# Application of a knowledge based design methodology to support fuselage panel design

**S. W. G. van der Elst and M. J. L. van Tooren**
**S.W.G.vanderElst@tudelft.nl**
Delft University of Technology
Delft, The Netherlands

**B. Vermeulen**
Fokker Aerostructures
Papendrecht
The Netherlands

**C. L. Emberey and N. R. Milton**
Tacit Connexions
Wokingham, UK

## ABSTRACT

Knowledge is a vital component of engineering design. Computer systems enriched with logic and engineering knowledge can support engineering design by automating repetitive and time-consuming processes. This automation is enabled using knowledge based engineering (KBE) techniques and can be obtained using dedicated KBE systems or augmented CAD systems, already pervasive throughout engineering industry. The development of these KBE applications is supported by a six-phase development process ranging from engineering process analysis to software development to business implementation. Distinctions and similarities exploiting alternative KBE platforms are addressed for each phase of the development process. An example KBE application is discussed, supporting the design of laminate aircraft fuselage panels. The implementation of the application is emphasised and five key-aspects required for a successful implementation are defined.

## NOMENCLATURE

| | |
|---|---|
| AI | artificial Intelligence |
| API | application programming Interface |
| CAD | computer aided design |
| COTS | commercial of-the-shelf |
| FML | fibre metal laminate |
| GLARE | GLAss REinforced aluminium |
| KA | knowledge acquisition |
| KBE | knowledge based engineering |
| KBS | knowledge based system |
| KE | knowledge engineering |
| KM | knowledge management |
| MDO | multidisciplinary design and optimisation |
| MOKA | methodology and tools oriented to knowledge based engineering applications |
| prepreg | unidirectional glass fibres pre-impregnated with epoxy resin |
| VB | visual basic |

## 1.0 INTRODUCTION

Modern-day aerospace market dynamics require an increasing focus on lifetime cost reduction, combined with a shorter time-to-market. Moreover, the design of aerospace systems is becoming increasingly complex, and fewer engineers with appropriate educational level and experience are available[1,2]. In order to keep meeting the stringent customer demands, the aerospace industry needs more cutting edge productivity enhancements.

A methodology that supports this shift in productivity enhancement is knowledge engineering (KE), providing a number of methods and tools that guides the process of acquiring, structuring and reusing engineering knowledge. Knowledge is a vital component of engineering design and significant reductions in costs and product development time can be realised if engineering knowledge would

**Paper No. 3383**. Manuscript received 1 December 2008, revised version received 19 March 2010, 2nd revised version received and accepted 3 June 2010.

https://doi.org/10.1017/S0001924000004061 Published online by Cambridge University Press

be reused to a larger extent and more often. Where previously the geometric model took a central position, today design knowledge should have the focus: knowledge should be managed and engineered as a key business asset[3].

Knowledge based engineering (KBE) techniques enable computer systems to be enriched with logic and expert knowledge in order to support engineering by automating repetitive and time-consuming design processes. The reuse of knowledge decreases the engineering resources required and relieves engineers from recurring and tedious activities, making more time available to exploit their creativity and engineering skills[4,5].

In the previous articles of the authors several KE technologies have been discussed to support the development of KBE applications[6]. An example was given of the application of knowledge acquisition methods in combination with a knowledge management tool. The result was an extensive knowledge base, supporting the development of a KBE application. Furthermore, a KE process is defined to guide the development of KBE applications using dedicated KBE systems[7].

This article focuses on the development and implementation of KBE applications using computer aided design (CAD) systems with built-in KBE functionality. Since the early days of KBE specific software packages, CAD systems have been augmented to incorporate engineering knowledge to facilitate automated rule-based modelling of geometry, e.g. Dassault Systemes' CATIA products and Unigraphics by Siemens. These augmented CAD systems act as budget alternatives to dedicated KBE systems, since many engineering companies increasingly exploit CAD systems as general drafting and modelling tools and are restricted in their purchase and use of other commercial of-the-shelf (COTS) software.

Similarities and fundamental differences during the development of KBE applications between augmented CAD systems and dedicated KBE systems will be addressed. Key success factors related to the deployment and implementation of applications will be identified. A practical example of a KBE application will be discussed, supporting the design of fibre-metal laminate sheets in aircraft fuselage skin panels using CATIA v5.

First the subject of managing engineering knowledge is covered in Section 2. Second, the definition and fundamentals of knowledge based engineering are addressed in Section 3. Section 4 discusses the knowledge engineering process applied to augmented CAD-systems, followed by an example KBE application using CATIA v5 in Section 5. Finally, the conclusions are presented in Section 6.

## 2.0 KNOWLEDGE MANAGEMENT

Knowledge management (KM) addresses the use of techniques and tools to make better use of the intellectual assets in an organisation. KM concerns the identification of vital corporate knowledge and capturing, representing and storing this knowledge from domain experts and existing repositories for understanding and reuse[8]. KM is usually involved in lessons-learned, innovations and process improvements. In order to describe the approach to knowledge management more clearly, the concepts of knowledge and knowledge acquisition will be explained.

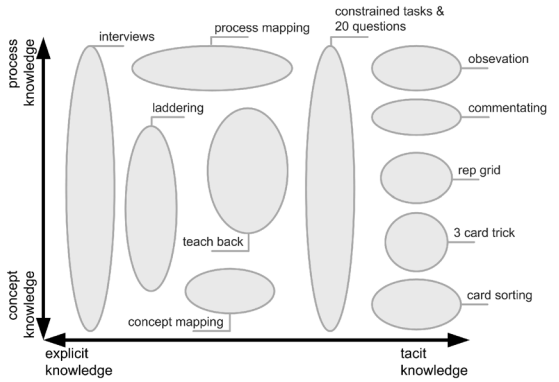



Figure 1. Knowledge acquisition matrix[6].

