INTELLIGENT DESIGN SOFTWARE FOR A SMART CAR

Knowledge-Based Engineering in the automotive industry

The European automotive industry is under threat. Manufacturers and their suppliers face massive competition from low-cost products manufactured in low-wage countries. Moreover there is another equally critical threat, from consumer demand for increased technical sophistication at lower cost. The EU-sponsored Pegasus project was initiated four years ago to face these challenges. The project researched new materials, manufacturing and assembly methods to decrease lead-time, reduce cost and improve manufacturing flexibility. The department of Systems Engineering and Aircraft Design (SEAD) was involved in the development of an Integrated Design and Engineering Environment, trying to achieve some of the aforementioned goals through intelligent software.

TEXT **ir. Reinier van Dijk, Ph.D. candidate, Systems Engineering and Aircraft Design**

With Daimler as the main industrial partner, Pegasus was built around a next generation car concept for the Smart Fortwo. This car is of low weight and cost, primarily through the revolutionary use of plastic material. Due to advances in material sciences the application of plastics is growing, ranging from interior to exterior components and the power train. Consequently all exterior body panels of the Fortwo are made from elastic plastics (see figure 1). Pegasus aimed to improve the development, manufacturing and assembly process of the Fortwo plastic fender component (figure 2), being a representative complex component in the overall car concept. The SEAD department led a consortium of companies that together would study how software could improve the "Product Development Process" (PDP) of this plastic fender component. The initial phase of the project was used to identify challenges in the PDP. This led to three main observations.

INDUSTRY CHALLENGES

In the automotive industry the most experienced engineers are eligible for retirement. Hence a brain-drain phenomenon is occurring with the result that valuable knowledge is lost. Wheels are re-invented, old errors are repeated and the use of best practices is not systematically enforced. Moreover the effective application of Multidisciplinary Design Optimization (MDO) techniques to improve product quality is desired, but not yet a reality. Interrelations between the different disciplines are barely considered, resulting in suboptimal designs. Finally a lot of repetitive design tasks are consuming too much time, increasing cost and reducing creativity.

THE IMPORTANCE OF KNOWLEDGE MANAGEMENT

The general solution to the aforementioned problems starts with the effective management of the most valuable engineering resource in the 21st century: our knowledge. Knowledge Management (KM) is a vital research field that comprises a range of strategies and practices used in an organization to identify, create, represent, distribute, and enable adoption of insights and experiences. Such insights and experiences comprise knowledge, either embodied in individuals or embedded in organizational processes or practice. With KM vital engineering can be made explicit, stored in a Knowledge Base (KB) and therefore retained (figure 3). It can be used to effectively train new recruits replacing the older workforce and increase inter-disciplinary awareness, important to achieve MDO from an organizational perspective. In principle the benefits of KM are endless.

THE IMPORTANCE OF KNOWLEDGEABLE SOFTWARE

KM alone, however, is not sufficient. Another vital component in the hunt for short lead times, low cost and high quality is the computer, and the role that we attri-


DAIMLER

Figure 1. All exterior body panels of the Smart fortwo are made from elastic plastic materials


PEGASUS CONSORTIUM

Figure 2. Smart fortwo fender


Figure 3. Knowledge capture


SCHNEIDER FORM

Figure 5. Digital mock-up of the mould

PEGASUS CONSORTIUM

Figure 6. Drilling holes in the mould

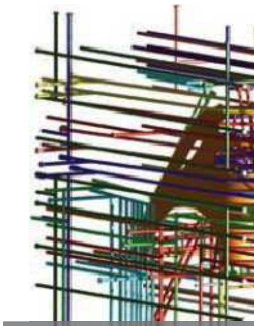


Figure 7. Cooling system s
the digital mock-
waterline length

bute to it in the PDP. It is time to revisit the idea that the computer is a mere "number cruncher" and realize that a computer can be used as a highly strategic resource. It can reason with knowledge to provide us with advice or help solve complex problems. The evidence of artificial intelligence and its practical relevance is building up rapidly. Unfortunately software developments in the engineering domain do not follow this trend with equal pace.

