

Borland®

Borland® C++Builder™ at Ware4 Technology

“Our goal: to deploy a next-generation customer service center, developed at low cost and delivered with zero defects. Our mission statement: to use the ‘latest and most valuable technologies.’ Our choice: Borland C++Builder.”

—Michael Mann, President,
Ware4 Technology

Giants of the EFT industry join hands

The electronic funds transfer (EFT) arena has seen significant consolidation and collaboration over the last several years. One recent collaboration of Fortune 1,000 companies (including Bank of America and First Data Corporation) required a solution to address the need for a new, state-of-the-art customer service center. The new service center would be required to support a network of 10,000 ATM machines/kiosks and provide assistance, manual intervention for the retrieval of cash, the voiding and refunding of transactions, account and transaction history information, the blocking and unblocking of ATM cards, and other processes. Achieving this goal would require superseding two functional, but aging systems: a single-tier system in need of upsizing, and a legacy mainframe-based system. The call center’s mandate for the redesign was to leverage Internet technology, allowing for a more flexible and real-time design, development, and use. Building the new system on Internet and Web technologies would provide a software environment that could integrate new technologies as they became available.

As is the case with most software projects, the redesign had to be completed within strict time and budgetary allowances. Ware4 Technology was chosen to provide the solution due to its successful track record—a record established by previous successes in designing solutions for financial institutions.

Technology

Application	EFT call center software
Tool	Borland® C++Builder™
Other tools evaluated	Microsoft® Visual C++®
Database server	Microsoft® SQL Server™
Hardware platform	Windows NT® 4.0, 2000, XP™
Number of users	50 machines
Team size	5 developers
Development time	6 months



Challenge

Analysis of the call center requirements yielded a simple approach: representatives would service customers through an elaborate, but flexible, decision tree. There were some initial technical requirements the project had to meet:

- The existing system had been built with C and C++. It was decided that the best use of resources and existing investments would be to leverage existing C libraries and source code where possible. The ability to easily compile and link with existing code meant that the C++ language would be used for this solution.
- The underlying presentation technology needed to be available on a wide variety of hardware and software platforms, therefore HTML was chosen.
- The database would have to handle both present and future usage loads and allow access by third-party reporting and querying tools. Therefore, a scalable database and flexible database architecture would be required of the development solution.
- The solution would be required to grow with future needs and technologies rather than be replaced. Therefore, the development framework must offer a comprehensive suite of technology designed with expansion in mind.
- Implementation time must be low, quality high, therefore a single source toolset would be required to cap off integration and debugging costs.

Ware4 Technology researched the available solutions such as Java,™ Microsoft® Visual C++® and Visual Basic®: Ware4 Technology determined Borland® C++Builder™ to be the best solution to deliver all the needs of the project and in the fastest possible time.

C++Builder™

The Borland® Solution

The development environment and functionality of Borland® C++Builder™ was key to the success of this project.

The customer service application runs on Windows NT® 4.0, Windows® 2000, and Windows XP.™ The main windows of the user interface seamlessly leverage Microsoft Internet Explorer® for HTML rendering. With C++Builder, integrating external components such as Internet Explorer is a simple drag-and-drop operation. Custom, interstitial code connects the properties and methods of Internet Explorer with the Borland framework, resulting in the ability to specify and display, in the browser, any form fields and any information from the database. The solution is extremely flexible as it allows the freedom to intermix HTML elements, ActiveX® Web controls, and programmatically constructed forms, fields, and behaviors.

The Borland DataSnap™ database application architecture of C++Builder was used exclusively. The common API allowed development and testing on single-tier databases and allowed self-contained technology demonstrations during development. When the call center went live and call volume was transitioned, Microsoft® SQL Server™ was used for reliability, scalability, and backup site replication, seamlessly replacing the single-tier test database. However, with the Borland database application architecture, any of the numerous Borland-supported back ends such as Oracle®, IBM® DB2® and Informix®, Borland InterBase®, Sybase,™ and others could have been used with equal ease or could be used in the future. The DataSnap architecture also gives the solution the ability to scale seamlessly without re-engineering to an n-tier system in the future as usage loads increase.

The Borland VCL (Visual Component Library) framework provided extensibility that was crucial to the success of the user interface. Its object-oriented structure allowed for the creation of customized, market-centric visual components. A suite of data-aware components were created by easily customizing existing components to handle the specific needs of electronic funds transfer, allowing better filtering and validation of incoming data, and less degradation of data in the tables.

The WebSnap™ development platform of C++Builder allowed for a tightly integrated, Web-based interface. C++Builder Web components are capable of addressing any Internet-related task such as serving up pages, accessing common services, and dynamically publishing tables to the Web. This helped the development team to easily leverage today's prevalent and burgeoning use of the Web. The customer service application, founded on HTML, benefited immensely by this encapsulation of Web-related tasks into easy-to-use components. An example of note is the use of the built-in TCP/IP of C++Builder, which was used for transaction authorization over IP. What would normally

take approximately 1,000 lines of socket code to implement correctly boiled down to one tenth that size. Incredibly, with the hardware in place, the software worked perfectly the first time out. This is a testament to the strength and quality of Borland's development solution.

Of course, a fully deployed application must have provisions for data input, reporting, user and database maintenance, etc., and according to Ware4 Technology, they could not have delivered all of this in such a timely manner without Borland technology.

The EFT call center results

With today's increasingly complicated software and tight development schedules, prudent software providers have learned to incorporate as many trusted libraries (or "black boxes") as possible while dedicating their precious engineering resources to fit and finish...and delivery! Borland C++Builder dramatically facilitates this approach to rapid application development. The high-productivity rapid application development (RAD) environment, with drag-and-drop component-based capabilities, significantly reduced the project's development time.

The feature-rich C++Builder framework—including COM, CORBA,® RAD database development, Internet components, ActiveX, and a broad range of reusable, extensible, and customizable development components—eliminated many potential integration and QA issues that are typically associated when using technology from disparate vendors.

The new customer service center is a resounding success:

- Average talk time and after-call work have both been decreased, allowing better utilization of representatives' time and reduction in staffing costs.
- The center offers a quicker and more competitive reaction to changes in company policy or market conditions.
- The HTML-based scripting has standardized operations across more than 100 representatives; dramatically raising the metrics on call quality and customer satisfaction.
- Maintenance and infrastructure costs have been reduced substantially.
- Training time (and consequently representative down time) has been reduced.
- The product was also integrated with the telephony switching equipment (Lucent) and the call management system (CMS), which resulted in customer call time being cut by approximately 10 seconds, a significant ROI.

Today, the customer service center processes more than 150,000 calls per month and the volume is steadily increasing. The new customer service software utilizing Borland technology, combined with HTML scripting, has increased revenue and brought about considerable savings in personnel costs.

Borland®

100 Enterprise Way
Scotts Valley, California 95066-3249
www.borland.com | 831-431-1000

Made in Borland® Copyright © 2002 Borland Software Corporation. All Borland product names are trademarks or registered trademarks of Borland Software Corporation in the United States and other countries. • 12898