

## Doug Wilson

Doug Wilson's role at Breakthrough is to assess client operations thoroughly and efficiently, in order to assemble appropriate project criteria. Because he understands the balance between requirements, project deadlines, and cost, Doug works within the client organization to reach consensus on competing objectives.

Before forming Breakthrough Technologies, Doug worked as a senior software engineer in the Diagnostics Division of Abbott Laboratories. He holds a B.S. in Computer Engineering from the University of Michigan.

## Jonathan Monroe

Jonathan Monroe specializes in applying current and emerging technologies to clients' specific software needs. Jonathan asks the right questions to identify appropriate system requirements. He also conducts knowledge transfer sessions so that client staff are comfortable taking full ownership of their new system.

Prior to forming Breakthrough Technologies, Jonathan worked as a senior software engineer in the Diagnostics Division of Abbott Laboratories. He holds an M.S. in Software Engineering from Southern Methodist University and a B.S. in Electrical Engineering from Purdue University.

## Randy Knapp

Randy Knapp specializes in compiling disparate pieces of information into a single, efficient solution. Randy is able to develop detailed designs from both implicitly and explicitly stated requirements. This serves him well when he attempts to define a software solution that meets the needs of internal customers with different objectives.

Before Breakthrough Technologies, Randy worked as a senior software engineer in the Diagnostics Division of Abbott Laboratories. He holds a B.S. in Electrical Engineering from the University of Illinois.

## Handheld Computer Manufacturer

### PROBLEM

A handheld computer systems firm was in the process of developing a digital personal assistant. The company needed a software application that allowed data to be entered into its product easily.

There were several client teams working on different aspects of the project, all feeling the pressure to get the PDA to market as quickly as possible.

While some of the teams had finished their work, other groups were missing deadlines and the development schedule was slipping. Breakthrough was brought in to solve a critical piece of the problem — the data entry user interface — which would allow several of the teams to get back on schedule. Given the lapses in the overall project schedule, the timeline for Breakthrough's development was short.

### SOLUTION

For this customer, time was of the essence. Although we pride ourselves on methodology and process, we understand that in mission critical situations there are no absolute answers.

By deploying an experienced analyst, Breakthrough was able to quickly gather requirements and create specifications in a matter of hours and move into the design and implementation phase after two days. This ability to quickly integrate the customer's values into our development process and still apply best practices to create an optimum solution is a trademark of a Breakthrough engagement.

### RESULTS

The teams whose timelines were affected by delayed completion of the data entry interface were able to enter the data that they needed to test via keyboard.

Rapid delivery of the interface solution enabled the dependent development teams to deliver their projects on time, even though their timelines had initially been compromised.

Development of the entire user interface was completed in four weeks. This enabled two weeks of integration and user testing prior to launch.

Breakthrough's creative stop-gap approach made up the time lost in the initial schedule slips.

---

### TECHNOLOGIES

*Implemented a graphical user interface in Windows CE that simulated typewriter keystroke entry.*

*Characterized system and user behavior using UML use-case specifications*

*Employed Object-Oriented Analysis to identify the system design*

*Utilized Microsoft Windows CE (Embedded Windows operating system)*

*Implemented with C++ programming language*

*Utilized Microsoft Foundation Classes as the user interface technology*

*Used Microsoft Visual Studio, Embedded Toolkit as the development environment*

*Integrated software with client's proprietary handheld computer system hardware*

## Global Healthcare Company

### PROBLEM

A \$13 billion global healthcare corporation hired a 40-person IT team to help them transition to object-oriented technologies for their medical instruments. This IT team was challenged to define each member's responsibilities more narrowly so that the cycle time from analysis to launch could be shortened.

### SOLUTION

Breakthrough developed standard, yet discipline-specific, analysis techniques to streamline the project. According to the model, the analysis team would focus on domain analysis and solution modeling; the design team would take the resulting application objects from the analysis team and map them to an operating system and database. The technique was created to be modular so that it could be applied to future projects.