THE PITFALLS OF TRADITION

Over time engineering software developments have mostly exploited the computer's ability to crunch numbers. Commercial engineering codes are used to solve flow equations while Excel sheets are used to add cost figures. Their use has become widespread, and it is not unusual to find that many engineers spend most of their time interacting with Computer-Aided Design, Engineering, Manufacturing (CAx) or other types of software. However, their use also has a downside. All this software needs to be set-up, operated and results need to be extracted. Moreover, all software applications were developed as separate entities without any regard to other applications. Consequently, it is hard to integrate them strategically, while the interchange of data requires tedious translation steps and data transfer. Furthermore, the knowledge required to perform these activities resides in people's heads making it a human effort which is also time consuming. Research shows that up to eighty percent of all human effort is spent on routine and dull design tasks which add little value to the final product. To help eliminate such tasks, it would be desirable to have intelligent software applications that automate this work for us. Fortunately the conductance of routine design tasks requires rather procedural knowledge of the logical steps need to accomplish an activity, and can usually be made explicit. If formalized well, Knowledge-Based Engineering (KBE) applications can be built that re-use this knowledge in order to automate repetitive work.

KNOWLEDGE-BASED ENGINEERING

KBE is based on parametric modelling techniques to describe a "family" of products. The parametric model consists of a finite set of basic objects (think of it as an inventory of flexible LEGO® blocks). These objects can be assembled in different ways to create the desired product configuration. All objects in turn have a finite set of attributes. Variations within a product configuration are achieved by changing these attribute values. Besides being flexible, the parametric model is knowledgeable and can store rules to automatically create a geometric model. As such, KBE is a useful technique to eliminate the time involved in geometry manipulation tasks in typical CAD environments. Moreover the model can incorporate rules to autogenerate "simulation-specific" models required by CAx systems. With KBE, different CAx systems can be consistently integrated to a single, central model and the time normally spent on the creation of simulation-specific models can be reduced from hours to fractions of a second, cutting down on design time and cost. Moreover automation, parametric modelling and integrated modelling are also essential ingredients to truly enable MDO. With KBE, the engineer can not only study multiple disciplines simultaneously, but also investigate multiple product variants automatically. Hence KBE enables a thorough exploration of the design space, benefiting product quality.

FINDING THE RIGHT DOMAIN

Now that we understand the challenges and solutions a little bit better, it is time to put this in practice. The initial years of the Pegasus project were used to identify the areas of the fender PDP where KM and KBE would prove beneficial. Car styling is an example where KBE is not useful. Styling deals with aesthetics. It is a highly creative process and hard to capture in clear-cut rules and product variants. In contrast, the development of plastic injection moulded parts meets the KBE selection criteria very well. The development of plastic injection moulded parts was defined as the principal area for an integrated design engineering environment (IDEE) demonstration.

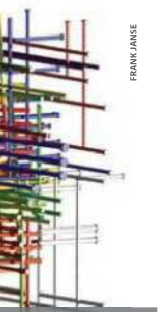
PLASTIC INJECTION MOULDING

Injection moulding is a manufacturing process extensively used for the mass-production of plastic parts, like the Fortwo fender. The process begins with plastic material being fed into the heated barrel of a moulding machine. When the plastic melt has reached its prescribed temperature, plastic is injected into the cavity of the mould. During injection the pressures in the cavity build up rapidly. The moulding machine tightly holds the mould plates together to prevent prema-

PEGASUS CONSORTIUM

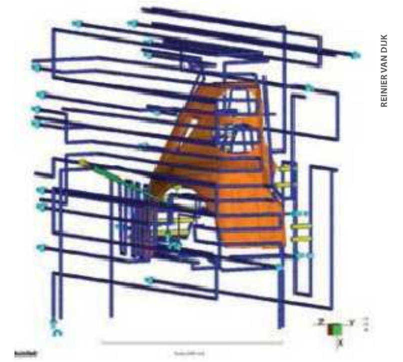
PEGASUS CONSORTIUM

Figure 4. The Smart fortwo fender is automatically removed from the mould by a robot arm. Clearly visible is the giant machine enclosing the mould.