### 2.1 Knowledge

Knowledge can be considered a dynamic system, driving a process that takes data and information as input in order to support decisions or perform actions. This results in the following definition[8]:

Knowledge is the {ability, skill, expertise} to {manipulate, transform, create} {data, information, ideas} to {perform skilfully, make decisions, solve problems}

There are various classifications of knowledge. With respect to engineering two important dimensions to describe knowledge are:
(*i*) Procedural knowledge versus conceptual knowledge;
(*ii*) Basic, explicit knowledge versus deep, tacit knowledge.

Procedural knowledge concerns processes, tasks and activities. It describes the conditions, under which specific tasks are performed, the order in which tasks are performed and the resources required to perform tasks. Conceptual knowledge concerns the description of concepts and their relation to other concepts. Hence, it addresses the ways in which objects are related to one another and their properties. An important form of conceptual knowledge concerns taxonomies, i.e. classes and class membership. Another type of conceptual knowledge addresses attributes of concepts.

Basic, explicit knowledge is the type of knowledge residing at the forefront of a specialist's mind and is thought about in a deliberate and conscious way. It is concerned with basic tasks a domain expert performs, basic relationships between concepts, and basic properties of concepts. Deep, tacit knowledge is at the other extreme to basic, explicit knowledge. It is knowledge residing at one's subconscious and often leads to automatic decision making or actions that seem to require no conscious thought. It is also referred to as experience, art or craft[6].

The corporate advantages of captured and stored knowledge are numerous:

- Disseminate knowledge to other people within an organisation to provide expertise
- Reduce the risk of knowledge loss in domains where only a small number of experts hold vital knowledge
- Reuse knowledge to enrich computer systems to perform tasks normally performed by human domain experts

### 2.2 Knowledge acquisition

Knowledge acquisition (KA) covers identifying, capturing and storing knowledge from humans and existing repositories in order to enable understanding and reuse of knowledge.

Despite the potential benefits of capturing and reusing knowledge, it has long been recognised that knowledge is hard to acquire from domain experts. The difficulties stem from a number of factors. First, domain experts are not good at recalling and explaining everything they know. The tacit knowledge which operates at a subconscious level is hard, if not impossible, to explain. Second, domain experts have different experiences and opinions that require aggregating to provide a single coherent picture. Third, domain experts develop particular conceptualisations and mental shortcuts that are not easy to communicate. Fourth, domain experts use jargon and assume most other people understand the terminology being used.

https://doi.org/10.1017/S0001924000004061 Published online by Cambridge University Press

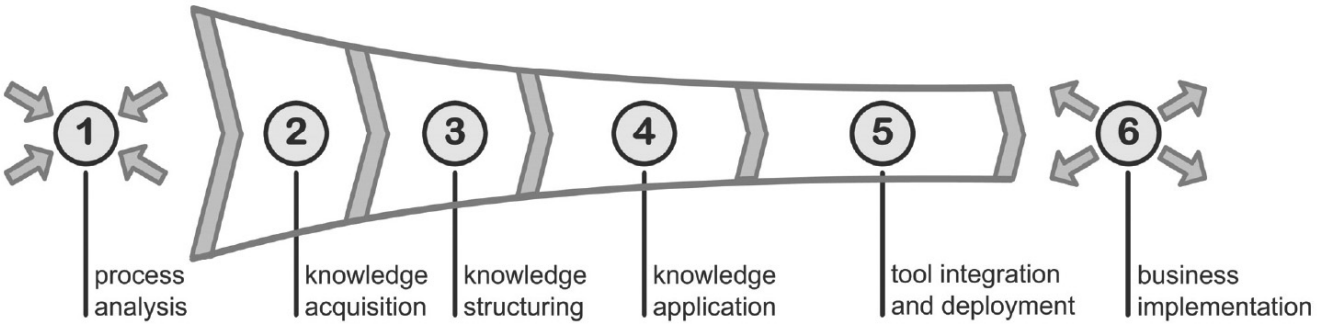



Figure 2. Knowledge engineering process for the development of KBE applications.

To deal with such difficulties a number of techniques and tools have been developed that considerably improve the process of acquiring knowledge. A diversity of knowledge acquisition tools is presented in the knowledge acquisition matrix, illustrated in Fig. 1. The knowledge acquisition matrix provides several tools in order to acquire the different types of knowledge.

Dedicated software tools make the process of acquiring, storing and representing knowledge more efficient and less prone to errors. PCPACK[9] is the comprehensive of such tools. It is used for creating a knowledge base, a special database storing organisational knowledge and information representing the expertise of a particular domain[8]. To represent the expertise efficiently, the structure of the knowledge base is identical to the structures that underlie human expertise. Psychologists have found that this is based on four main components; concepts, attributes, values and relations.

## 3.0 KNOWLEDGE BASED ENGINEERING

According to Milton[8] KBE concerns the development of software systems to support engineers, usually design engineers, to increase their productivity. KBE systems are able to capture and reuse product and process knowledge to solve engineering problems involving the manipulation of geometry, product configuration and analyses and computations. The main objective of KBE is reducing time and cost of product development by means of the following:

- Automation of repetitive and non-creative engineering tasks
- Support of multidisciplinary design and optimisation (MDO) during the conceptual phase of the design process

The KBE cornerstones are rule-based design, object-oriented modelling, and parametric CAD[5]. KBE systems are considered a subset of Knowledge Based Systems (KBS) empowered with CAD capabilities. KBSs are based on methods and techniques from Artificial Intelligence (AI) which in turn aims at creating intelligent entities[10]. KBE found its first application as follow-up of CAD to enable designers to reuse models. KBE differs from CAD systems that are concerned with a single, mainly geometric model. KBE address the design of a family of products and therefore contains generative models representing the rationale and knowledge behind the design.

