



June 11, 2007

Taking Control of Application Development

One of the sad truths about application development in corporate America is how much of what really happens is a hit-or-miss proposition.

At one point or another, just about everybody experiences the same arcane process when rolling out a new application. The first thing that happens is that some executive agrees to sponsor the application, which then results in months of interviews with business analysts from the I.T. department who are trying to wrap their minds around the everyday processes that drive the company's business. Months of mind-numbing conversation then usually result in a specification that gets delivered to the application developers, who then dutifully code whatever was handed to them.

Then comes the big day when the application gets launched with great fanfare, only to fall back to Earth like a lead balloon because the users of the application can't make heads or tails of the thing concocted by the business analysts, who in turn blame the developers for not understanding what was intended versus written in the original specification. This usually results in developers, with clipboard and stopwatch in hand, spending a lot of time with end users trying to figure out what went wrong with the application, or why the performance of certain application activities is roughly equal to the speed of molasses.

Fortunately, there are two new developments that promise to make the new application rollout experience somewhat easier than the corporate equivalent of root canal. The first is an offering from Knoa, called Experience and Performance Manager, that helps automate the tedious process of gathering data about what the end user is actually experiencing when using an application. The Knoa system provides monitoring technology that eliminates the clipboard-and-stopwatch approach so developers can more readily see what part of the application has an awkward interface or convoluted workflow process.

For example, British Telecom used the Knoa suite to help roll out a customer relationship management system using Oracle's Siebel software, says Stuart Smith, director of CRM architecture and performance for British Telecom. In the wake of pricing deregulation, BT needed to replace a legacy CRM application to have a system in place that could handle 16,000 defined users and up to 7,500 concurrent sessions. And just to make things interesting, BT needed to have the system in place and working by this month or face fines from the United Kingdom's Office of Communications. In other words, the

application had to work right the first time and every time, which unfortunately is the exception rather than the rule when it comes to new application development.

That, of course, brings us to the root cause of the problem the gulf that exists between developers and the people who use their products. There are new types of modeling tools from companies such as Sybase that promise to make it a lot easier for developers to iteratively model an application with end users before writing code. That means rather than being dependent on business analysts to interpret what end users need, developers can sit down with end users and model the application in a series of iterative steps that is more likely to give the end user the thing he needs the first time. Compare this with the current dysfunctional process, which requires I.T. to deliver applications and then spend the next year ironing out the interface and workflow issues that are preventing the people who paid for the development of the application from realizing its potential true value.

This is a touchy subject around a lot of I.T. departments, because it holds the potential for eliminating the role of the business analyst while also forcing a cultural change that would have developers interact directly with end users. The simple fact is that the average developer is a rare breed who just doesn't think like the average person, so it may still be preferable to have some sort of buffer between developers and the end users they serve. Of course, you could argue that the more developers understand about the business they serve, the more likely they will be to come up with ways to leverage applications to drive some real innovation that gives the company a competitive edge.

We could debate the finer points of this issue forever. In the meantime, however, the thing to remember is that for the first time, organizations are gaining more control over the application development and rollout experience, and it's high time they started exercising it.