Software-as-a-Service: The Spark That Will Change Software Engineering?

Greg Goth

Software-as-a-Service (SaaS) is receiving a lot of attention in analysts’ briefings and technology trade press articles. In the past year, SaaS has emerged from its pioneering group of start-ups and medium-sized vendors to be embraced, albeit awkwardly, by software giants including Oracle and SAP.

Much of the attention SaaS has garnered in recent months has focused on the new business model that on-demand software enables. However, some veteran technologists who’ve adopted SaaS for their own livelihood, and analysts as well, say that the phenomenon might well be the catalyst for a far wider-ranging discussion on software development for the next generation. While the chasm between what developers deliver and their customers’ expectations has long been a computing industry bugaboo, pioneers of SaaS see a new era dawning where the obstacles to communication between users and programmers—and sometimes the lines differentiating those roles themselves—are much diminished. A likely casualty of the new way, these experts say, is the long development cycle of the waterfall software-development model.

“I do not feel waterfall development was bad,” says David Norton, research director at Gartner. “It’s given us a lot of software over the last 30 years, but I think its time is up.”

Norton says the rise of the highly interactive Web 2.0 model and iterative development have dovetailed to force even the most traditional programmers to at least consider the end of lengthy development cycles. Citing iterative deployment of software that runs satellites, Norton says, “if they can do it in space, they can certainly do it anywhere.”

Technology and culture driving business

Hub Vandervoort, chief technology officer for Progress Software, says he, too, believes the SaaS wave is one of the “gravitational forces” driving changing development trends as well as deployment models.

“I’ve been around the ASP [application service provider] model, SaaS delivery, managed service delivery, exchanges, and portals since 1997, and would have predicted the change five years from then,” Vandervoort says. “It’s still not entirely here, but we can show much more evidence today than five to seven years ago.”

One major technological factor in advancing the new development models might be the rise of service-oriented architecture (SOA) and Web services standards. The ASP model, championed in the late 1990s and early years of this decade, never took off because its one-to-one architecture was inherently difficult to scale. SaaS technology, however, takes advantage of a one-to-many SOA-enabled architecture that can offer customized services to different customers, and even different branches of the same enterprise. One example is a credit-risk scoring application offered on a SaaS basis by Fidelity National Information Services (http://download.sonicsoftware.com/open/news_events/webinars/062807/index.htm).
The platform can combine, in near-real time, up to 600 customer attributes from 20 disparate data sources, including credit bureau attributes, US Postal Service address histories, motor vehicle license data, and more. It can also customize those attributes on the fly by bank branch or product line. For instance, a bank in San Francisco might have different red flags for account fraud from a bank in New York City. Yet the SaaS-based application can customize those attributes, working off the same code base.

These capabilities are getting a lot of attention from customers. McKinsey & Co., a management consulting firm, released survey results in April from its enterprise software customers showing SaaS and SOA emerged as the two most important trends affecting responders’ businesses (www.interop.com/downloads/mckinsey_interop_survey.pdf). Thirty-one percent of the 850 customers surveyed said SaaS would be the most important trend (up from 30 percent in the 2006 survey) while 25 percent said Web services/SOA would be most important (up from 24 percent in 2006). Perhaps even more tellingly, the survey found almost three-quarters of the group, 71 percent, favorably disposed to adopting SaaS platforms.

Gartner’s Norton says this burgeoning awareness of the new delivery model is fostering discussion of how these applications are built. After a long hiatus, he sees businesses are starting to ask what method vendors are using in building business process management (BPM) solutions.” “I just spent two hours talking with a BPM provider with a SaaS offering,” he reported. “When they first released their core offering, they thought small companies would love it, because it combines BPM, customer relationship management, low license cost, and no overhead. To their surprise, it was big organizations, Fortune 500 companies, that came knocking on their door.”