### 3.1 Knowledge based engineering platforms

Three types of platforms can be identified that enable the development of KBE applications: dedicated KBE systems, proprietary KBE system and augmented CAD systems.

Cooper[11] defines that dedicated KBE systems require five important features: functional coding style, declarative coding style, runtime value caching and dependency tracking, dynamic data types, and automatic memory management. The object-oriented perspective that engineers usually have is supported through an object-oriented modelling approach. These dedicated KBE systems are built on an open-ended and standards-based programming language that enables the definition of modifications, extensions and higher-level abstractions for specific domains. Examples of dedicated KBE systems include the former ICAD by KTI and GDL by Genworks.

Contrary to dedicated KBE systems following the textbook definition, two fashionable alternatives exist: augmented CAD systems, and proprietary KBE systems.

Proprietary KBE systems are developed using languages that are not standards-based and only accessible by the vendor. Consequently, development of design applications cannot be conducted independently from the vendor and customisation or extensions of KBE functionality may involve additional expenses. The accessibility of the proprietary systems is also inconsistent with the ability to easily communicate with other open-source or standards-based software tools widely adopted in the field of MDO. A contemporary example of a proprietary KBE system is AML by TechnoSoft.

Concerning CAD systems, several vendors now claim to provide built-in KBE functionality, such as KnowledgeWare for CATIA v5 and KnowledgeFusion for Unigraphics. However, the augmented CAD systems have some major deficiencies regarding performance and modelling capabilities. First an entire CAD system must be running as well as the KBE augmentation in order to access the computer interpretable knowledge. This results in significant computing overhead, during development as well as during run-time. Second, they don't provide full generative modelling. Central to any CAD system is the geometric model, compared to a generative knowledge model embedded in dedicated KBE systems. The KBE functionality associated with CAD systems merely enables the application of pre-defined scenarios or scripts in order to perform rule-based manipulation of the geometric model. On the other hand, the benefit of developing design automation applications using these contemporary CAD systems is that most engineering intensive industries are already familiar with the CAD tools. Even more so, no supplementary costs are involved since no additional software is required[11].

Due to the low financial threshold and their widespread exploitation throughout industry, the augmented CAD systems are considered to be the most suitable alternative to dedicated standards-based KBE systems. Therefore, in the following section, the development of KBE applications will be considered mainly with respect to dedicated KBE systems and augmented CAD systems.

## 4.0 KNOWLEDGE ENGINEERING PROCESS

Knowledge engineering (KE) is the engineering discipline integrating KM and KBE techniques by developing software applications that conduct engineering processes normally requiring human expertise. It addresses the management of corporate knowledge and the creation of design applications that automate repetitive and time-consuming processes by reusing this knowledge.

https://doi.org/10.1017/S0001924000004061 Published online by Cambridge University Press

Although huge benefits can be gained, knowledge engineering and KBE techniques are still to become mainstream technology in industry. KBE used to belong to the promising tools developed by research and development departments of deep-pocketed engineering companies and academia alike. Expensive software licenses, a lack of computing power and programming languages geared towards expert developers acted as barriers to the widespread adoption of KBE technology in industry[11]. However, independent software vendors entering the market with affordable KBE software suites and the technological developments in the IT industry have now enabled KBE to become mainstream technology.

To reduce the investments and risks involved in the development of KBE applications several methodologies have been established, such as CommonKADS[12] or the methodology and tools oriented to knowledge based engineering applications (MOKA)[13].

Whereas the different methodologies focus on different aspects of the development process, they incorporate much of the same principles from the fields of knowledge management and KBE. For example the MOKA methodology focuses specifically on structuring and formalising the associated knowledge[13]. The established methodologies have been analysed in order to define a comprehensive process that better suits the object-oriented paradigm and modular build-up of applications developed using dedicated KBE platforms[7,14]. The result is a renewed KE process, aimed to further reduce the cost, time and risks of the development of KBE applications while supporting the reuse of separate software modules to generate novel applications. The process consists of six subsequent phases as shown in Fig. 2.

### 4.1 Process analysis

The first phase of the KBE lifecycle concerns an engineering process analysis, aimed to identify process improvement opportunities by applying lean principles. The process analysis is mainly focused on knowledge-intensive processes and products belonging to a larger set or family, ensuring a large applicability of the design automation tool. The deliverable of the process analysis is a value map, addressing the process' bottlenecks and key improvement opportunities using KBE techniques.

During the analysis of the engineering design process the flow of information and required expert knowledge is monitored. The analysis focuses on three main characteristics:

- Required engineering resources
- Repetitiveness of engineering process within product family
- Nature and maturity of expert knowledge

The required engineering resources and the number of process cycles provide insight in the total investment and effort involved in different recurring processes for a family of products. It should also offer information about the longevity of the applied knowledge. The domain expert knowledge is assessed to determine its nature and maturity. When processes are highly frequent, time-intensive, clearly defined and not likely to change in the near future, KBE techniques can enable automation. During the process analysis, the possible software architectures, i.e. KBE platforms and other required software packages such as Knowledge Management tools are examined. For repetitive engineering tasks that address the creation or manipulation of geometric models and heavily rely on virtual inspection an augmented CAD system is an appropriate choice. For other engineering tasks dedicated KBE systems seem more suitable, since they allow the incorporation of non-geometric knowledge more easily and can do without the heavy overhead of a full-blown user-interface.

