IBM

Moderator: Angelique Matheny
November 18, 2008
12:00 pm CT

Operator: Good afternoon my name is (Cassandra) and I will be your conference operator today.

At this time I would like to welcome everyone to the teleconference. All lines have been placed on mute to prevent any background noise. After the speakers remarks, there will be a question and answer session. If you like to ask a question during this time, simply press star then the number 1 on your telephone keypad. If you would like to withdraw your question press the pound key.

And now I would like to turn the call over to Angelique Mattheny. You may begin.

Angelique Mattheny: Thank you very much (Cassandra), hello everyone and welcome to the Rational Talk View Teleconference, Quality in Action Using Rational Quality Manager.

I'm Angelique Mattheny with IBM Rational and I'll be your host for today's call. I want to make just some quick introductions. Joining us today is Brian

Bryson, Brian is Technical Evangelist for IBM Rational Quality Management. Also Matt Holitza will be with us today, Matt is Marketing Engineer for IBM Rational Quality Management.

Today’s discussion is about the integrations with and leading practices for our new software quality offering - Rational Quality Manager. A Web based software test management environment that enables teams to deliver greater quality software, and prove operational efficiency and continuously lower risk.

Now you want find any slides for this teleconference, this calls are really for you. We want this to be interactive and this is your chance to get your questions answered directly from our experts and discuss what’s on your mind.

As the operator mentioned you should press star 1 and she will open up your line at any time, it doesn’t have to be at the end of the presentation, so if you have a question don’t be shy.

Also if you’d like to submit questions to our panelist after this teleconference please email us at Askusnow.US.IBM.Com, that’s A-S-K-U-S-N-O-W @ U-S.I-B-M.COM. Just but the title into teleconference in the subject line.

Well I think you’ve heard about enough from me so let’s get started. Brian I’ll turn it over to you.

Brian Bryson: Thank you Angelique and thanks everyone for joining today we got a fairly decent crowd which is kind of a surprise to us I would say since we have such a new product here.

We didn’t really know what to expect from this mornings call we haven’t done a lot of these teleconferences they are kind of a new event for us.

I guess the idea of what we’re trying to accomplish here today is to just have an open forum where if you already actually been using the product - now grant it it’s only be out a few weeks.

But if you got some questions on using it we’d like to address those or just some - just looking for some basic starters. We’re going to be glad to provide that kind of information.

And otherwise just have a very simple open discussion about two things really today, I guess the focus of what we’re talking about is Rational Quality Manager and Rational Test Lab Manager.

So as Angelique mentioned by name is Brian Bryson I’m with the product marketing team. And my partner today is Matt Holitza Co-Worker as well with the product marketing team.

And I guess what we have is - I guess to start off with is - as an opening comment you know, Rational Quality Manager and Rational Test Lab Manager are very new products offerings from IBM Rational .

We - the official release is October 28th so we’re - let me quickly look at the calendar here we’re looking at maybe three weeks or so ago just before Halloween.

They are addressing some new and - new and - you know the needs that they are addressing. One is really the need for basic test management and quality management, and that’s what Rational Quality Manager was developed for.

In the past we've had products that address this need you may be familiar with Rational Test Manager, you may be familiar with the ClearQuest Test Manager.

They are all sort of previous technologies to address this need but you know, based on feedback from those tools we've started in a new direction technically speaking from an engineering prospective on where we want to go with Test Manager.

You know the primary feedback we got on both of Test Manager and ClearQuest Test Manager of our previous test management offerings, was that we need - our teams today - and if I sort of paraphrase and generalize. Team quality teams today are generally more separated geographically distributed.

Often that meant you know, just across cities. Sometimes that meant across the country. Sometimes that meant around the world. And what was effectively required to sort of unify and get that team to collaborate and work together was some of solution that was Web based but that it also integrated a high degree of automation with the current tools.

So we really need you know if I sort of simplify this down to the common denominator. You know, the feedback we got from current customers and prospective customers looking at the tools was - we need our tools to be really collaborative and automated - automated in the sense you know those integration to you know, automation tools such as you know, functional test or performance to actually run test.

And a tool that gave us a very strong reporting capability so that we're able to sort of keep track, govern, manage however you want to put it, our testing (products). Let us know where we are, how much it's left to go how we're progressing with our past salaries etcetera.