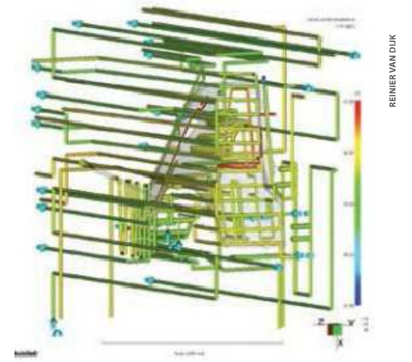
FRANK JANSE

napshot from
up. The overall
is significant.



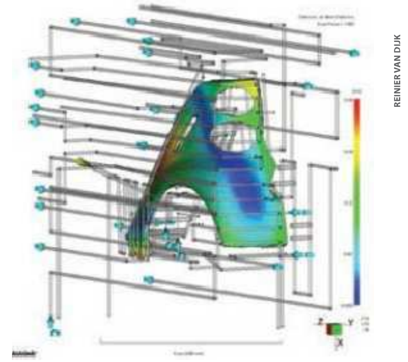
REINIER VAN DIJK

Figure 8. Moldflow simulation model



REINIER VAN DIJK

Figure 9. Cooling water temperature profile



REINIER VAN DIJK

Figure 10. Component warpage

ture opening during this phase. When the cavity is fully filled, the melt is (actively) cooled and hardens until it reaches the final shape. At that stage the machine clamping force is fully released, the mould opens and the new plastic part is ejected and (robotically) removed from the mould (figures 4 and 5). After removal the mould closes again and another injection cycle commences.

The design of injection moulded parts is a challenging endeavour, and in particular, need to be manufacturable. Certain product features are hard to mould, resulting in complex and expensive tooling and in the worst case scenario lead to expensive design overhauls. Given a potentially suboptimal design, the mould maker in turn should develop tooling within ten weeks from initial order, leaving them little time to optimize the design, compromising quality. Process characteristics like cycle time, raw material cost and cost of energy are largely an outcome of part and tooling designs, however they are usually also objectives that are determined at the beginning of a project. These are just some examples of the overall complexity of various systems and stakeholders. The process is full of system dependencies, but should result in parts quickly, cheaply and with high quality.

INTEGRATED DESIGN AND ENGINEERING ENVIRONMENT

The IDEE was developed to support collaborative and integrated development of plastic injected parts. KM techniques were used to elicit valuable product and process knowledge from experienced automotive engineers, mould makers and moulders. Their cumulative knowledge was captured inside a KB. A KBE application has been built incorporating a parametric model of a plastic injection mould. For a range of mould configurations, the tool is able to automatically generate geometry and integrate various CAx systems to it for simulation purposes. Finally it uses Process Integration and Automation (PI&A) software to link the different software applications (KBE and CAx) into a seamless framework, control their execution and support all data exchange in automated fashion. In summary, the software framework relies on three basic software components provided in table 1. IDEE has proven to be a useful tool for material selection, automated feedback on the manufacturability of plastic parts and as a planning tool and collaboration platform. IDEE also allows the engineer to study different materials, mould configurations and moulding process settings. Optimization algorithms have been incorporated to find optimal designs automatically. A case study on the cooling sub system design of a plastic injection mould will serve as a more detailed example of the IDEE's working principles.