Together the risks and a cost-benefit analysis should justify the implementation of KBE techniques. The analysis phase is concluded by a down selection of engineering processes to be automated.

### 4.2 Knowledge acquisition

During the knowledge acquisition phase expert knowledge involved in the engineering process is captured and structured. The knowledge acquisition phase has an iterative character and consists of structuring, capturing and validating the expert knowledge. The deliverable of the knowledge acquisition phase is a knowledge base, a digital repository containing a detailed description of knowledge concerned with the selected engineering process.

The quality of the captured knowledge largely determines the success rate of the development process hence the resulting KBE application. To guarantee a successful result, the acquisition process is performed in close cooperation with the domain experts. The involvement of domain experts is vital to the project for two main reasons:

- Identification and dissemination of relevant knowledge
- Validation of quality and completeness of the captured knowledge

In order to ensure the ability for future reuse of the captured knowledge, it is recommended that the knowledge base is not catered to a specific implementation (augmented CAD systems or KBE systems) and embeds a neutral structure suitable for all forms of knowledge reuse.

### 4.3 Knowledge structuring

The third phase focuses on modelling the captured knowledge. The captured engineering knowledge is analysed and (re)structured to suit the development of the application, taking into account the exploited type of KBE system. The deliverable is a revised engineering process, providing a structure and lay-out for the KBE application. It is considered a formal model and should provide a stricter, platform-oriented definition of the engineering knowledge used to support the communication between knowledge engineers and software developers. In other words the so-called formal model provides a software perception of the product and process and acts as a blueprint for the KBE application. Concerning the automation of engineering design processes using augmented CAD systems the formal model will centre on the process knowledge. The model represents a set of procedures and routines to be performed by the CAD system and determines the sequence of procedures and possible paths of execution or scenarios.

With respect to dedicated KBE systems, the formal model of the knowledge base provides an object-oriented structure describing the hierarchical built-up of the central product model. In this case, the emphasis will be on the product knowledge. Execution paths for the application will be self-determined during runtime.

### 4.4 Knowledge application

The fourth phase, knowledge application, addresses the development of the software involved in the KBE application. Following a functional decomposition the software is divided into several separate modules to increase the reuse and the expressiveness of the software code. The deliverable is a set of software modules providing full capability to execute the engineering process under consideration.

The development of the software modules is considered an iterative process. During the development of the application, additional or undiscovered knowledge might be identified and the associated models from the knowledge base need adjustment to ensure that they accurately represent the structure and process of the application.

Developing applications using dedicated KBE systems requires the programming of a central product model. Using object-oriented and high-level programming languages, the resulting code volume is considered very low. Furthermore, programming languages with a higher level of abstraction require lower entry-level programming skills. Utilising augmented CAD systems, not all modelling functions might be directly available through the built-in application

https://doi.org/10.1017/S0001924000004061 Published online by Cambridge University Press

programming interface (API). Therefore work-arounds are not uncommon resulting in high volume, ambiguous and inexpressive software code: maintenance, extensions and reuse of components requires more effort and higher level programming skills.

### 4.5 Tool integration and deployment

The fifth phase addresses the integration of the software modules to form the KBE application. It includes the development of communication interfaces and the distribution of the application itself. The deliverable is an automated design application based on the KBE techniques offering engineering services.

For augmented CAD systems the set of procedures and routines performed by the built-in KBE functionality are connected. The sequence of procedures is fixed and the number of possible execution scenarios finite, implying predictive behaviour according to a fixed script as defined in the formal model of the knowledge base. Intermediate results are validated after each subroutine. Interfaces for both human interaction and coupling of external tools, for example input from data bases or optimisers are developed. However, to establish communication with other software tools expert software development is required using the API of the CAD system. Since execution of the application requires a running CAD system, they are usually deployed directly onto stand-alone desktop systems. In contrast, the integration of software tools using dedicated KBE systems is fully service-based enabled through loose coupling of software agents and enables distributed web-based deployment[15].

### 4.6 Business implementation

The last phase concerns the implementation of the KBE application in the design process. Since the flow of information within a business process will change when deploying KBE applications, a re-evaluation of the design process is needed to prevent the occurrence of new bottlenecks[16]. Configuration management and maintenance are conducted to ensure traceability of the knowledge rules invoked and reproducibility of the resulting solutions. Furthermore, an essential step in the implementation of KBE techniques is to recognise that KBE implies an important change in the work of engineers. Therefore, more practical attributes to a successful implementation are support and training of end-users. Overall, five groups of key success factors for the implementation of KBE applications can be identified as depicted in Fig. 3[17]:

*can*

- Provide training in the operation of the application
- Provide a useful and usable user manual
- Stimulate users to share best practices in using the application

*want*

- Focus on topics important to the business and engineers
- Communicate KBE vision, need for the business, results and experiences of users
- Evaluate the usefulness and usability of the application on a regular basis

*have*

- Plan the development of the application in terms of required resources and release date
- Make reservation in project planning to practice using the application
- Provide support during the lifetime of the application

*must*

- Convince management of possible payback in terms of lead-time and resources

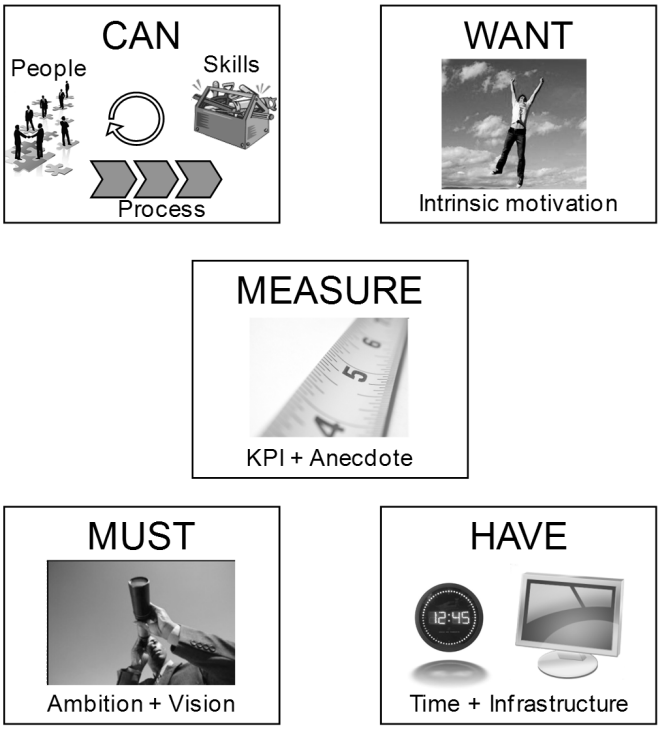



Figure 3. Five key-factors for a successful implementation of KBE applications.

- Have a well-respected engineer promote the use of the application

*measure*

- Develop a measuring system to evaluate application performance

After the implementation an evaluation based of the abovementioned success factors is performed. The next section will discuss the development of a KBE application for fuselage panel design and provides an evaluation of the above-mentioned success-factors.

## 5.0 KNOWLEDGE ENGINEERING APPLIED IN AEROSPACE

The increased significance of the weight and life-time cost of aircraft have stimulated the feasibility and development of hybrid laminates as a replacement for aluminium in skin panels for the civil aerospace sector. At Stork/Fokker AESP, The Netherlands,



Figure 4. GLARE fuselage panels on the Airbus A380. (courtesy of Fokker Aerostructures)

https://doi.org/10.1017/S0001924000004061 Published online by Cambridge University Press

one of the core competences has become the design and manufacturing of Fibre Metal Laminate (FML) fuselage skin panels for the civil aerospace sector. A FML consisting of alternating layers of aluminium and unidirectional glass fibres pre-impregnated with epoxy resin ('prepreg') is GLARE or Glass Reinforced aluminium[18,19]. GLARE is developed for the skin panels of the front and aft fuselage section of the Airbus A380, offering improved damage tolerance results and significant advantages in weight reduction. Some of the largest panels measure 10 by 3·5 metres as depicted in Fig. 4.

Despite the benefits, hybrid laminates also increase the product complexity, since not merely the structure but also the composition of the material needs to be designed[18]. The laminate parameters can be locally tailored to achieve an optimal structure. The constraints to this process are various, e.g. manufacturing constraints limit the dimensions of the aluminium and prepreg sheets applied in the FML.

### 5.1 Prepreg ply division process

Within the FML engineering process, various opportunities for the application of KBE are identified. The most beneficial application is the automation of the sub-division process of the glass fibre sheets or prepreg plies. Considering the complexity and the dimensions of the hybrid laminate fuselage skin panels, manufacturing of the skin panels requires division of the glass fibre sheets into separate strips in the engineering model, see Fig. 5. This sub-division of prepreg plies is a formalised and complex process, since the prepreg strip edges are exposed to a large number of constraints, such as 'restricted' areas (splice areas, see Fig. 6) and a maximum strip width. Using trial and error, engineers divide the prepreg plies into separate strips one by one. However, each strip generates additional constraints for the subsequent plies and strips, increasing the complexity. The prepreg ply sub-division can be classified as a constraint satisfaction problem, since the main concern is to find a solution satisfying all constraints.

The sub-division process of the prepreg plies accounts for approximately 20% of the total time required to prepare the engineering model for production. Due to its time-consuming, repetitive and deterministic nature the conventional design process can be improved considerably applying KBE techniques.

### 5.2 KBE application development

The KBE application has been developed using CATIA v5 and the Visual Basic (VB) programming language. A tailored algorithm has been developed in order to solve the constraint satisfaction problem. A mathematical representation of the panel is generated and all feasible ply divisions are examined and ranked according to minimum cutting waste. A set number of solutions is captured and presented to the user for selection using the CATIA v5 modelling environment.

In general a KBE application based on CATIA v5 is composed of the concepts and relations shown in Fig. 7. As described in the previous paper of the authors[6], much effort is put in making knowledge explicit and formal, resulting in an expert knowledge base. This knowledge base describes the product components and engineering process steps, hence forms the basis for the application. The application is composed of a set of geometric CATIA primitives and VB software code. The product view is composed of the geometric primitives as imported via the input file, and is instantiated by the software code. The engineering process steps are represented by different procedures and routines in the software code, providing a series of geometry manipulations in order to generate the required result. Since the geometric primitives are mostly imported via the input file, the emphasis during programming is on the process steps hence procedures. The application is developed iteratively: after each procedure the outcome is validated before the next step is integrated.

To facilitate the training of users, a default input set is created, consisting of CATIA geometry and input data in the form of a configuration file. This configuration file contains all data that is needed to execute the application and is displayed in Fig. 8.

Furthermore, a user manual is developed to instruct the user on how to operate the application. The user manual describes what input is required and how to present it, how to launch the application, the steps in the execution of the application and the corresponding user interface. It is possible to run the application without manually having to enter any input data, by using the default input set. To that purpose the configuration file is selected each time input is required. Accordingly the user can become more familiar with the application in terms of process steps and in- and output. Finally, best practices regarding the operation can be captured and shared using the best practice database linked to the application.

KBE applications can be operated according to different philosophies. Applying a black-box approach, the application automatically

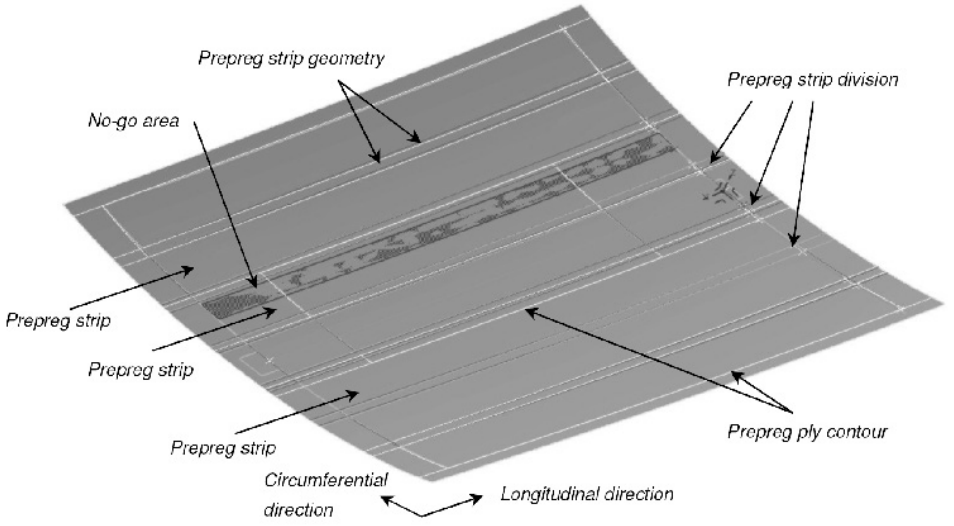



Figure 5. Division of prepreg plies into strips to support the manufacturing of FML fuselage panels.

https://doi.org/10.1017/S0001924000004061 Published online by Cambridge University Press

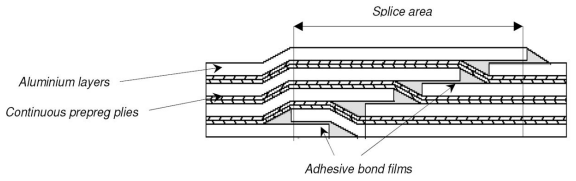



Figure 6. Splice areas connect different aluminium sheets to form one continuous panelmanufacturing of FML fuselage panels.

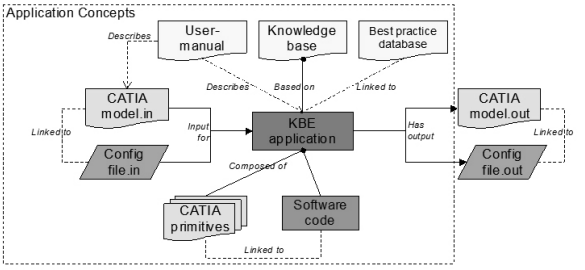



Figure 7. Concepts and relations of a KBE application.

generates all output based on a given input set. If the output is not satisfactory the user has to change the initial input, and re-run the application. A second approach is to guide the user through the process step by step, by asking to select reference geometry and manually enter input data, see Fig. 9.

After each process step the user can see the (midterm) result. This latter approach is selected for the development of KBE applications at Stork Fokker AESP. An important reason for doing so is that the user controls the path of execution and how to proceed, based on intermediate results. Creative process steps can thus be incorporated in the application process flow. Secondly, if the application should terminate prematurely, the user knows at what point in the process this occurs and can decide to continue manually or change input for this particular step in a new run. Finally, it is anticipated for reasons of acceptance that the user wants to understand what the application is doing 'under the bonnet', how input is processed to generate output.

### 5.3 Implementation

The need for proper training in the use of the application is identified upfront. Well before the start of the actual use of the application, a training session has been planned (can). The concepts and relations in the application were explained. Using the input set and process steps described in the user manual, the use of the application was demonstrated. Next the user could practice using the application, based on a 'real life' problem. Although the project involving the design of laminate fuselage panels is executed in CATIA v4, the KBE application is developed using CATIA v5. Therefore, in addition to training in the use of the KBE application, the user is also trained in CATIA v5.

After the application had been deployed and the training was completed, the user or operator of the application was re-assigned to work on a different project, since the start of the original project had been postponed. Although a frequent occurring event in project-oriented organisations, it implies a substantial waste of cost and time since re-training of the same or a new user was inevitable. As the original project restarted, time had become limited, up to a point that the user proposed to conduct the process manually rather than invest in (re)-learning how to use the application. Management had to free time for the user (have) and stimulate the use of the application (must).

### 5.4 Evaluation

Given the implementation of the KBE enabled prepreg division process, an evaluation can be made based on the five groups of success factors as described in Section 4.6.

