing functionality while diamonds are bug fixes where functionality remains the same.

Besides capturing the rib design process the structure of the rib itself was also captured. This means that the different structural features in the rib and the relations between the different elements were identified. This proved essential in the knowledge and tool development process, because the machined rib philosophy applied resulted in more links between features than traditionally was the case. By having a diagram available showing these links experts could judge if rib design philosophy would produce ribs that met the requirements. The structure diagram of the machined rib is shown in Fig. 5.

The knowledge captured by the knowledge engineer at SFA had to transferred to the actual tool developer at DS. According to the MOKA methodology transferring the knowledge requires formalising the knowledge. This means translating the informal model, as for example shown in Fig. 4, to a more formal description of the software required. However because the informal knowledge model already had the form of flow diagrams this model was used to transfer the knowledge to the software engineer. To make this transition run smoothly multiple sessions between knowledge engineer and software engineer were required. It was essential that these sessions were face-to-face and demo tools and informal knowledge base were available. In this way un-clarities could easily be explained. These sessions took place at SFA. This had the added advantage that Knowledge experts were at hand to explain any un-clarities the knowledge engineer could not explain to the software engineer.

In a sense the formalisation process described by MOKA was executed in the face-to-face sessions, because the software engineer added comments to the informal knowledge model that she required to translate it into software code. Furthermore the objects required for the software were identified and formalised by the software engineer. The objects required were however software, in this case Catia V5 with a VBA interface, dependent. This meant that all the formalisation steps were software specific. The actual formal model used for the development of the software was not captured in the knowledge base, because the knowledge engineer at SFA controlled this knowledge base and the software engineer at DS had no permission to change or amend it. This lack of knowledge base update will have to be avoided in the future projects to ensure that all knowledge about the KBE tool and the KBE tool development is stored in it. Furthermore the formalisation step needs to be made software independent to ensure that the tool can be easily rebuilt in other software packages.

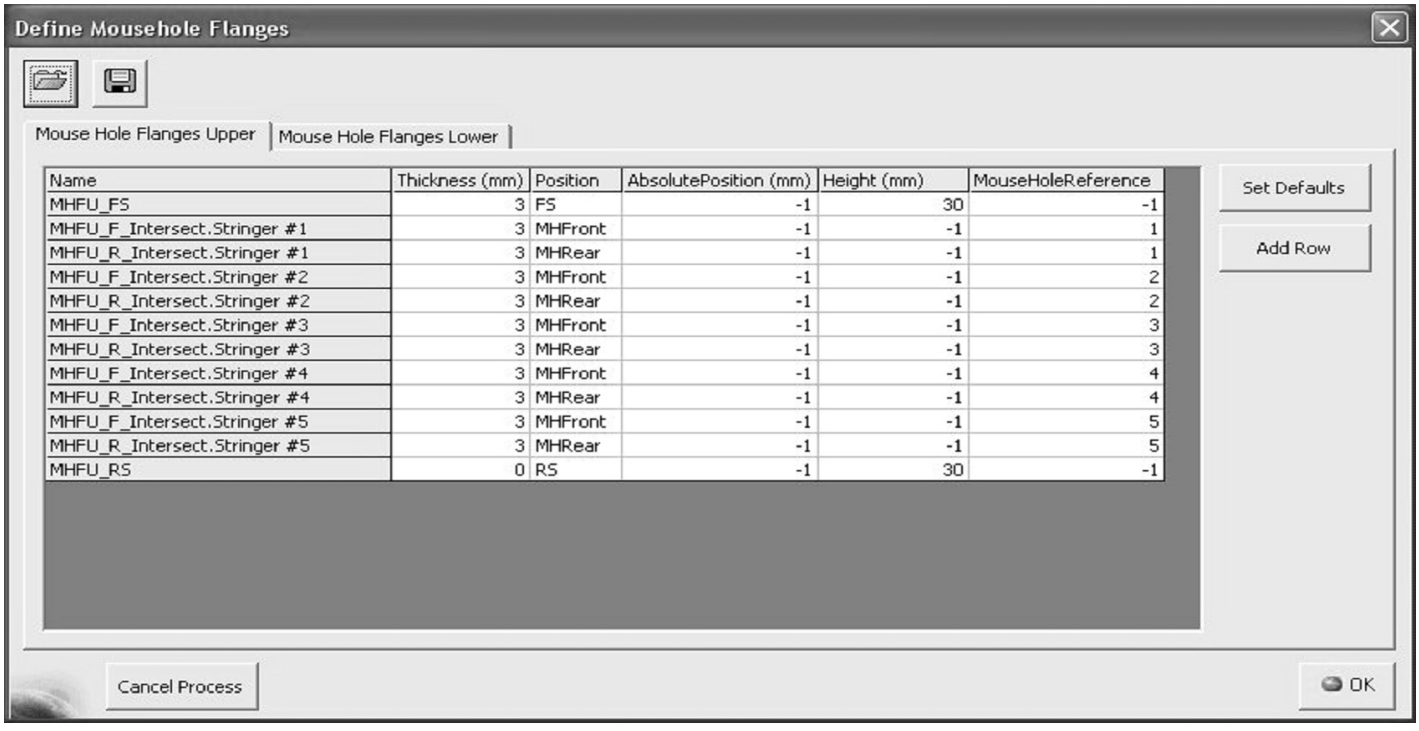
6.0 PACKAGING THE ACQUIRED KNOWLEDGE IN KBE TOOLS AND ACTIVATING THEM FOR USE IN EVERYDAY PRACTICE

