



 Print Article  Close Window

From: www.cio.com

Is Your Cloud Project Ready to be Agile?

– David Taber, CIO

February 21, 2012

In the decade since the [Agile Manifesto](#), the movement has encouraged a number of best practices like test-driven development, user-centered design, iterative development, clean code, refactoring, continuous integration, and—arguably—cloud computing. I'm a card-carrying Agile zealot, and to me its benefits are unarguable. (Although here's a [great spoof site](#) that does argue against it.)

There's a catch, though: not every IT organization can really implement Agile, let alone profit from it. There are organizational, project, and personnel characteristics that can make Agile downright dangerous. The awesome price of freedom is that you have to live up to its obligations.

Is your IT organization ready to be Agile, seriously? Score yourself on these questions:

- **Is this the right project?** Be discriminating about where you apply Agile, as certain projects just fit better. For example, if the project has a lot of code that was developed using waterfall methodologies or is inextricably bound to tools and infrastructure that can't be modernized, you should do your learning elsewhere. If the project by its nature must be released as a "big bang" or slash-cut deployment, you'll need to have a very experienced Agile team.
- **Are the expectations for UI functionality and schedule reasonable?** You can't design a good UI with a gun to your head.
- **Is this the right project, revisited?** Is the project already in trouble? Does it have achievable goals? Is the project late, already over budget, and mired in politics? Take a pass on

Agile until things are in balance.

- **Are these the right users?** Agile projects demand a lot from team members, and that goes double for the internal users who are on the team. Are the user representatives flexible yet consistent, well informed yet willing to learn new habits, energetic without the need to throw their weight around? Do the user representatives actually know the business process, and don't have to guess "how would this work in the business?" Do they have an inherent sense of what is technologically within reach given your budget and schedule? Are they personally investing their time in the team's success, or are they likely to point fingers? Think Disneyland: Users must be at least this mature to ride the Agile roller coaster.
- **Can the focus stay on business value?** Agile deliveries mean doing something small and meaningful and valuable, and doing that fast. Over and over again -- to deliver the most valuable things first and to build trust. If your company uses terms like requirements document, cost per function point, or defects per KLOC, Agile is going to be a rough transition.
- **Does upper management really get it?** Some Execs can't stop from "helping" -- scrutinizing weekly progress reports, asking who's assigned to specific tasks, demanding schedule updates mid-iteration, celebrating hoped-for success before they're delivered. They may also think Agile just means they can ask for more features at any point in the project without consequences. Dilbert has a great series on this topic: if you see the boss [here](#) and particularly [here](#), you might not want to push Agile quite yet.
- **Does finance have the right headgear?** If the finance folks are talking about value and investing to maximize business results, you're on the right track. If all they can talk about is fixed budget, defined deliverables, and the words "compliance" or "variance," you haven't got a chance.
- **Does your IT team have the horsepower?** Agile is hard, and -- let's face it -- requires considerable brainpower. Companies that really are Agile at scale -- firms like ThoughtWorks or Salesforce.com -- have staff that are smarter than the average bear. And it isn't just smarts: it's an attitude of "deliver or die," including the occasional all-nighter for the sake of solid, clean code. Developers need to have a good visual sense, be flexible, and focus on delivering the smallest amount of code to satisfy the need. Agile developers don't just listen passively to the users -- they add value to the conversation. Agile-excellent firms tend to have a knack for hiring well...and when they don't, firing fast. Be honest with yourself when scoring this one.
- **Do you have the right process maturity?** Because cloud computing must span several vendors' domains, developers will have to leverage several languages, libraries, and layers of infrastructure. And that means that the available infrastructure is fragmentary, at best. Whether it's debugging, configuration management, deployment, or error logging, your team will have to roll a lot of their own operational infrastructure. So when it comes to internal processes for parallel development, continuous integration, deployment validation, and real-time troubleshooting, your team is going to need a lot of discipline and good reflexes. Without these, Agile only means chaos. Again, be honest with yourself when scoring this one, as it's not an area where you want to do a lot of On-The-Job-Training.
- **Does the team have the right attitude about what's being developed?** It's OK to make a UI that's "good enough," particularly if you do it under budget. Agile teams assume that UI code is disposable, and may have to be completely replaced in 18 months even if everything is right today. For most UI projects, perfectionism simply doesn't pay.

Moving to Agile takes a mix of energy, skills building, internal trust, investment, and patience. Agile provides a huge strategic advantage over the longer term, but only if you give it time to succeed in the shorter term. Pick your project carefully, put some of your best folks on it, give them the room to learn...and to make a few mistakes. Try to convince the execs to help less. If this doesn't sound like your organization and project, reconsider.

I'd like to thank Bo Laurent and Rich Mironov for their excellent contributions to this article.

David Taber is the author of the new Prentice Hall book, "[Salesforce.com Secrets of Success](#)" and is the CEO of [SalesLogistix](#), a certified Salesforce.com consultancy focused on business process improvement through use of CRM systems. SalesLogistix clients are in North America, Europe, Israel, and India, and David has over 25 years experience in high tech, including 10 years at the VP level or above.

Follow everything from CIO.com on Twitter [@CIOonline](#)

© 2012 CXO Media Inc.