So you know, from that the Genesis was Rational Quality Manager, we released it October 28th, it is a Web 20 based tool. Runs on a sort of a clinical server architecture where you're accessing a server via browser, Internet Explorer or Firefox.

And you can get to it from anywhere, so it kind of addresses that first distributed need where teams are spread out you know, across cities across different buildings even, across - or even across the globe.

It's built upon our jazz - our jazz server architecture, and I don't want people to get too confused about this. But that point is significant in that the jazz architecture is a new direction in technology for IBM Rational.

The idea with the jazz architecture is that all tools, and we're not talking just testing her, we're talking about development architecture, modeling, requirements. All of our tools should better integrate and better share information.

Now if you've been a long time Rational Customer you'll realize that we had various ways of sharing information between tools. You know some of the most basic import/export functionality to a little bit probably you know, at best sort of direct API Integration where you know, what tool would pass specific information to another.

And you know over the years that has served us well and served software development team as well - you know, it gave it us that base level of data sharing.

But what we came to realize is you know, our architecture was really limiting what we thought we needed to do to get teams to be - to better work together to better collaborate, to better share information and data.

So we started from the ground up and we've built what's called the Jazz Foundation Server. So the Jazz Foundation Server essentially is a server architecture that's going to underline all of our tools.

Quality Manager is one of the first on there, there is two others that are currently available one is the Rational Team Concert which is a tool for work item tracking and defect tracking, version control. And the other is Rational Requirements Composer which is a tool for - better managing the elicitation process of requirements, so making sure that - not so much the management of requirements. But sort of you know, what's been done, what hasn't been done tracking aspect.

But the aspect is sort of better elicitation of requirements, better capture of requirements. Provide sort of a graphical interface for you to work with your customers, or to work with your stakeholders to better you know, understand what they need the system to do.

So yes, there is a modeling aspect to it, there is a storyboarding aspect to it, there is a screen mock up aspect to it.

And it's a tool that you can use to better capture and better illicit requirements from your stakeholders.

So you know the functionality of it aside you know the key point is that's it built upon this new foundation, this new Jazz Foundation, this is really a new direction from Rational that's really going to tie us all into one base product which is a great thing. So it's really going to enable sharing of information across teams.

So Rationale Quality Manager is the third Jazz Product that was realized. And it's you know, quite simply it's a tool for test management. It allows you to create a test plan. It allows you to link that test plan to your requirements. It allows you to create test cases, it allows you to execute those test cases. And be execute those test cases I mean, it allows you to launch or functional test type, it allows you to launch (unintelligible).

It allows you to write - to write and execute manual test. So for you know, the majority of testing that is manual it's actually an interface in there to do manual testing and then it lets you view all your results.

So that's kind of the basics of Rational Quality Manager there it's a new tool for test management built on Jazz it's really meant for (unintelligible) teams to better collaborate, automate and really manage or govern or report on their software projects.

Now, what I'll do here is maybe I'll pass it off to Matt because what I - maybe we can do here Matt is just talk a little about what is Rational Test Lab Manager the second item on the agenda - or not that we have a formal agenda but the second tool that we're really talking about today.

And maybe after that we can dig a little bit deeper on the - you know, some of the stuff you can do with it. So Matt why don't you see if you can give us a little bit of a overview of what really is Rational Test Lab Manager.

Matt Holitza: Sure thanks Brian so Rational Test Lab Manager is an extended component of Rational Quality Manager. And so Brian mentioned the fact that you can tie requirements and test cases to your test plan.

Well, Rational Test Lab Manager also allows you to tie your actual lab resources and environment configurations to your test plans as well so you really have a holistic test plan.

And Rational Test Lab Manager the core functionality of it is that it allows you to inventory all of your test lab assets in one consolidated place. It allows you to store all of your inventory of your physical your virtual machine that you run test on.

And it actually does a whole lot more than that too because once you have those - once you have the inventory in place then you can make that visible to your entire team. And your entire team can then make a request of different la resources, reserve different lab resources for different periods of time.

And this allows the team to see in one place where all the lab resources are, who is using them allows Test Managers to see what needs do I have in the future because I have request for lab resources in the future.

And it keeps the team on the same page instead of having spread sheets spread out everywhere, people are owning different lab resources.

So then beyond that it also allows you to execute your test on these lab resources. So now you can not only you know, schedule and request your lab resources but you can execute all of - of course all of our testing tools - functional tester, performance tester, service tester, (app) scan tester addition.