The next step of the KBE cycle is the packaging step. 'Packaging' means: using the acquired knowledge from the previous step to create the actual KBE software tool. This software tool is put into action in the Activate step. Both the packaging and the activate steps will be discussed in this section.

Before developing the KBE application software first the software platform on which it will be created has to be chosen. In this case the machined rib KBE tool uses the visual basic interface of Catia V5. Catia V5 was chosen because it is the CAD software package used in the project to develop the empennage of a large business jet. The visual basic interface was chosen because it allows relatively simple programming of the software modules, while still allowing the programmer to use all the modelling options necessary. It also provides satisfactory performance. Programming code was used only where necessary, in other words where the standard functionality of Catia did not suffice. What this means is that the software mostly controls the combination of features pre-modelled in Catia and changes the parameters controlling the dimensions of these features. The pre-modelled features are so-called PowerCopies, which model a number of CAD elements based on a set of input geometry and parameters. In the applications the tool user, through user interface forms, sets these parameters values. Default values, based on the input from experts, are given for these parameters. The diagram in Fig. 6 shows how a PowerCopy is used.

DS conducted the actual packaging of the tool itself. This packaging consisted of two main tasks, developing the PowerCopies and developing the software code that assembles all the PowerCopies into a rib design. Because the input and output for each PowerCopy is fixed, development of each of them can be handled as a different task. In this way different PowerCopies could be created in parallel by different people, reducing the tool development lead-time. The development of the software code that assembles all the PowerCopies and development of the accompanying graphical user interface (GUI) proved to be the most time consuming and was performed by one person, who also coordinated the PowerCopy development.

https://doi.org/10.1017/S0001924000004164 Published online by Cambridge University Press



Name	Thickness (mm)	Position	AbsolutePosition (mm)	Height (mm)	MouseHoleReference
MHFU_FS	3	FS	-1	30	-1
MHFU_F_Intersect.Stringer #1	3	MHFront	-1	-1	1
MHFU_R_Intersect.Stringer #1	3	MHRear	-1	-1	1
MHFU_F_Intersect.Stringer #2	3	MHFront	-1	-1	2
MHFU_R_Intersect.Stringer #2	3	MHRear	-1	-1	2
MHFU_F_Intersect.Stringer #3	3	MHFront	-1	-1	3
MHFU_R_Intersect.Stringer #3	3	MHRear	-1	-1	3
MHFU_F_Intersect.Stringer #4	3	MHFront	-1	-1	4
MHFU_R_Intersect.Stringer #4	3	MHRear	-1	-1	4
MHFU_F_Intersect.Stringer #5	3	MHFront	-1	-1	5
MHFU_R_Intersect.Stringer #5	3	MHRear	-1	-1	5
MHFU_RS	0	RS	-1	30	-1

Figure 8. Example of a form used by the rib designers to adjust rib parameters.