IDEE FOR COOLING SYSTEM DESIGN

The standard method of cooling is to pass a coolant (usually water) through a series of holes drilled through the mould and connected by hoses to form a continuous pathway. Development of the cooling system is a typical topic for MDO. First of all critical dimensions and part quality are affected by cooling. During cooling the part will shrink, which is a natural phenomenon. However any differential shrinkage will cause warpage. This will result in a part that does not follow the intended shape, which is obviously undesirable. The cooling system must therefore be designed such that uniform shrinkage is achieved and therefore minimum warpage. Cooling is the biggest contributor to the overall cycle time, which should be minimal in mass production processes to retain production capacity. Hence, the cooling system should be designed such that it cools the part quickly. Moreover, cycle time and energy consumption are related to each other; the longer the cycle lasts, the higher the energy bill, increasing costs and (potentially) environmental impact. Finally, the typical length of all waterlines in a complex production mould can be of the order of several hundreds of meters. Consequently the annual manufacturing time and cost involved of machining and

Table 1. IDEE core software applications

Software	Company	Discipline
PCPACK	Tacit Connexions	Knowledge Management
GDL	Genworks	Knowledge-Based Engineering
Optimus	NOESIS	Process integration, automation and optimisation

manual labour are significant. The objective of minimizing cooling system length is a good way to minimize manufacturing time and cost.

In summary, the mould maker has to perfectly balance a diverse and conflicting range of disciplines, including (but not limited to) cycle time, manufacturing and moulding cost, part quality and environmental impact. In practice the mould maker only has 24 hours to design the complete cooling system layout. Such a time frame does not allow him to run multiple what-if studies for different sizes, positions and constellations of waterlines even though this is highly desirable. The IDEE was therefore configured to enable a mould maker to draw an arbitrary cooling system in his Digital Mock-Up (DMU) and investigate its behaviour.

It was identified that three computer aided engineering (CAE) software applications were required to simulate product behaviour for all different disciplines, see table 2. Firstly Moldflow was identified as a widely-used commercial software application capable of simulating plastic flows and cooling processes and determining cycle time, shrinkage and warpage. The IMCEC was custom-built during the Pegasus project since no software tool was able to represent all highly company- and machine-specific rules and knowledge required for cost and energy calculations. Based on interviews the IMCEC application has been programmed to automatically derive cost and energy estimates on the basis of a fender-specific body of knowledge. The CycleIT software was used for Life-Cycle Assessments. CycleIT is based on an immense database located in France, which holds environmental impact data for transportation, manufacturing, assembly and recycling processes.

The cooling system definition in the DMU (CATIA v5) was represented by a system of drill holes. The central KBE application holding the product model of a general injection mould was extended with the capability to extract the drilling lines from the DMU (figure 7). Moreover some algorithms were incorporated to re-engineer the waterways and to recognize special cooling system elements from those drilling holes. The KBE application was instructed in how to create Moldflow, IMCEC and CycleIT specific models in order to closely integrate them into one process. figure 8 shows an automatically generated Moldflow model. Subsequent cooling and warpage analyses provide for example cooling water temperature profiles and warpage results (figures 9 and 10).

Table 2. CAE software application

Software Application	Disciplines	Outputs
AUTODESK® MOLDFLOW® SYNERGY	Plastic flow and cooling simulations	Cycle time, shrinkage, warpage
Injection Moulding Cost and Energy Calculator (IMCEC)	Cost and energy estimates	Manufacturing/moulding cost, moulding energy consumption
CycleIT Automotive Software	Environmental impact estimates	CO2 emissions, human health, etc.