Vandervoort and Treb Ryan, CEO and cofounder of SaaS infrastructure pioneer OpSource, both say the technological advancements underpinning the new methodologies are being complemented by a new “ground up” ethos that will force academic program leaders and enterprise strategists to retool their own thinking. In fact, Ryan says the shift is a generational shift similar to one he remembers from two decades ago. Just as the young technologists of the late 1980s created both ad hoc and formal transitions of enterprise data from mainframes to PCs and client-server architectures, he says the next-gen architectures of on-demand software are being pioneered by those who have grown up working with instantly available Web-based applications.

“I think what we’re seeing with SaaS is less about how software is going on-demand, and more about how the generation of users who have grown up with the Web as a technology are coming into the workforce,” Ryan says. “And they expect the tools that allow things they’re used to—collaboration, immediate ubiquitous access, and so on—will be there for them.”

“Web 2.0 and socially-oriented computing, as most people think of it, is about Facebook and mashups and things like that,” Vandervoort says. “While that’s a big component of the overall discussion, what I try to do is take those concepts and say, ‘How do I take those ideas, which are incubated in the Internet kiddies’ domain, and put that in real business terms—enterprise quality of service, or levels of security, compliance, audit, control and so forth—that are enterprise-worthy or government-worthy, and still keep all the beauty and openness and free-flowing nature of the Web 2.0 world?’”

Uneasy transition

Gartner’s Norton says the transition to SaaS-based architectures is still in its early phase.

“By 2010, 15 percent of large companies will start projects replacing their ERP backbone with a SaaS offering,” he says. “A little later, Tier 1 consultancies will offer SaaS services, and 30 to 40 percent of vendors offering SaaS service by 2012.”

Norton estimates about half of the Web 2.0 projects visible to end users are still developed using noniterative development methods, but he sees that changing.
"There is a huge chasm, but the methods have to jump," he says. "They are starting to do it, because we are starting to see the first round of pain points with SaaS, where businesses are starting to go out, circumventing IT in many cases.

“They’ve picked a vendor, run with it, and suddenly somebody says, ‘Hey, it would be cool if we could integrate this data with our legacy ERP system,’ and that’s when IT gets the phone call. So we’re starting to see where you have to involve IT right from the get-go.”

However, Norton says he has seen the promise of some flexible projects run aground just as they might become more useful in a cross-enterprise manner, because corporate executives lose their nerve and fall back on old development methods as projects get larger.

“They don’t know what they’ve got, and it’s easier to say, ‘If we put the standard controls in place, we can control this beast.’ They only have the illusion of control.”

In all but the most daring organizations, it will take time to realize that the illusion of control might best be modified in favor of a collaborative, nonhierarchical approach. Vandervoort says the next generation of developers is coming out of universities well-informed of these technologies, but are receiving little to no formal training in how to use them in enterprise settings.

“The shift that has to occur, both in academic training and in enterprise thinking, is to move away from the idea that IT builds the answer for the user,” he says. He sees Web 2.0 enabling IT to shift its thinking toward enabling users to build their own solutions. In doing that, he says users will find their own answer via the path of least resistance, or POLR.

“If people can use instant messaging and mashup and other social networking forms to find POLR,” Vandervoort explains, “IT is no longer building applications and processes. They’re saying, ‘Here’s some infrastructure, build your own.’ But in the enterprise world, IT also has to think about these technologies and provide enough surveillance, audit supervision, and governance. If I was speaking to academic leadership, I’d say train your kids to think this way. They’re already apt to go POLR, now just teach the responsibility that goes along with that.”

OpSource’s Ryan says he looks at his own children and remembers his days as a young computer user, pushing the boundaries of what worked and what didn’t.

“The idea that they’ll be installing software in 20 years is laughable,” he says. “That’s no more likely to happen than I would go back and start using a mainframe. That said, the mainframe is still here, and can teach us a lesson. When the client-server world goes, traditional installs are going to have a very hard time growing. They’ll survive by the same model as the mainframe guys did who learned to adapt and change. Those that don’t will get left behind.”

Cite this article: