The migraton project that put Murphy's law to the test

fecher and Maginus had to overcome several challenges before the VB6 back-end of Maginus' Order Management System could be successfully migrated to C#.NET

What does Manchester, England based Maginus do for its clients? Answer: Everything that's behind the "buy" button. The firm has been at the forefront of e-commerce since the late 1990s and the highly diverse client list spans a range of industries and business models. Among its clients are the catering equipment provider Nisbets, AutoStyling Truckman, the waste and recycling container specialist Straight and the Royal Society for the Protection of Birds. Whereas consumers generally interact with the front end of each respective website, the back end, namely the Maginus Order Management System (OMS), is internally used throughout the organization by warehouse operators, purchasing department staff, the contact center taking phone orders and finance departments generating reports. The modernization project conducted by fecher had to first effectively handle some unexpected challenges before the complex business software could finally be successfully migrated from Visual Basic 6 (VB6) to C# and the .NET framework.

The VB6 migration was part of a major software modernization project that Maginus had launched in 2017. Apart from replacing the outdated programming platform, it also migrated the application server from Solaris to Linux, moved all deployment from on-premise to a cloud-based approach, switched its database technology and refreshed all other Microsoft components to ensure they were .NET compliant. "Of all these project sub-components, the VB6 migration was the most challenging one," Maginus' Technical Director and project leader Martin Pickering explains. "Compared to this dramatic change in the technology underpinnings, upgrading from one version of Unix to another is relatively straightforward."

> *"Other vendors wanted to sell us tools for the migration, fecher offered to take the risk for us and actually do it."*

"Our original plan was to perform the migration by ourselves," Chief Operating Officer Simon Weeks remembers. "However, when we quickly realized it involved an extensive amount of work, we began researching firms that could offer this as a service." They identified a few companies that wanted to sell a tool for the Maginus team to use and others who wanted to rewrite the entire application in the form of an offshore-type project. "We calculated that it would have taken us four years to explain the intricacies of the software to someone well enough to equip them for the rewrite and that this probably would not have been successful anyway. fecher was the only vendor that offered to actually perform the migration for us in a tool-based approach."
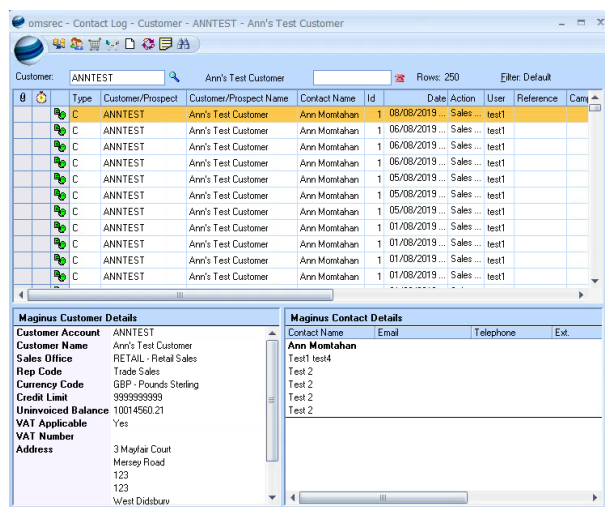
Clear expectations are set

Before finally committing to the project, fecher conducted a proof of concept to demonstrate what the outcome would be and to obtain a well-founded estimate of the amount of work the project would entail. "We picked a representative module of relevant size, sent fecher the code and after six weeks they came back to us with a working prototype in C#," Pickering remembers. "This was essential as it reassured us that they knew what they were doing and could achieve substantial results with our product." "fecher, for its part, had the reassurance that this was something they could do and could calculate estimates for. Their fixed price offer then removed for us a lot of the risk from the project," Weeks adds.

> *"For us, it was a highly collaborative project*
> *from the very beginning."*

The migration project was set up as a joint effort from its inception in early 2018. "As the entire future of our company depended on the success of this project, we did not want to treat it as a black box," Weeks recalls. "We saw it as a highly collaborative project from the outset." They therefore decided to actively play three principal roles in the collaboration. The first of which was to explain to fecher how the software worked from a technology point of view. In this role, it was mainly Pickering and his team of developers who worked with fecher to get the project started.

The second role pertained to a group of super users commissioned to create a number of "test case" videos. "Rather than trying to communicate how to use the software and what our expectations were via specifications and user manuals, we just did it as a show & tell through lots of videos," Pickering explains. The important thing was that the videos needed to be individually shot; each having a clear purpose, a clear start and an ending with a measurable outcome. Later in the project, these videos would form a basis enabling fecher to verify the migration results before they handed the code back to Maginus. At this point, their third role began: The final level of testing included user acceptance tests and went way beyond what was in the videos.
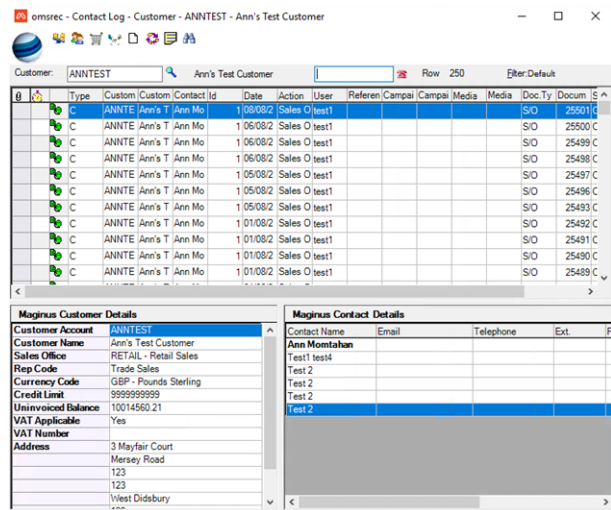