As was already discussed in the previous section regular sessions between SFA and DS were held in which the latest test version of the tool was discussed. In these discussions feedback on the tool functionality and also the look and feel of the tool, in other words the GUI, was given to DS. In this way the finalised tool delivered by DS met the specification based on the knowledge base and the number of issues encountered during the actual activation of the tool was minimised.

After the packaging phase conducted by DS, the tool had to be activated at SFA. This activation process had different steps and were executed by the knowledge engineer at SFA:

1. ***Testing the delivered tool and adjusting where necessary***

   The tool delivered was developed according to the requirements specified in the knowledge base and based on the previously discussed feedback sessions. However the functionality required and therefore the tool requirements were constantly changing resulting in changes in the tool. Furthermore testing of the complete design space to be covered by the tool revealed areas where the tool was unstable. These instabilities had to be solved. Supplying the software developer with a better, more detailed, test specification should reduce the amount of work in this step in future projects.

2. ***Training the tool users***

   Tool users are trained by interactive sessions. This basically means that the tool user and tool development engineer from SFA sit together and built a rib with the tool together. This step is repeated each time there is a new user for the tool.

3. ***Roll out of the tool so users could access it***

   During the development of the tool it was kept in a test environment. This means that only development engineers could access it. To use the tool in daily practice it had to be made available to the tool users. The tools were made available by packaging the tool into a neat set of files and placing this package on a network drive.

4. ***Provide support to users and adjust functionality of the tool where necessary***

   Once users started using the KBE tools to create ribs there was a lot of request for enhanced functionality and also for bug fixes. Creating new functionality or solving bugs requires developing and rolling out a new version of the tool.

After first release of the tool at SFA several updates have been created. Two types of updates can be distinguished; version updates and bug fixes. With version updates the functionality of the tool changes, with bug fixes on the other hand functionality remains the same. In Fig. 7 the update history of the tool after first release can be seen. As can be seen at first there are many updates required to fix the bugs found by the users and also to add functionality required by the users. In this period it might have been better to just have a beta version of the tool in use that could be updated easier then real released versions of the tool, which require some form of version management. After this initial period the number of updates remains steady and largely limited to bug fixes. This changes from 25-01-08 onward, then an increase in update frequency can be seen and also the functionality of the tool changes frequently. In this period a new project, in this case the design of the horizontal stabiliser, started use of the tool. As a result the functionality of the tool had to be changed. This might have been avoided by going into a new beta version phase with this project and combining all the enhanced functionality into one new version of the tool.

7.0 THE RESULTING MACHINED RIB KBE TOOL

The machined rib tool was intended to automate 60% of the design work. This means that after using the tool the designer still has to finish the rib design by adding details such as the connection of the rib to a spar. The designer uses the tool by running the application an interacting with it when necessary. This interaction consists mostly of defining the design parameters using a GUI. An example of a form used to adjust the rib parameters can be seen in Fig. 8. The parameters determine the size of the features in the machined rib. The tool has a built in function to use parameters from a previously recorded configuration file. In this way previously stored configurations can be re-used. The complete tool activity diagram including the interaction of the user with the tool can be seen in Fig. 9.

https://doi.org/10.1017/S0001924000004164 Published online by Cambridge University Press