And even test real time you can execute test using Rational Test Bud Manager.

And then we took it a step further even Rational Test Bud Manager also embeds Build Forge. And what Build Forge allows us to do is actually go out to lab resources and pull back the information about that lab resource. So operating system - any software that’s installed on it, what hardware it’s using, so that testers now have the capability of searching through the inventory based on configurations.

What Build Forge also allows us to do is now deploy testing tools or even software builds to these test lab resources. So if you wanted to run a functional tester test on a Lab Resource A for example, and Lab Resource A happens to be a virtual machine doesn’t have functional tester on it, well that (unintelligible) Lab Manager can go out and actually deploy that tool to that machine, and then run their functional tester test on it.

So it allows you to more automate some of those processes that you know, have not been automated in the past. You know, we’ve done automated testing, performance testing more recently security testing. But now it’s actually automate the deployment of testing tools to lab machines.

Test Bud Manager allows you to do some capacity planning so you’re a test manager and you want to know you know, where are the - what configurations are more often used by our team. And so next year what lab resources do I need to acquire or plan for. And now I can optimize the resource usage of my lab and really save money in so doing so I don’t have a lot of lab machines laying around that aren’t being used.

So really from a high level that's Rational Test Bud Manager and you know, now Brian should we open it for questions or you want to go into more detail?

Brian Bryson: Yes, so we have - so there is a couple - two ways to go here, what I'd first like to do is - you know, there is obviously a fair amount of people that have dialed in and you know, it could be that you're just here to get the basics and that's you know, perfectly cool.

But if you've - maybe you've been on the Beta program, or maybe you've actually started using the tool, or maybe have some specific questions. I'd like to before I go into some of the stuff that I have prepared for today, just see if there is any immediate issues that we wanted to get into.

So it's - to actually get on the phone and participate in the discussion it's just a simple star 1, that'll put you through to the operator. And you know operator feel free to interrupt at any time if anybody does come in even if it's as move forward here. We'll you know, we'll stop and take those direct questions.

Now...

Operator: We do have a question now...

Brian Bryson: Oh, great let's get right into it.

Operator: Okay you have James Leone.

James Leone: Hey there.

Brian Bryson: Hi there I missed your name I'm sorry could you repeat that.

James Leone: James Leone...

Brian Bryson: Oh James.

James Leone: ...from St. Judes Medical.

Brian Bryson: Simple enough how you doing James?

James Leone: Pretty good, I have a question for you, one of our big problems - and this question pertains to Rational Quality Manager.

Brian Bryson: Mm-hm.

James Leone: A big problem is we commonly have concurrent releases going on parallel. And when it comes to test case management, has does Rational Quality Manager provide a branching in a sort of the test cases.

So if I have the same test case...

Brian Bryson: Mm-hm.

James Leone: And in version 1.1 it goes through certain steps, but in version 1.2 there is a new feature, so that test case needs to be altered a bit. But we don’t want the 1.1 version altered.

Brian Bryson: Sure.

James Leone: How do you handle that kind of parallel situation?

Brian Bryson: So there is like a couple of ways you can handle it, I guess first off - there actually is versioning capability in Test Lab Manager. So you can have - I mean Test Lab Manager - in Quality Manager.

James Leone: Okay.

Brian Bryson: So you can actually have versions of your test plan and versions of your test. Now that's one way of doing it and - it would help getting into some of the you know, specifics about how you run everything. It's one possibility.

Another possibility and I think this might - is probably the more common way of approaching the problem is to use configurations. So built into Quality Manager is this concept of configurations, and it's not really designed - or wasn't originally designed for addressing what you were talking about.

Originally it was for providing different test cases to run on different platforms. So if you had a different test case you know, for Windows '98 versus Windows XP versus the Lynx Box or whatever. Or maybe a different version of the test to run on - a machine with a you know, a different processor of a different database or something that defines that environment.

That was kind of what the design was for but you can also tweak that so that you have different test cases. Different versions of test cases kind of perversion of your applications, so you can have a version of a test case for you know, Version 1.1 of your application, Versions 1.2 of your application. And tie different test to it that way.

So it's not originally the original design but that's kind of one of the things with Quality Manager is that you can - you know the architecture is open enough that you can do things like that, and have different test cases for different versions.

So you - it kind of depends but you definitely have two possibilities there of ways you can do that.

James Leone: Okay so if you have the version I assume one person is going to be working on the older version and then the other person will be modifying the newer version?

Brian Bryson: Mm-hm.

James Leone: Can you work in that kind of set up?

Brian Bryson: Yes.

James Leone: Okay, now when you’re talking about the test and what - are you talking about like an automated test are you talking about a manual type scripted test where you have like a list of Step 1, Step 2, Step 3?

Brian Bryson: Well at the current time we create what’s called a test plan which is a verbal - I guess a kin to a manual set.

James Leone: Okay.

Brian Bryson: And eventually someone will convert that into a script, a lot of this is just Legacy and dealing with the FDA and I think it’s just kind of the way things have been type situation.

James Leone: Sure.

Brian Bryson: So it’s good to have that plan English, here are the steps, here is what we’re doing.

James Leone: Mm-hm.

Brian Bryson: As opposed to looking at a script from whatever is functionally executing the test.

James Leone: Yes - so yes, I guess that’s - I’m trying to figure out which way is the best way for you to go with versions or configurations.

Brian Bryson: I’m actually playing in the tool right now, it’s unfortunate we don’t have a Web sharing aspect of this right now. But we’ve got - we got a concept of execution records.

Matt Holitza: Yes, this is Matt. The one thing I would think of is that you know, if you’re trying to you know, for compliance reasons, the test execution records might be the way to go. Just because then you have a record - you know, it keeps a record of all that information every time it’s run on a different version. So that might be you know, one consideration there.

Brian Bryson: Yes now that’s going to give you the reporting side...

Matt Holitza: Yes.

Brian Bryson: But now since we’re talking about something more of a manual test, could you attach a different manual test - because you got you know, two developers working on two simultaneous versions of the test and that’s fine.

Can you - yes you can - so what you get James is effectively your - you've got two parallel versions of the test. So it's not going to be a complex branching of you know, of one test that has a branch that says, if this is Version 1 do this step, if this is Version 2 do this step.

You'll have two parallel copies of the test tagged by version, or you know, I'm saying version to mean you know, which version of the application of whatever you're testing.

You tie that into the same test case, and then the tool you know, you'd identify it runtime, which one you wanted to run based on the version number. You would have two parallel test.

James Leone: Okay. Okay, and just one final quick question.

Brian Bryson: Yes, sure.

James Leone: RQM is phasing out ClearQuest Test Manager is that correct?

Brian Bryson: Well you know, well yes and no. So no end of life has been announced for Test Manager of ClearQuest Test Manager for that matter. So for customers that are using it you know, keep it at. You know if that's - if you're happy with it and it's doing that you need and you're getting the results you want, you know, we have no plans to eliminate any of these things.

But that's it, you know realistically from a engineering prospective where we are investing our dollars and you know, what are recommended test management solution is moving forward is Rational Quality Manager.

So that means you know, don’t expect a lot of enhancements for any of the other tools. And you know, all of the new features and all of the new technologies is going to show up first in Quality Manager.

Now if you are - if you are a current Clear Front Test Manager User, or Clear current Test Manager User, there is migration utilities to get you into quality management. So there is no loss of data or having to rekey anything.

But yes it is our new direction we’re not eliminating any of the old stuff - but this is the way - you know, this is the new path that we’re heading on.

James Leone: Okay very good, and how do I go back on mute?

Man: That’s a good question.

Operator: (Unintelligible).

Man: You can’t.

Man: You’re stuck with us now.

Man: You’re stuck now do you know any good jokes?

Operator: I’m closing your line now are you finished?

James Leone: Yes, thank you.

Man: Thanks operator.

Operator: And there are no further questions - you’re welcome.

Brian Bryson: Is - there is no more questions that what you said?

Woman: No further questions.

Brian Bryson: Perfect, well so here is what - and feel free to interrupt any other time with additional questions - anyone.

I’ve got you know, seeing as Quality Manager is kind of a new thing, there is a few things about it that I find - people - you know, people come up to me and say you know, show me Rational Quality Manager.

And there is three or four or different things about the tool that I wanted to just talk about today. As just to you know, throw different basic ideas going. Because these are things I think Quality Manager does kind of well and needs are interesting that don’t show up you know, in our other offerings or in fact many of the other test management tools that are even available on the market today.