Breakthrough also coached the IT staff on techniques for gathering necessary business and scientific information from primary stakeholders including hardware designers, chemists, and marketing staff.

### RESULTS

The client successfully launched a diagnostic critical care system that continues to be one of the division's largest revenue generators, available in the healthcare marketplace worldwide.

Each member of the client's Development Team built an area of expertise and ownership that resulted in greater accountability for the completion of their particular phase in the development life cycle.

The majority of the architecture and expertise created for this project were easily applied to development of the next generation of the product.

The software development approach introduced for this project was considered a great success by the client and has been applied to all subsequent medical instrument development projects.

---

### TECHNOLOGIES

*Used UML use-cases to characterize system and user behavior*

*Employed Object Interaction Diagrams to describe complex system interactions*

*Utilized Shlaer-Mellor techniques to create the software architecture for translating application object action language into source code*

*Documented application analysis with the SES Objectbench CASE tool, including objects, states, and methods; all models were OS- and language-independent; Source code was generated directly from models as part of the build process, so the analysis models were always fully in synch with software implementation*

*Managed application persistence with off-the-shelf, object-oriented database*

*Used C++ as implementation language*

*Built embedded development environment on the pSOS real-time operating system*

*Developed PC interface with Microsoft Foundation Classes and off-the-shelf and existing libraries for report generation, serial communication, statistics, and data reduction*

*Utilized Microsoft Visual Studio for application interface development environment*

*Maintained formal source code control using PVCS and defect tracking with a custom system*

## International Computer-based Testing Firm

### PROBLEM

A computer-based testing firm, a division of a \$25 billion international education, entertainment, and publishing corporation, needed repositioning as the leader in several key computer testing areas. These included:

*Content Management*, or the creation, function, and maintenance of components of a computer-based test.

*Data Management*, or the import, analysis, export, and reporting of client registration, test results, performance data, and demographics.

*Content Distribution*, or the means by which tests are distributed and administered, including languages, locations, and proctor options, as well as the e-commerce applications that enable payment for the content.

For this new positioning, the firm needed a redesign and implementation of its test creation, registration, and delivery network. The timeline for bringing the first release of the new system to production was aggressive.

### SOLUTION

With Breakthrough's help, the initial network was successfully launched in nine months. This involved training and mentoring the client's existing IT staff. We developed the methodology and format for a requirements database, instituted a configuration management process with tools, and implemented a weekly build process to track defects and their resolutions.

Following the first phase, the company's test delivery model changed, requiring that tests be available for administration over the Internet. Although this requirement was not included in the original project criteria, Breakthrough's initial analysis was thorough, and the resulting scalable solution allowed for easy adaptation.

### RESULTS

Prior to implementation, the industry standard for setting up an electronic test was 10 business days. Tests can now be published on demand and in real time over the Internet.

The computer-based test development tool is an innovative synthesis of subject matter expertise and software. Business and technology needs were met and the system's features were uncompromised.

New Internet technology was successfully integrated into the system without altering the majority of earlier phase components.

While the old testing network was a stand-alone database that resided on a single PC, the new network is client-server based, enabling multiple, geographically dispersed users to access the same data via the Internet.

The client was able to repurpose product and service offerings to penetrate new and previously untapped market segments, resulting in additional revenue streams.

---

### TECHNOLOGIES

*Employed VBScript, JavaScript, IIS, COM, MTS, Active Server Pages, and stored procedures in SQL Server to ensure maximum scalability for the web-hosted application service in the Windows DNA web architecture*

*Integrated this web system with online e-commerce systems and the customer's own payment-processing systems*

*Built all thick client applications with C++ using the Microsoft Foundation Classes in the Visual Studio development environment*

*Used XML and XSLT technologies throughout when data was crossing system boundaries or stored in discrete packages for access by third party systems*

*Employed Java and Microsoft technologies (including C++, Visual Basic, COM, ATL and MTS) to create client- and server-side components, maximize reuse of software in the system and reduce maintenance cost*