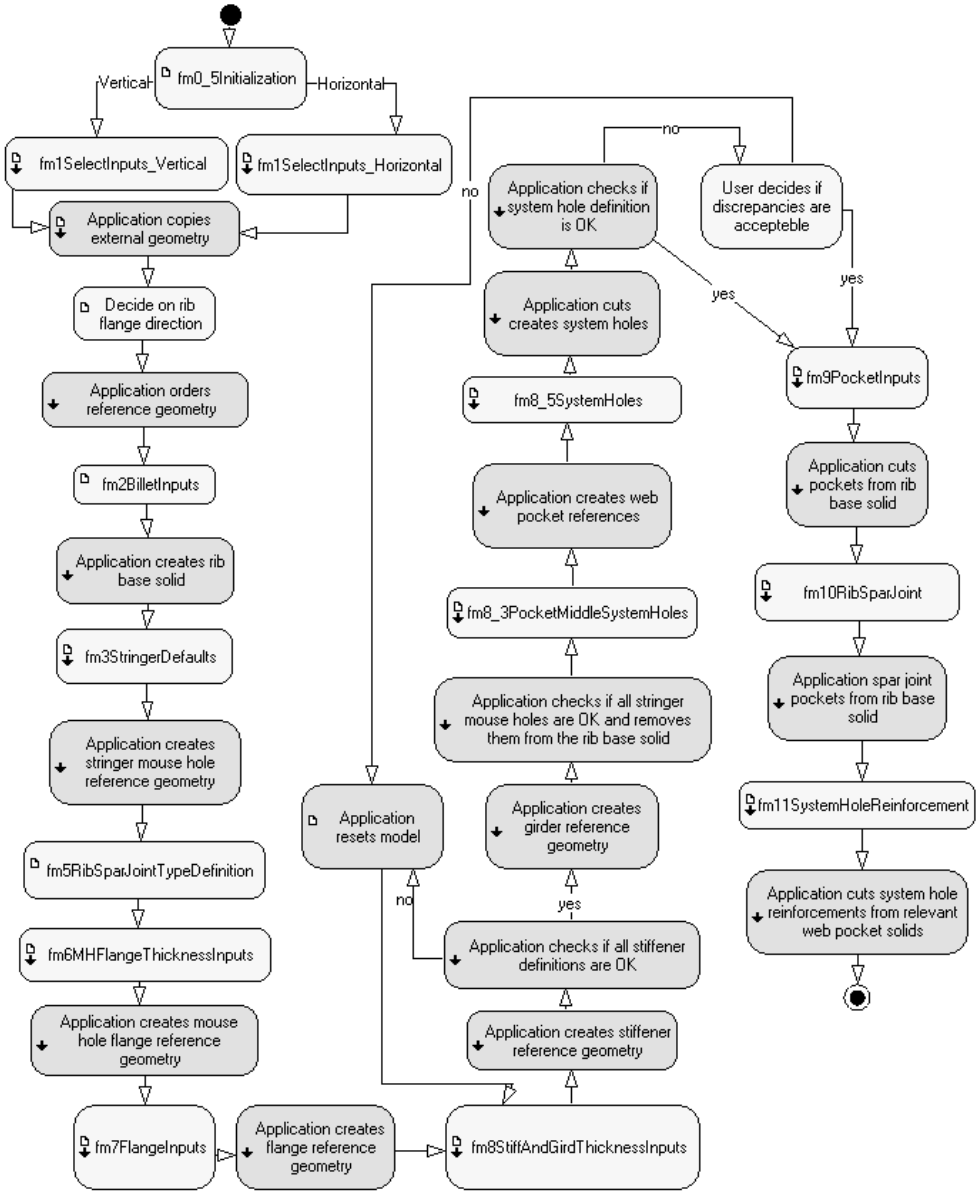


Figure 9. A process diagram of the most recent version of the tool. Lightly coloured activities require human interaction; darker coloured actions are performed automatically.

