



Switching to new agile development software and processes in the midst of a major release was a gamble for a developer of corporate performance management solutions. But the risk paid off with reductions in testing and packaging time as well as a nearly 30 percent increase in developer capacity.

Emma Browning sums up her company's problem succinctly: "We were never sure how much 'done' we were at any given time." Browning, director of product development at Acorn Systems' Houston office, recalls the frustrations of having her developers finish a feature for its corporate performance management software, send it into testing, and then have bugs show up that needed fixing only after her developers were already embroiled in designing the next feature.

"It wouldn't be obvious that we weren't done with something, because the bugs would sometimes take a lot longer to fix than we anticipated," she says.

Complicating matters, the privately held firm, with employees spread across the United States and the Ukraine, was using more than half a dozen disparate tools for requirements testing, defect tracking and program management. Along with constantly scrambling to keep information in sync, the company also had to be nimble enough to adapt to the rapidly emerging business intelligence market.

"We have to make decisions and adjustments as the business climate changes," says Torsten Weirich, Acorn's executive vice president of product development. "New opportunities present themselves, other opportunities don't pan out and we have to shift gears," Weirich notes.

Now that the firm has been using Rally Software for close to one year, "We spend a lot less time trying to know where we are," Browning says. The decision to purchase Rally in August 2005 was made by Weirich. "We brought Rally into the company because we had been working with a variety of different software and we'd tried a bunch of different methodologies. Administration was becoming a nightmare," he recalls.

The three big questions that need to be answered on a regular basis in order to manage a software development process are: "Where are we?" "When will we be done?" and "How will we know when we are done?" Weirich kept asking those questions, he says, but "it drove my leads nuts."

While the firm could produce stable, properly built software for its customers, it had a tough time getting information and tracking individual work items. "We could never get an accurate picture of what would happen if we did one thing, or another. The ripple effect that takes place when we said, 'Let's just do this,' was always far larger than anyone expected and it was a huge destabilizer."

The company had a major 2005 end-of-year product release scheduled when Weirich decided to pilot Rally while continuing to plow forward with the old patchwork system. But less than two months into the pilot it became clear that "the product we were trying to bring out wasn't going anywhere," he says. Within two weeks, the Acorn development team had made the full shift to Rally and had 25 developers up and running with the new tool. "Doing that switch midstream helped us meet our critical end-of-year deadline," he says.

While Rally drew some skeptical responses at first, mainly due to the process changes inherent in adopting agile development practices and two-week iteration cycles, it has proven popular with the development team, Browning says. "The thing we really like are the charts that show you a project's burn-down rate. You start off an iteration with an estimated time. As tasks get completed and accepted, the number keeps going down. A bar graph shows you every day the remaining work you have to do and how much has been accepted. If things are going as they should be, your to-do bar is going down and your accepted bar is going up," she explains. "If your to-do list starts growing, a danger flag comes up. Lots of other things show you clearly if the iteration's in trouble and you have to stop and re-plan."

The two-week iteration cycles make it much easier to see whether features approaching release dates will be ready. "You know exactly where that feature is, as opposed to trying to go through the bug list right before a

deadline. We used to get to the end of a long cycle and if the bugs had not been flagged right, it could delay us for weeks or months. And, you had developers being pulled back in to fix the bugs after they were already working on something else very complex. This way, the bugs are fixed within the two-week cycles, so everything's still fresh in their minds," Browning says. "This has gotten us out of the trap of working on something, finding another problem and working on that, and then still having a big mess of stuff that has to be tested at the end" of a cycle.

Another attractive feature of the software is its ability to capture requirements, align them with test cases, and track the progress of both new and existing development projects and defect fixes all in one place, Weirich says. "We can get a picture of where we are and when we think we'll be done based on estimates and available capacity. As part of that, we're also capturing requirements and test cases automated with acceptance criteria."

Where the company used to have three-hour quarterly sessions on requirements, it now has ongoing collaboration with its stakeholders. "Our customers work closely with us as we're adding the new features they've requested. We get validation in that process," Weirich notes.

The agile process also includes a "daily standup call" involving a handful of U.S. developers and their counterparts at the company's testing office in the Ukraine. "Everybody hates having a daily meeting scheduled because they worry about it eating up their time. These meetings are really helpful because they are only five to 10 minutes and we got training on how to keep these meetings running on track. It used to be easy to get off track on one problem, but if that happens now we get the relevant people together at another point in the day," Browning says.

At the end of each iteration, the Acorn developers have begun doing a review where they demonstrate the just-completed features to the entire team, the product owner and interested parties from the rest of the company, including staff from planning, sales and consulting. "That's helping the relationship between development and the other groups because they can see things are progressing," Browning says. "They can see which features are coming along and if they see something they'd like to add, they can talk to us about that. We might have time or we might not, but the point is we're getting feedback more often than just at the end of the release. That really cuts down on miscommunications."

Adoption didn't present a lot of problems, Weirich says, because his staff was so frustrated with all the different moving pieces of the earlier, patched-together system. "The team members have been good about adapting. There were some complaints early on because the agile process was quite different - something of a culture change - and we needed to talk about how to make the process work for us," he notes.

In general, "we've gotten far greater productivity," Weirich says. "Our testing and packaging time was at around 50 percent and it's now down to 15 to 20 percent. That's led to a restoration of confidence in the development team and it's created excitement about the team. It's an interactive, transparent way of managing our priorities."