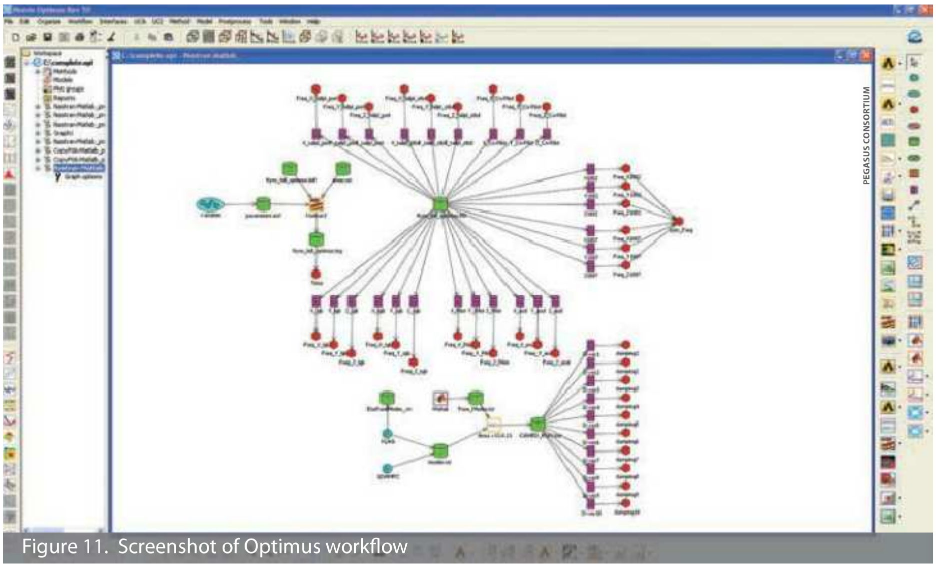


Figure 11. Screenshot of Optimus workflow

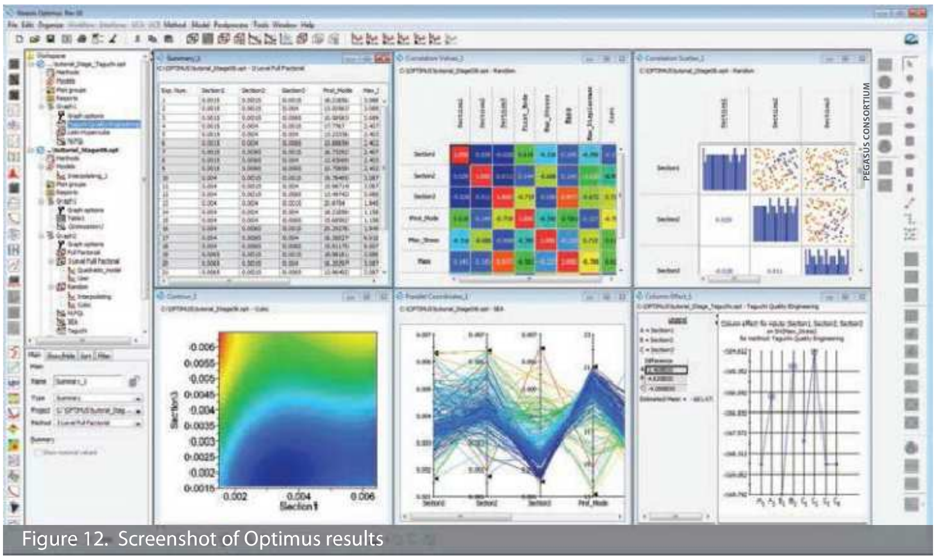


Figure 12. Screenshot of Optimus results

Finally, all software applications were integrated into one automated worfklow. Optimus controls their execution and supports all data exchange in an automated fashion. The resulting software worfklow is shown in figure 11. For each study Optimus automatically extracts the important results. An example of results obtained with Optimus is given in figure 12.

CONCLUSION

An important aspect of the IDEE software framework is that all software applications are either directly developed as web services or slightly adapted to be available as such. On condition that there is an Internet connection available, this enables a user to connect to these software services essentially from anywhere in the world at any time. The user does not need to install any client-side software and what he sees on his screen is a single software environment with which to interact. The IDEE framework will be extended to include more aspects of the PDP for plastic parts. Initiatives have started to prove the IDEE on an industrial scale. If you felt inspired by the IDEE, be assured that its methodology is very well applicable to almost all engineering domains. It only depends on these domains to realize that. If you want to contribute to IDEE developments, do not hesitate to contact the author! ✈

References

Pegasus project website: http://www.pegasus-eu.net/