

IBM Podcast

[MUSIC]

MATHENY: Welcome to this IBM podcast, Top 10 Software Delivery Secrets, Do More With Less. I'm Angelique Matheny with IBM. Providing more value with fewer resources -- easier said than done, right? Every organization today is looking for ways to do more with less these days, and only the companies that implement the most effective methods will survive.

In this podcast, we'll discuss specific secrets of delivering more valuable software to customers despite having fewer resources. We'll include secrets related to process, development, infrastructure and information. And it's my pleasure to introduce George DiCandio, Distinguished Engineer, Development Manager, Requirements Definition and Management. Hi, George, welcome to the podcast. Thanks for joining us.

DiCANDIO: Hi, thank you. It's my pleasure to be here.

MATHENY: George, let's go ahead and get started. You know, everyone has a different take on what does more with less mean. Is it really possible to do more with less?

DiCANDIO: Well, yes, Angelique, it really is. I know it sounds like a cliché, but in tough times like these, it's even more important for software delivery teams to reassess how they do business. They need to look at tuning and streamlining their processes, automating their tools and just generally eliminating waste.

You know, I found throughout my career in software development focusing on continual improvement of my teams' software delivery process can pay off many times over. You know, don't get me wrong, it's not easy. It can often times means slowing things down for a while while you get your bearings, but in the end it's always worth it.

MATHENY: And George, how does this relate to software delivery?

DiCANDIO: Well, when you're looking at improving in software delivery or doing more of it with less, I found it's really useful to look at four key areas: improvements in process, improvements in development, improvements in information and information you gather about your processes and your development, and also infrastructure.

If we take a look at process, this is really meaning the process that you're using for software delivery all the way from requirements definition and capture through code

development, test and deployment. Without a smooth and efficient process, it's really easy to have a lot of waste and burn through a lot of excess time and money.

Development is just what it sounds like: it's focusing on the development of code. You know, are you using the right techniques and tools? Do you have the right expertise available? Are you doing proper reuse? Are you building quality in from the start? These types of areas are really...look at streamlining when you talk about development.

Information is something that's often times overlooked. You know, are you gathering the right information so that you're getting the right feedback from your development processes?

You know, are you understanding the project health at the right times?

Are you seeing the right metrics in a timely manner? You know, are your defect trends going the right way? Are you testing your requirements instead of just the code? These types of things. Getting this right information, measuring it, can often times make big differences in how efficient your processes are.

And the fourth thing, also often overlooked, is infrastructure. When times get tough and you're really

under the gun to deliver, often times the support that you give your development teams as far as both having the right tools, having the right infrastructure set up so that they can get builds, do builds, all of this comes into play.

And I think in looking at these four key areas -- process, development, information and infrastructure -- you can really look at trying to optimize your development processes and doing more with less.

MATHENY: So, George, what secrets can you share with us today?

DiCANDIO: Well, I've got a few secrets that I can share, I suppose. We'll go through each of the areas, and I can give you some of my views on some of these things and talk a little bit about how some of the Rational tools can help some of this. So, I'll highlight a few in the process area.

One of the problems with process often happens when teams and projects grow really large. So, one technique that I found really works well is breaking larger teams into smaller teams. This may seem obvious, but if you look at doing this it really decreases the number of communication paths, especially if you put clear leaders in charge of each team or task.

So, for instance just take a relatively small 10-person team. If each person has to communicate directly with each other, that's, you know, an overwhelming number of communication paths -- 45 to be precise. But if you break that 10-person team into three sub teams, you've eliminated this to only three communication paths between the teams.

This same technique can actually work with tasks as well. So, taking large, sometimes seemingly insurmountable tasks and breaking them down into smaller milestones, giving the team concrete goals to focus on. So, these are some techniques sort of breaking your process down, and Rational with its Jazz platform can really help deliver some benefit around helping you break your tasks and teams down.

And the Rational Jazz Platform can really help in breaking down teams and tasks. All of the Rational Jazz-based tools have team features like planning, collaboration, transparency, to help teams communicate more seamlessly, to help to define teams, to help record communications between teams so that people can see what's going on within the different teams no matter where they're located, in what time zone, or what their role on the team is.

Another process tip is to keep it simple. Often times we see customers come to IBM or to Rational, and they're looking to completely revolutionize their process and to

roll it all within a very short amount of time.

Overwhelmingly we see that these types of initiatives are really doomed to failure.