*Can*

Training in the use of the application is identified as a value adding activity. Nevertheless, given the shift in priority of work, the training had to be redone, resulting in waste. The user manual proved to be

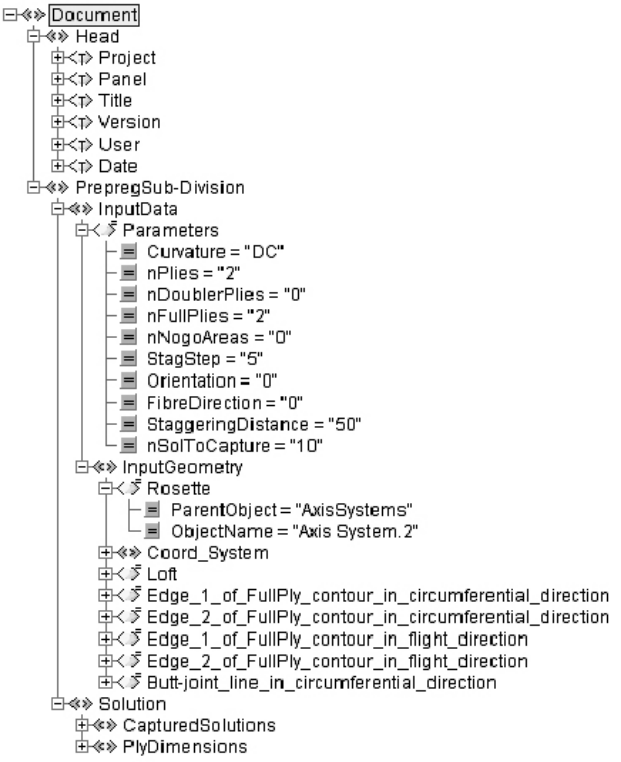



Figure 8. Configuration file.

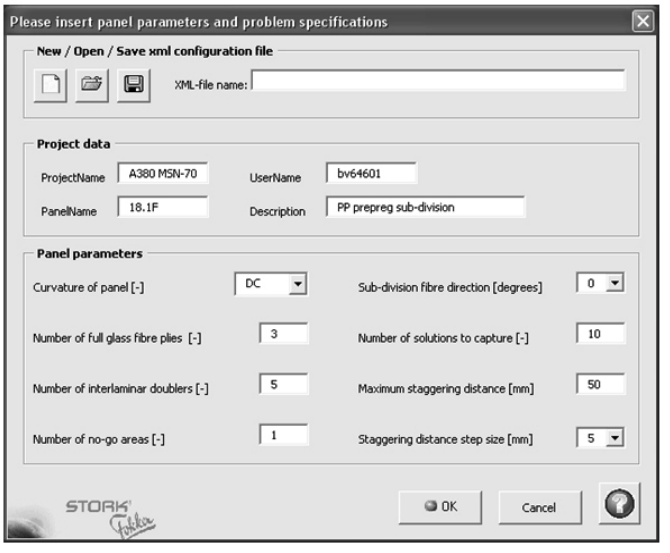



Figure 9. Data entry form, example of a user interface.

https://doi.org/10.1017/S0001924000004061 Published online by Cambridge University Press

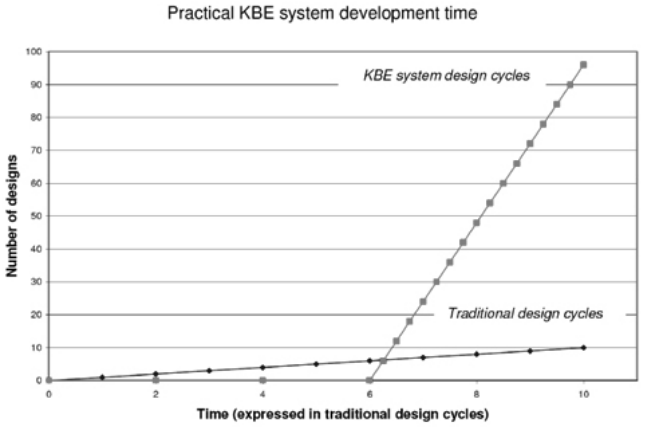



Figure 10. Time and design cycle comparison between traditional and KBE enabled prepreg division process.

useful during the training phase. However, it did not contain sufficient information to obsolete the need for retraining of the user after an interval period of not having used the application. Regarding the best practice database, no major contributions can be reported at this moment, since the concept has been introduced only recently.

Looking at the design process efficiency in terms of training and learning, it can be stated that best practices concerning the use of the application should be shared and reused. This can be done either by having the same user execute the same process steps on different products, or by having a best practice database operational and available to all end-users.

*Want*

Regarding the intrinsic motivation of the user, it can be stated that the user perceived solving the prepreg division problem as time consuming and frustrating. The user preferred to use the application to perform routine and dull tasks. However, given the very limited time available to solve the problem, it is a natural reaction to turn back to more familiar ground and apply the traditional design process. Users tend to be not so receptive to change if the work they regard as challenging is automated applying KBE techniques. In that case an external motivation is needed, to convince users to use the application (must).

*Have*

An important factor in the successful implementation of the KBE application is the time created by management to invest in developing, training and using the application. As discussed in the deployment section, project lead-time was limited, and the user proposed to go back to the manual process if time would be made available.

Furthermore responsiveness to errors or user comment has proven to be of key importance. Time pressure is constantly present, and when faced with issues users are likely to return to familiar ground, i.e. manual processing, if no (instant) resolution can be provided.

*Must*

During the knowledge acquisition and application development phases, a senior engineer acted as domain expert. His approval of the application in terms of functionalities and quality of the output convinced management and fellow engineers of the usefulness. To convince management of possible return on investment, lead-time reduction had been assessed in terms of traditional design cycles. With respect to the traditional design process a lead-time reduction of 75% can be gained by employing the KBE application, see Fig. 10. This reduction convinced management of the need to deploy the application in the project at hand.

