

E.J.'s Journal

Adventures in Technology

This is part nine of an ongoing series.

BY E. J. SIWEK, CMP

We are now approaching the end of what has been a long road. And this is the end of only the first phase of a three-phase implementation effort. Our overall project goal is to install the OverQuota® system into all our international offices as well as provide access to our field-based consultants. Going into our 10th month, we have reached the testing phase for the installation of the system in our Connecticut headquarters. This is the hub of our activity and updates our international and field staff.

In our case, a small subset of our project team is testing software. This phase will be followed by the acceptance testing phase, which will include our entire team and use live data. In our project plan, we have allocated four weeks of software testing.

In essence, software testing is the period during which you begin to compare what your developers have written with project specifications. Although not a recommended method for others, we continued to develop the software through the initial software testing process, which is typically used to find errors and bugs. It allowed us to understand the product's features and strengths.

In a normal setting—one with a dedicated resource—you would enter sample data to see how the data was being processed. During the course of the data entry, you would note errors and bugs to program developers. The developers

would take those reports and review their work to determine where a coding error may have happened. When the code was updated, the testing would continue to make certain that the error had been fixed. At Excel Partnership, we've decided to add a component of actual development here as well.

Why? Several factors influenced our decision. Some new staff at Excel Partnership were included on the team when we launched the project in January 2000. In choosing the project team, we carefully balanced new and veteran employees. In the course of the first few

...software testing is the period during which you begin to compare what your developers have written with project specifications. Although not a recommended method for others, we continued to develop the software through the initial software testing process, which is typically used to find errors and bugs.

months, it was obvious that we were defaulting to process steps "that had always been done that way." The newer employees really couldn't comment since they were still learning their own jobs as well as the software system. Needless to say, it took time for new employees to share their own knowledge as well as gain enough confidence to challenge the status quo. From this factor alone, it made sense to fold development into the testing phase.

Another factor was the learning curve our IT director experienced over the same nine months. As each week went by, he gained a deeper understanding of how OverQuota worked as well as how our processes could best fit the application. Regardless of how much process mapping you do early on, once you gain a deeper understanding of an application's functionality, you can adjust your processes to take advantage of the application's broader functionality. In retrospect, these two factors, along with our approach to software testing, encouraged further development.

As we moved through the software testing phase, the IT director made it a point to walk through each phase of the program with the process owners questioning them about the look and feel of the program as well as their internal processes. These reviews were with small groups of staff who understood their workflow much better than they had nine months earlier. The result was general openness about what worked well and what could be improved.

During the testing phase, we use essentially phony data so we can review the flow of the program, evaluate the design and feel of the data-entry screens

and shake out any obvious coding errors or simple oversights. In our case, we have broken the testing into two separate trials. First, we walked through user functions. In this stage we determine how one would use the software to process the daily work in a given function. Using a new program forces one to examine the actual steps used to complete a process. From the programming side of development, you can code only what is told to you. If the team member isn't thorough in stepping through the process during the planning phase, the result will be less than perfect. By walking a user through at the functional level, these oversights can become obvious.

The second stage of testing was geared toward testing both the system performance and the application's ability to perform in a network environment. We assembled the primary team members and all hit the system at the same time. We didn't restrict entry into any area of the application. With OverQuota, all members of an account team are constantly kept up to date with an account's activity. The activity could be a purchase, an inquiry or the fact that one of the team members contacted the account by phone or e-mail. At the end of the day, you could see all the account's activities.

In little time of testing at this level, two things became obvious. First, it was apparent we had to be concise in logging information about a customer to minimize the amount of information one had to review just to stay current. Second, we learned we needed to invest in some more RAM for our computers. While manufacturers will tell you that an application will run on a certain amount of RAM, additional RAM will

Regardless of how much process mapping you do early on, once you gain a deeper understanding of an application's functionality, you can adjust your processes to take advantage of the application's broader functionality.

increase the speed of the application.

As we've moved through the testing phase, another issue that surfaced was how the database should be distributed on the network. While the IT department would prefer to keep systems on central network servers to reduce and simplify ongoing maintenance, many applications require sections to be loaded locally on one's root drive. The benefit is much faster access. OverQuota is one of those systems. In a simple test, the access time to open one of the application's databases was improved by almost 75 percent by loading part of the application locally. For the time being, until we are completely through with the development and final testing, we will keep the entire application on the network. However, when we launch, adjustments will be made.

Next month, I will discuss finalizing testing and loading real data for the first time and will look at our training and final outcomes of acceptance testing. ▲



E.J. SIWEK, CMP, is director of marketing for Excel Partnership, based in Sandy Hook, Conn. He is also president of Flashpoint Technologies, LLC, in Bethel,

Conn. Send comments and questions related to this column to publications@mpweb.org.