And the best changes that people are able to make is to really start small, maybe change a single step that they think will have some huge impact or potentially high value, and implement that, see the results. The team starts getting motivated around these results and they get excited about actually seeing more results and making more changes.

One important way to ensure that you're seeing these results is to make sure that you're measuring the right things. And this is again, where tools and the Jazz-based products can help. All the products have dashboards built into them so that you can measure progress, keep your teams on track and get the business result that you want.

So, when process comes into play it's really just some simple tips: breaking large teams into smaller teams, large tasks into smaller milestones, and using the tools that you need in order to track this. And then, keep it simple.

MATHENY: George, those are some really great tips on process. Do you have any for development?

DiCANDIO: Yes, as a matter of fact I do. I've been in

development almost my entire career and I've also noticed a few things in the development area. So, first thing that is often overlooked is really building quality in from the start. Quality starts with requirements and well-defined requirements.

In fact, I've been reading recently that 71 percent of all projects that fail, fail because of poor requirements. Teams just aren't focusing enough on getting the requirements right in the first place. Rational has really been focusing on this area and has been introducing a number of new products to help teams get requirements correct.

One of these products, also in the Jazz platform, is called Rational Requirements Composer, and it allows teams to really define their business goals then decompose them into concrete project requirements making sure that their requirements are always in support of their business goal and allowing them to track back to those requirements.

It also allows them to elaborate with visual storyboards, use cases, glossaries, so they can more easily see what's going on with their requirements, define their requirements and share their requirements with the stakeholders before they actually take the time and go implement them incorrectly.

The second tip in development is really leveraging your team. So, if you're like most development shops today, you have a fairly complex development environment and you have a large diverse set of skills on your team. It's probably the case that your team is spread out across the world. And your success is really determined by your ability to leverage these team members and their skills to matter where they're located.

You know, certainly traditional methods like telephone calls and sometimes even videoconferences and net meetings can work. But when you're doing development, Rational's Jazz platform can really help bring the team closer together.

We have tools like Rational Application Developers Collaborative Debugging features that allow one team member to suspend the debugging session and have it picked up by someone else on the other side of the world. We have the collaboration features in Jazz like Team Central, custom RSS feeds, taggings, pervasive commenting throughout...

Features like this that really help teams collaborate and leverage each other's expertise no matter where they're located. So, leveraging your team, another really important tip.

MATHENY: George, you covered a couple of the categories

you mentioned. What about information?

DiCANDIO: Yes, so, information is all about getting the information that you need to make the right decisions. And this is no different than what you might do in the stock market. You really want to act like a spy on your development processes. Figure out what the right measurements are and start measuring them, start examining them, make them visible to your team.

So, Rational tools also have some capabilities that can help you gather this information and figure out what information to gather. So, the first thing you might want to look at is what we call MCIF or the Measured Capability Improvement Framework. This really talks about processes that you could be following and what the proper measurements are to track to make sure that you're doing those processes correctly. So, they give you a head start on figuring out perhaps what you want to track.

Another product that Rational has that's really exciting around gathering information is the Rational Insight Product. This is a data warehouse for Rational Development Tools. It's powered by the powerful Cognos Reporting and Data Analysis Engine...

So when you report off the information that Insight gathers

from all the Rational data development tools, you're going to have the ability to organize that and present that any way that you want. It's really about measuring twice and deciding once. You want to get the information that's...all the information that's available to you before you make a decision.

MATHENY: And George, our last question today, where can we find out more?

DiCANDIO: Well, that's a great question, and there's a lot of more information about this topic on the Web. We have a Software Delivery Best Practices white paper called, Do More With Less, and the link to that will be included in the show notes.

MATHENY: It sure will, you're right. And George, thank you so much for sharing your time today to discuss software delivery secrets, do more with less. We really appreciate it.

DiCANDIO: It's been my pleasure, thank you very much.

MATHENY: That was IBM's George DiCandio, Distinguished Engineer, Development Manager, Requirements Definition and Management. If you're interested in more podcasts like this one, check out the Rational Talks To You Podcast Page at

www.ibm.com/rational/podcasts.

Like I said, we'll include the link to the [e kit] Software Delivery Best Practices, Do More With Less that holds the white paper George mentioned, so check it out. This has been an IBM podcast. I'm Angelique Matheny. Thanks for listening. Keep tuning in as Rational Talks To You.

IBM Podcast

[MUSIC] [END OF SEGMENT]