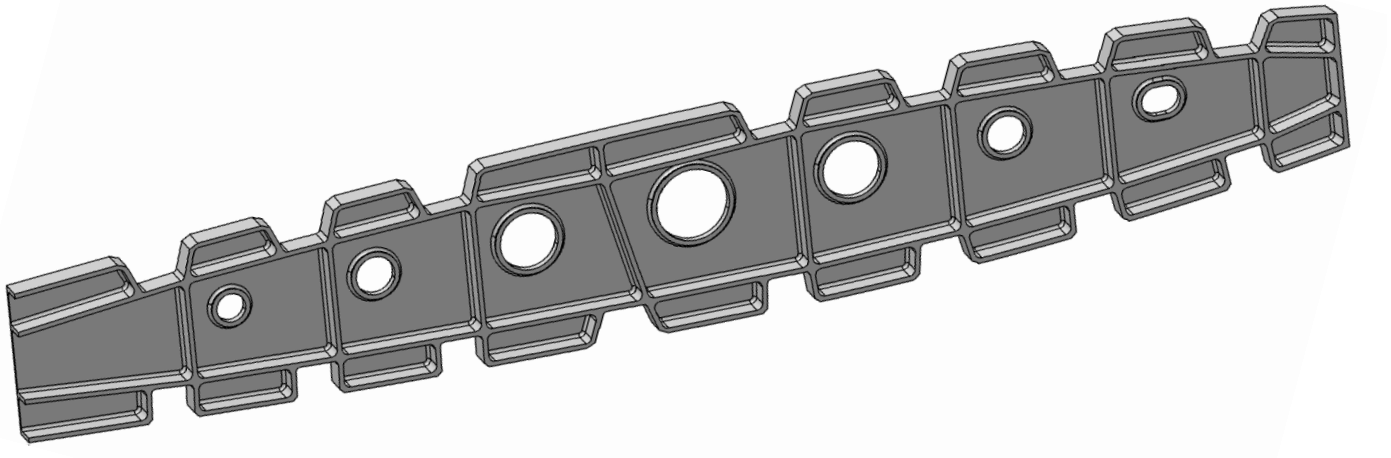


Figure 10. Model of a machined rib created with the machined rib tool.

https://doi.org/10.1017/S0001924000004164 Published online by Cambridge University Press

The machined rib tool stores the design methods and best practices used by SFA to model machined ribs. When designers use the tool they are guaranteed to use these methods and best practices. The knowledge base accompanying the machined rib KBE tool also stores these methods and best practices separate from the machined rib design tool. This allows designers to consult these methods and best practices even when not using the tool. In the aircraft component development program in which the tool was used these elements proved very important to transfer the knowledge about these methods and best practices to the SFA engineering office in Romania, which was the first user of the tool. This presented some challenges but most of them were quickly overcome. First of all roll out required that the tool developer was on site in Romania for one week to ensure proper operation of the tool and to train the users of the tool. After this regular e-mail and telephone contact proved sufficient to enable the use of the tool in Romania and also to gather the feedback from the users in Romania for tool updates. In this process it proved essential to have a single point of contact on both sides. This meant that there was one user in Romania and one developer in Holland. Having one point of contact made sure that there was an understanding on what was required as feedback and also helped in getting new versions of the tool rolled out and up and running.

The machined rib model created by the tool is built using the subtraction methodology. This means that one starts with a base solid and subtracts elements from this solid to reach the final result. Big advantage of this method is that, when used properly, the machine-ability of the model is guaranteed. In case of the machined rib KBE tool the base solid is created by using references such as the upper and lower skin contours and spar planes. The element subtracted from this solid consist of pocket, stringer mouse holes and system holes. A rib resulting from the machined rib KBE tool can be seen in Fig. 10.

