

From a cynical engineer's perspective, PLM (product lifecycle management) might sound like the latest buzzword for selling the latest piece of software. That view changes, of course, when the software begins to make that engineer's life easier.

PLM software controls and records the life of a product from concept to manufacturing. The idea is to embed engineering knowledge into the software, so that mundane tasks can be automated and future designs benefit from lessons already learned, rather than having to start with a blank sheet each time.

Intrinsys is the PLM consultancy arm of engineering company Integral Powertrain. It specialises in offering companies the insight it has developed for its own needs and for clients. Embedding knowledge into PLM systems can be a key tool to maintaining competitiveness, it says.

This kind of knowledge-based engineering (KBE) is within everybody's reach, says Andrew Early, an application engineer at Intrinsys: "It's not just for the BOCs and British Steels of the world that have to invest millions in these things."

In addition to competitiveness, there are also advantages for in terms of knowledge retention. Embedding knowledge in software means it doesn't leave the company when employees do. For engineers, it can make work less repetitive and act as a prompt to stop and think about ways of working that may have become more about habit than logic.

The idea behind KBE is that computers can be taught to do the repetitive, boring tasks so that engineers can concentrate on the bigger picture. It is about more than just defining macros, though. At its most basic level, KBE involves setting sensible parameters for variables and specifying the relationships between interrelated values. This allows such figures to be calculated automatically and fed into any further calculations, ensuring consistency and avoiding keying-in errors.

Logical checks

Beyond this, logical checks can also be put in place so that, for instance, holes cannot be drilled too close to the edge of an object, based on the material and thickness being drilled through. Values can be passed from the design of one item ready for the design of connecting or associated parts.

But KBE can be taken further, embedding common practices, assumptions and industry-specific knowledge into the way the software works, effectively turning cross-sector software such as

Catia into one that is tailored to a specific task.

Developing templates takes this process a step further. This involves sitting down with a company's engineers and critically assessing current processes and working practices. It should become clear during these sessions what knowledge is being brought to bear when tasks are tackled, helping to demystify the design process. Lessons learned from previous challenges need to be recognised and experience-based assumptions need to be identified.

Come the next job, a customer's specifications can be applied to the template created from this newfound understanding. The extent of the template will vary between items, and may well leave a degree of undefined, free geometry that the operating engineer must resolve.

As Luke Barker, technical director of Integral Powertrain points out, talking about processes often uncovers greater repetition than expected: "Things tend to turn out to be more 'templateable' than you think. If there's knowledge there, you can capture it."

This process not only removes repetition, but also means that each new

job builds on the design breakthroughs and techniques learned from previous work. It removes the need to reinvent the wheel every time a new design is required.

Integral Powertrain has been able to template various aspects of its engine design work. Its templates allow the desired engine speed and fuel pressure to dictate various properties of the gudgeon pin, which connects the piston head to the con rod. This information flows through to help choose the design of piston head, which in turn dictates the properties of the crankshaft. The software checks the balance of the crankshaft

and, through a series of iterative steps, allows the engineer to tweak a selection of variables until the best solution is found. The benefits are clear to Nick Lawther, a senior engineer at Integral Powertrain: "A process that took three weeks to get a fully balanced crank can now take half a day."

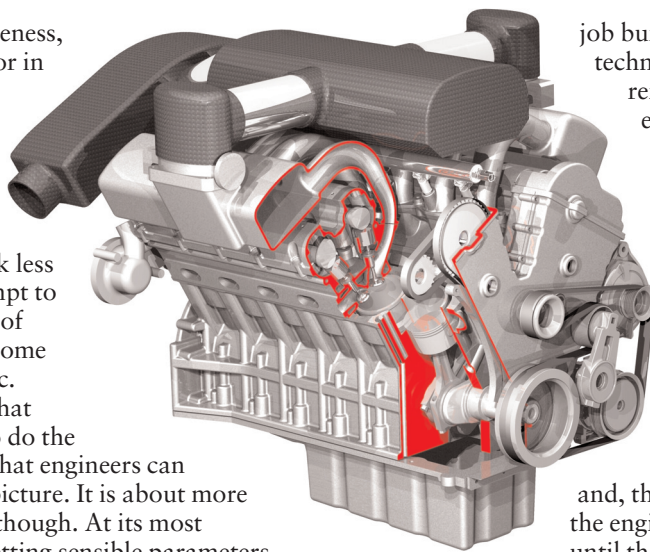
As well as achieving results faster, Integral Powertrain says this method allows it to evaluate more options than would have been possible without the templates.

This addresses one of the major concerns about KBE: that embedding assumptions into parts of the design and development process can prevent innovative or original ideas. After all, it is difficult to achieve the sought-after paradigm shift if your software is entrenched in the existing paradigm.

Barker dismisses the idea: "A lot of people think that knowledge-based engineering locks you into a specific solution. I think it does the opposite – it makes you look at the rules that you're using to see how they work and how they can be improved."

In the know

Companies using "knowledge-based engineering" can avoid reinventing the wheel, cut down repetitive tasks and speed up the design process, as *Richard Butler* explains



Souped-up design: Software can be tailored to simplify working on inter-connected systems such as engines