The migration started with the integration code for the application server which went smoothly; it became stable almost immediately. Then, after going through the migration tool, the forms, around 550 altogether, and the accompanying often complex business logic needed some code adjustments. This hybrid of tool-assisted migration and manual finalization by experienced software specialists is customary and standard in modernization projects by fecher. Such projects often also include a modernization of the application's user interface.

Every VB6 form was transformed …

"We were however not interested in altering the way the software worked, behaved or looked as part of this migration," Pickering says. "Our ultimate goal was to get rid of VB6 and not take unnecessary risks by adding in functional renovations." For this reason, the project followed a strict 1:1 migration approach meaning each VB6 form was migrated to a functionally and optically corresponding WinForms form.

... into an equivalent WinForm form

Anything that can go wrong, will

Later on, when trying to modify these migrated WinForm forms in Visual Studio, many could not be opened at all or the attempted change led to a corrupted result. "It turned out that the culprit was a bug in the vbPORTER tool used by fecher that took them a good while to sort out," Pickering explains.

> *"fecher really took ownership of the problem and did everything they could to resolve it."*

The second obstacle the project ran into was even worse: The original VB6 software relied heavily on an ActiveX server executable for context switching, which for example let a user to jump from one application to another while continuing to work with the same customer. "It was a heart-stopping moment for all of us when we realized that this concept has no equivalent in the .NET world," assures Pickering. The senior architects from fecher and Maginus immediately sat together and came up with a solution—a rewrite of the inter-process communication in WCF technology.

"We ended up dealing with this in a huge joint effort by both teams," recalls Weeks. Obviously, this additional development project meant a further delay for the project, even impacting software that was not part of the migration. "What impressed us the most, however, was the attitude that fecher took about the situation. They really owned the problem and did everything they could to solve it. Their tenacity and perseverance has been a hallmark throughout the project."
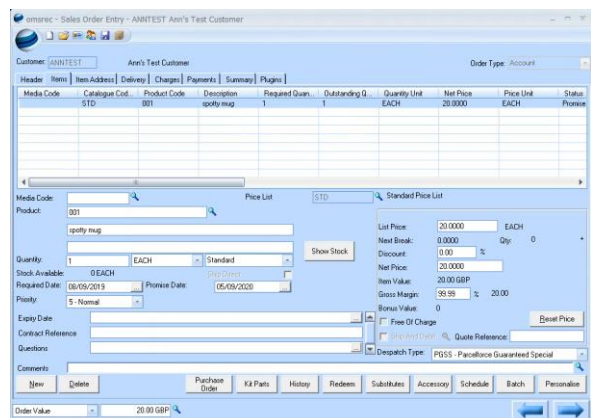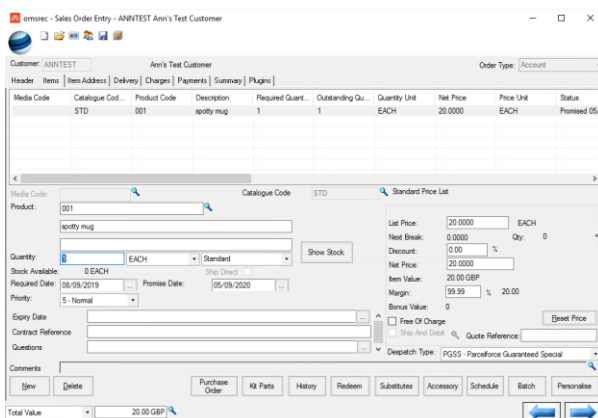
All's well that ends well

Simon Weeks, Chief Operating Officer, Maginus

After all of the final testing was finished and the remaining bugs were ironed out, the software was finally ready to be field tested in April 2020. Since the announcement of Maginus 10, as the new version of their software will be called, customers are keen to get their hands on it. "Feedback from our beta users is very promising so far, but it is still early days," Weeks summarizes. "For us, this is a long term investment. The previous incarnation of the technology has lasted us for over 20 years. We see this version of the software carrying us into the next 10 years or so."

"Despite the delay of several months, I think we did extremely well on this project," concludes Pickering. "We gained a lot of confidence from the very beginning because the delivery was phased and we weren't just dropping our code into a black box and waiting for fecher to hand over the finished product. Moreover, regular key milestone progress reports also helped us when things were not going entirely as we would have liked. We always knew that any problems would be solved and that we would not be abandoned."

Martin Pickering, Technical Director, Maginus

To which Weeks wholeheartedly agrees: "fecher always stood by their promises. Our project probably ended up being more expensive for them than they originally anticipated, but they basically did what they said they were going to do. fecher was always honest and extremely professional!"