None of these things I’ve got kind of listed and prepared here are earth shattering. They are just nice little things that Quality Manager happens to do, little tidbits of you that Quality Manager does that make it kind of a neat tool.

So, first of all I’ve got probably listed - I don’t know I got one, two, three, four, five, six - I got six or seven things here that are kind of neat that I wanted to talk about. Two of them - you know, because we couldn’t do any live demonstrations today.

What I’ve done is I started to actually record some little mini demos of these small little features they you know, minute - minute and half type demos.

And I’m starting to put them on YouTube so if - for the first two anyway I just had time to get the first two up over the last couple of days. If you go to a YouTube.com/Rational Tester, that’s our test channel if you will.

And at YouTube.com/RationalTester you’ll see two new videos that are called tidbits. And that’s exactly what I was going to talk about today was some of these little tidbits that make Quality Manager kind of neat.

So feel free to flip over and have a look at those right now. The first one I was going to talk about was the duplicate data tidbits if you will. And the idea here is this is actually - this actually comes from one of the design principles that went into Rational Quality Manager.

So one of the - you know, one of the neat things about working for IBM is it’s such a huge company and there is a lot of assets that we can leverage when we’re putting products together. Two of the key ones is that there is a big research and development arm at IBM Rational that we basically steal ideas from and turn them into products.

And the other is the huge consulting workforce that we have as (unintelligible) or global business services, global technology services, global services consulting etcetera.

Something like a 150,000 consultants - IBM consultants that go out there everyday and do consulting. Now obviously not all them do testing, but there is a specialized testing branch. And one of the things they came back - you know, when we were designer Quality Manager one of the things they came back to us with, was the conflict of duplicate data.

They had a done a study in 2005 to 2008, so it’s a four year study. Where they went across hundreds of projects and they tried to figure out what are the number 1 - you know, what are the best things that you can do to make a testing project more efficient.

And the number 1 thing across all the projects that they looked at in this four year study to improve the efficiency of a testing project - of a Quality Manager Project was, to eliminate duplicate data. So that means not entering in duplicate requirements you know, kind of boiling it down. And not having duplicate defects.

So it’s not directly you now, affecting testing you know, we’re talking - we’re not talking kind of before the requirements and the after - the defects of the testing process - right?

And the issue was the more duplicates you have, you’re doing duplicate processing, you’re wasting time and effort because you’re - in the case of requirements you’re writing down things that are already known.

In case of defects you’re reporting problems that are already known. And you know at some point you are going to have to consolidate these are down and match them up and tie them up.

And it turns out that it was a you know, a huge waste of time and effort to you know, to get this duplicate data in there. So you know, one of the things that they ask us for is - can you make - find someway to prevent duplicate data from actually getting into the tool in the first place. And we’ve done that with the tool.

And it's not - it's one of these features that doesn't necessarily pop out to you as you're using the tool. It's not something that you know, that you can go over to the Website and read the marketing material, I don't even know if it's even listed to be honest.

But, it's one of the most powerful things in the tool to make you more efficient. So the idea with the duplicate data is when you go to create any type of data in the tool for a requirement, a work item, a defect.

And it's - and defect is a great example, you know you find a problem, you type in a defect. One of the things you can do is there is a little button that says - that you click on it says find potential duplicates.

And what Quality Manager will do it'll go through all the existing records of defects and requirements and work habits, and whatever it happens to be - whatever happens to be that you're creating. And it looks for similar phrases, similar tags to what you're about to enter.

And then it'll give you this list and you'll be able to see if there is anything that sorts of the ballpark of what you're about to enter so that you don't go reporting a problem that's already known.

You would instead go you know, modify that record and say you know, this also happens when etcetera, etcetera.

So the duplicate data is one of the simplest most powerful features of Quality Manager. And that's if you go to YouTube.com/Rational Tester, you can actually see a quick demonstration of that, where I show you can prevent yourself from entering any requirement in that case.

The second one is - that I wanted to talk about was requirements updates. This one is a little bit more prevalent when you look at the tool you probably wouldn't miss this.

But again there is - it's one of the ones that I wanted to bring up and it's the other one that I was able to create a video for on YouTube. So again YouTube.com/RationalTester, you'll see a tidbit about changing requirements.

And the situation here is if you're doing requirements based testing, and I think the majority of people out there are. If you're doing requirements based testing, it's really important to know if you're requirements - you're underlying requirements are changing.