The machined rib model created with the tool is a normal Catia model. This means that there are no features in it that make it different for a model built without a KBE tool. Although a normal Catia model there are some features that make it stand out form models created interactively by designers. First of all it is fully parametric, this means that all parameters that determine sizes or thicknesses in the model are stored as parameters. These parameters are stored in parameter sets located throughout the Catia feature tree. This has in fact been one of the biggest difficulties using the machined rib KBE tool. Designers are not used to changing a design using parameters and are also not used to having different parameter sets throughout the Catia feature tree. This problem has become particularly apparent when transferring design responsibility of a Catia rib model from one designer to another. To solve this tables have been added to the tool manual informing them what parameters of a feature can be changed and where to find them. This solution has however not solved all problems. In future additional time and resources will have to be reserved for training the users. This training has to consist how to use the KBE tool and will also have to make them familiar Catia features often used by the tool. An example of such a feature is the extensive use of parameters.

The lead-time reductions of 40% over the initially estimated design times have been achieved by using the machined rib KBE tool. Although not as high as anticipated there are indications that the initial time estimates might have been too low, which would mean that the actual lead time reductions are bigger then 40%. One reason for not achieving the time reduction envisaged was that the finishing of a machined rib generated by the KBE tool took longer than expected. In other words, more details than anticipated needed to be finished by hand.

Not all 35 machined ribs recognised in Fig. 3 were designed using the KBE design tool largely because of the problems with transferring models between designers mentioned before. Therefore the return of investment has not been as high as anticipated in the justification phase. However when the tool is used in future projects its development will certainly become profitable.

The less than anticipated design lead-time reduction and difficulties transferring models between designers can largely be attributed to the complexity of the rib design itself and the rib design process. This proved to be a problem especially because the experience of SPA developing KBE design applications was limited. Better results have been achieved with automating the design of simpler products[3].

8.0 CONCLUSION AND FUTURE DEVELOPMENT OPPORTUNITIES

The development and use of the machined rib KBE tool has reduced lead-time and standardised the design methodology. Lead time reductions of 40% with respect to estimated design times without the use of a KBE tool have been achieved. Design methodology is standardised because it is embedded in the tool.

Embedding the design methodology in the tool also means the model build up of each rib is the same. Something that helps processes further on in the development process such as NC programming and stress analysis.

Using a KBE tool proved very useful when transferring design methodologies between different company locations. A methodology developed by SFA Holland was first used and fine tuned by SFA Romania.

The rib design process has been formalised and stored in a knowledge base. This knowledge base provides valuable information about this process and also allows the knowledge about this process to be applied in future projects. This knowledge application can be with or without the use of the machined rib KBE tool because the knowledge rules are explicitly stored in the knowledge base.

Because the structure of the machined rib model is now fixed it can be used as a solid basis for the NC programming. This programming process could in theory also easily be automated. This is one of the future developments anticipated for the machined rib KBE tool.

The machined rib KBE tool was developed specifically for one aircraft component development program. Now that this program is almost finished the lessons learned and knowledge developed in this program can be used to further develop the modelling capabilities of the machined rib KBE tool. Future aircraft component development programs will surely benefit from the development of this tool.

REFERENCES

1. MOKA consortium, Managing engineering knowledge, Professional Engineering Publishing Limited, London, UK, 2001
2. http://www.epistemics.co.uk/
3. LAAN, T. VAN DER, VERMEULEN, B., DALEN, F. VAN Automating the development process of thermoplastic composite sheet ribs using knowledge based engineering techniques, 26th International Congress of the Aeronautical Sciences, Anchorage, Alaska, US, 14-19 September 2008.

BIOGRAPHY

Ton van der Laan received a MSc. from Delft University of Technology in 2001 in the design of composite structures. After which he specialised in research focussing on Knowledge Based Engineering and Cost Engineering. He has published multiple papers in these research areas. In 2008 he received a PhD. from Delft University of Technology for research focusing on the previously mentioned research areas. Currently he is working as a Knowledge Engineer at Stork Fokker AESP implementing multiple Knowledge Based Engineering applications.

https://doi.org/10.1017/S0001924000004164 Published online by Cambridge University Press