

IC programming platform migrated from VB6 to C#

Application modernization project by fecher generates added value for Data I/O and their worldwide electronics manufacturing clients

With roughly 100 employees, Data I/O is one of those hidden champions: When it comes to programming integrated circuits (ICs), everybody knows the company as the #1 global player since its inception in 1972. Their clients can be found everywhere around the world in the automotive, wireless devices, telecommunications, industrial controls, and consumer electronics industries. While the business was primarily focused on hardware, its software has increasingly become an important part of their product offerings. An application modernization project by fecher has recently helped to generate added value for Data I/O and their clients by migrating the core handler software from VB6 to C# / .NET.



Data I/O, with headquarters in Redmond, Washington, is the leading global provider of advanced data and security programming solutions for flash, flash-memory based intelligent devices and microcontrollers for automotive, Internet-of-Things, medical, wireless, consumer electronics, industrial controls and other markets.

www.dataio.com

“Software is not only an important part of our current offering, it has also become a way to differentiate our products from the competitors’,” Carl Olson, Director of Hardware Engineering, explains. Accordingly, Data I/O’s portfolio of software products and their capabilities have been systematically expanded over the years.

A milestone in this process occurred in the year 2000 when Data I/O acquired a former competitor. The automated handler machinery they produced, a pick-and-place machine handling the ICs during programming, came with an advanced software solution to operate it. This software, AH, communicates with a separate package, the job setup software, which defines the actual task to be performed. In addition, AH has a user interface that runs on the machine’s console where it can be used by the operator to start and stop the job, modify parameters and look at statistics in real time.

VB6 – from heritage to legacy

“The handler software was very functional and the fact that it was developed in VB6 did not really represent a problem to us back then,” Carl remembers. Therefore, Data I/O adopted the software to become the core solution for all their handlers.



PSV7000 - Data I/O's automated programming system

Data I/O continued to maintain and develop the AH software up to the current AH700 version. “When Windows 7 came out, running a 32-bit VB6 application on 64-bit Windows seemed far from ideal. Further, compared to the new UI design, the software started to look dated.” After discussions with management and the development group, Carl’s conclusion was clear: AH700 had to be updated to a modern software platform.

“By 2015, the list of hardware and software systems that we needed to interface with had grown dramatically,” says Mark Knowles who became Lead Engineer for the modernization project. “To cope with the

integration requirements and take advantages of new capabilities, we really needed a modern software architecture.” They made the decision to migrate from VB6 to the Microsoft .NET framework with C# as the programming language. “.NET was the platform that most of our other software solutions were based on. Migrating to from VB6 to VB.NET as a language was not an ideal alternative. The cost to migrate to VB.NET or C# would have been the same, however VB.NET would not deliver all the advantages of C# that we gained with a Microsoft .NET solution.”

“The driving forces were to modernize and also find developer resources more easily.”

The decision was not only a technical one, as Carl adds: “Most developers are more likely interested in C# than in VB.” So, the ability to hire new talent was another plus for a C# solution. “Basically, the driving forces behind the change was to modernize, to make the software more flexible and to allow us to find more resources that can easily come in and help when needed.”

Migration as a packaged service

Given the limited development resources at Data I/O, outside help was needed. “Migrating an entire product with roughly 140.000 lines of code seemed too much for our internal resources”, Mark sums up. In a quick Google search, he found a handful of companies who offered VB6 to C# migration. As it turned out, the application modernization specialist fecher was the only vendor who offered this migration path as a packaged service and therefore became the clear choice. “There was a well-defined, structured migration process which fecher had lots of experience with,” Carl explains. “They even provided a slick frontend tool that gave us an analysis of our VB code and a good quote for the migration to C#.”

“There was a well-defined, structured migration process which fecher had lots of experience with.”

To present the results, discuss the offer and further scope the project, a meeting was scheduled at Data I/O in December 2015. “Konstantin came over from Germany, gave a great presentation and laid out where we were going. Heike, from fecher’s US office, was very responsive and organized everything extremely well,” as Mark remembers. “It was immediately clear to us that they took our business serious and really understood theirs.”

Therefore, it was a straightforward decision to move forward with the proposal that the application modernization specialists had put on the table - a migration of the existing VB6 code to C# / .NET at a fixed price and with a guaranteed delivery by the end of 2016.

A truly international project

The project started in spring 2016 with forming the project team. In addition to Mark, who led the project on the Data I/O side, and a software engineer in their German office, fecher contributed a team leader in Germany, team support in Washington, D.C. and a group of three project engineers in the nearshoring center in Romania. “As we were all used to working in international project teams, communication was never an issue,” Mark explains. “Everything was documented in Microsoft SharePoint, so I knew at any time what was going on. The different time zones did facilitate the process. When I put an issue in SharePoint in my evening, it was usually solved by the next morning when I came back to the office.”

“Communication was never an issue.”

At the start of the project, Data I/O provided the AH700 source code and the fecher software engineers started migrating it to C#. In early summer, when the first software modules were converted, the test phase began. “Our test team checked out the software that came back and whenever there was a problem, we opened a bug in SharePoint, adding test classes or screen videos to further explain the issue. Usually by the next morning, the bug was fixed,” Mark describes the everyday routine for the next few months.

During the migration, the project scope was expanded to include additional executables and DLLs. For example, the software to run the labeler was an additional executable. Still, the project was delivered on-time. By the end of 2016, the existing code was converted to C# and the system was up and running again on the .NET framework.

Adding value for the user

“At that point, we analyzed the situation and decided to go a step further,” says Mark. Instead of deploying the freshly migrated .NET application right away, Data I/O decided to take the time to further refine the software architecture to a model view controller approach and re-skin it with Windows Presentation Foundation (WPF). “Being able to do this was the main reason why we wanted to move to .NET in the first place. And now, after the AH700 was all ported and every functional aspect was done, the software was working beautifully. We even had an application that looked a lot better than it did in VB6.”

“We had an application that looked a lot better than in Visual Basic.”

“Customers are not really interested in whether the software is VB6 or C#, they want to know what value it brings to them. That is where Data I/O wants to focus the rest of our time in 2017,” as Carl puts it. A good example for such added value is the new context sensitive help technology. By pointing at screen elements, the operator now gets information on how to use each setting and what it is good for - either as text or even as a pictorial. Also, the resizable new user interface makes better use of the larger screen sizes of today’s handler equipment.

Last but not least, the new architecture will help to seamlessly integrate the AH700 software with other applications in use at the client sites and bring it in sync with the reference architecture that Data I/O has come up with during the migration period.

A true technology partner

All this would not have been possible without the move to .NET. “The work that fecher performed for us is the foundation for a significant software release that we are doing later this year, that not only allows us to present a nicer and more intuitive user interface, but is up-to-date in every single aspect of software engineering,” Mark reports.

“fecher hit on all cylinders.”

“There are three aspects of working with fecher which we really enjoyed: They were committed on delivering what we needed and do what they said they would do, they did it in the timeframe they said they would do it in, and they did it to budget,” Carl adds. “Those are three huge factors when you are dealing with an outside partner. fecher really hit on all three cylinders.”

Contact & Legal Notice:

Published by: fecher GmbH
Otto-Lilienthal-Str. 12
63322 Rödermark, Germany

Phone: +1 (202) 629-0294
+49 (6074) 80577-00

E-mail: info@fecher.eu
Website: www.fecher.net

CEO: Günter Hofmann