*Measure*

To evaluate application performance, a benchmark of the manual process has been established. The measuring system compares the manual process benchmark relatively to the KBE enabled process in terms of required resources and lead-time.

## 6.0 CONCLUSIONS

- A KBE application to support the division of prepreg plies in FML fuselage panels has been successfully developed using the CATIA v5 CAD system. Using the application a lead-time reduction of approximately 75% is achieved.
- In order to develop the application a six-phase development process is applied. Especially during the structuring, application and implementation of knowledge differences during development between contemporary augmented CAD systems and dedicated KBE systems become apparent.
- It has been identified that implementing KBE techniques in design processes constitutes a large change from the traditional process. Different success factors have been identified, determining the success of the implementation. The article demonstrated that besides a focus on technological challenges in the KBE lifecycle (can factors), also attention should be paid to factors determining the acceptance of the application by engineers.
- The intrinsic motivation of users (want) can be positively influenced by focussing on KBE applications that automate routine and dull tasks. Users should furthermore be given the time to be trained in the use of the application and to learn how to best deploy the application in their daily work (have). The program management should emphasise the importance of using the application (must), because of process standardisation, quality improvement or lead-time reduction (measure).
- Implementation of KBE applications is not a one-time activity. Different users will operate the system in time, resulting in continuous need for training and support. Finally, the process settings are different for each use case, e.g. available lead-time, resources, project managers. Thus attention should be paid to the success factors throughout the lifetime of the application.

## REFERENCES

1. National Science Board, The Science and Engineering Workforce, National Science Foundation, 2003.
2. VINES, J. Key Trends in the Professional Engineering Labor Market, National Engineering, Science and technical Skills Summit, 2007.
3. DRUCKER, P. *Management challenges for the 21st century*, Butterworth-Heinemann, 2001.
4. VAN TOOREN, M.J.L., NAWIJN, M., BERENDS, J.P.T.J. and SCHUT, E.J. Aircraft Design Support using Knowledge Engineering and Optimisation Techniques, 46th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference, Austin, Texas, USA, 2005.
5. LA ROCCA, G. and VAN TOOREN, M.J.L. Enabling distributed multidisciplinary design of complex products: A knowledge based engineering approach, *J Design Research*, **5**, (3), pp 333-352, Inderscience Enterprices Ltd, 2007.
6. EMBEREY, C.L., MILTON, N.R., BERENDS, J.P.T.J., VAN TOOREN, M.J.L., VAN DER ELST, S.W.G. and VERMEULEN, B. Application of Knowledge Engineering Methodologies to Support Engineering Design Application Development in Aerospace, 7th AIAA Aviation Technology, Integration and Operations Conference (ATIO), AIAA-2007-7708, Belfast, Ireland, 2007.
7. VAN DER ELST, S.W.G. and VAN TOOREN, M.J.L. Application of a Knowledge Engineering Process to Support Engineering Design Application Development, 15th ISPE International Conference on Concurrent Engineering, Belfast, Northern Ireland, 2008.
8. MILTON, N. Knowledge technologies, Monza, Italy, *Polimetrica*, 2008.

https://doi.org/10.1017/S0001924000004061 Published online by Cambridge University Press

9. PCPACK, Software package, version 1.4.4R, Release 5. Epistemics, Nottingham, UK, 2006.
10. RUSSEL, S. and NORVIG, P. *Artificial Intelligence: A Modern Approach*, 2nd ed, Prentice Hall, 2003.
11. COOPER, D. and LA ROCCA, G. Knowledge-based techniques for developing engineering applications in 21st Century, 7th AIAA Aviation Technology, Integration and Operations Conference (ATIO), AIAA 2007-7711, Belfast, Northern Ireland, UK. 2007.
12. SCHREIBER, G. *Knowledge Engineering and Management, The CommonKADS Methodology*, Springer, 1999.
13. STOKES, M, *Managing Engineering Knowledge, MOKA: Methodology and Tools Oriented to Knowledge Based Engineering Applications*, Professional Engineering Publishing Ltd, London, UK. 2001.
14. VAN DER ELST, S.W.G. and VAN TOOREN, M.J.L. Domain specific modelling languages to support model-driven engineering of aircraft systems, 8th AIAA Aviation Technology, Integration and Operations Conference (ATIO), Anchorage, Alaska, USA. 2008.
15. BERENDS, J.P.T.J., VAN TOOREN, M.J.L. and SCHUT, E.J. Design and implementation of a new generation multi-agent task environment framework, 49th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, 4th AIAA Multidisciplinary Design Optimization Specialist Conference, AIAA-2008-2142, Schaumburg, IL, USA. 2008.
16. VERMEULEN, B. Knowledge Based Method for Solving Complexity in Design Problems, Dissertation at Delft University of Technology, Delft, The Netherlands. 2007.
17. VAN DER SPEK, R. and KELLEHER, M. Knowledge management, reducing the costs of ignorance. www.dnv.com/services/consulting/knowledge_management/Publications
18. VERMEULEN, B., VAN TOOREN, M.J.L. and PEETERS, L.J.B. Knowledge based method for fibre metal laminate fuselage panels, International Design Engineering Technical Conference and Computer and Information in Engineering Conference (DETC/CIE 2005), Long Beach, California, USA. 2005.
19. VLOT, A. and GUNNINK J.W. *Fibre Metal Laminates, an introduction*, Kluwer Academic Publishers, Dordrecht, The Netherlands. 2001.

https://doi.org/10.1017/S0001924000004061 Published online by Cambridge University Press