So, how does Rational Quality Manager let you know that, and there is really three ways. First off is the dashboard, so if you want to see just your basic requirements coverage you'll see on the dashboard and there is - in the video you'll actually see it.

Around the dashboard the first thing you see when you log in is what requirements are covered which are not. And you know, just a very simple yes/no pie chart, one part is green, one part is red. More green better, more red bad.

You can even click on that pie chart on the dashboard and drill down and actually see what is and what is not covered.

There is also some reports so there is - I don't remember exactly, I think there is 7 or 9 different reports where you can actually get perspectives on requirements coverage.

The third is right in the test plan itself, so what's unique about Quality Manager is our definition of a test plan. I think most people when they think about what is a test plant, they think about a word document that's written at the beginning of a testing project, and then you know, never to be looked at again.

Quality Manager has a little bit of a different definition of a test plan, in that the test plan contains that same information you'd put into a word document like you know, what are we trying to do, what are we trying to accomplish etcetera, etcetera who is on the team all that stuff.

But the test plan also has dynamic links to all your requirements, and all your test cases. So at any point if you wanted to look at test plan and say what is this test plan actually testing, what requirements are being validated you can bring that up even when requirements are being added and removed. What test cases are being run as part of this test plan, so you have that information.

Now, when a requirement changes the test plan is aware of this, and you can look at all of the changed requirements for a test plan. You go into the requirement section of the test plan, you can sort by status of the requirements, and bubble up all the changed requirements to the top. And then you can click on that requirement to drill down and see which test cases are impacted by that change.

Straight from your test plans you have a really dynamic test plan that gives you a lot of information. So requirements changes one of the nicest things that Quality Manager does.

Let me give you one more and then maybe Matt I'll throw it back to you to talk a little bit about Test Lab Manager so we don't get stuck on one thread here.

And again at any time star 1 with any questions and we'll you know, we'll tangent down whatever - you know, what alley you want to go down.

So let me talk about one more thing about Quality Manager and then we can talk a little bit about Test Lab Manager for a second.

So, one of the other neat things in Quality Manager that is probably sort of a yes - duh, situation to any tester, but it's something that we've never had in any of our test tools. Is a concept of test case waiting, and the idea here is on any given testing project you're going to have lots of test cases, you know, hundreds of test cases.

But when you look at those test cases some of them are going to be far more important than others. So what we've done in the tool is we've enabled you to put a test case weight as a property of a test case. And what that lets you do is identify your most important test cases.

Now, where this - you know, you get an instant benefit there where you could easily identify the test cases that are most important for a given test plan. You could run just the most important test cases for a test plan.

Okay, so there is lots of options right there off the top of - or lots of benefits right out of the top of doing this waiting.

Another benefit comes in though at reporting time, because at reporting time when you want to see you know, what test cases have passed and failed, the tools will also keep track of - will also consider those ways.

So, you know, if you have a 100 test cases that pass, of which you know, 50% of them are really important, and 50% are not. In Quality Managers Case with these waiting, the 505% that are more important will show up much prevalently across all the reports as an indication - in there results.

So you’ll get a- they’ll have a much better impact - a much bigger waiting on the report.

Now also, when the test case actually runs you have the ability to distribute the weight of that taste case. So, let’s say - and the weighting scale is arbitrary, typically we recommend a 0 to 100 scale. But you as the test designer can decide whatever the scale maybe it’s a 1 to 10 thing.

Let’s say you have a test that you know, is weighed at 100 - so a very important test. You run it, and it kind of mostly works but there was a couple little bugs along the way.

Well one thing you can do is at the end of that test after it’s run, you have the ability to distribute the weight across the results. And you can say well, the test mostly works so I’ll give it a 80% pass. But 20% of it didn’t work there was some small defect or maybe you found a spelling mistake or something along the way. And so you give it a 20% fail.

So you have the ability to distribute your test results across this weighting mechanism. So you got a really powerful way to know what’s important about your test. And a really - a really powerful way to distribute your test results so that your reporting is much more accurate.

So there is three of the kind of cool things about - about Quality Manager that I you know, I want to talk about, I got a couple more. But, what I'd like to do is maybe see - see if Matt could talk a little bit more about Test Lab Manager, give us a little bit about that.

And then maybe I can come back afterward s and give you one or two more Quality Manager things and we can address questions along the way, does that sound good Matt?

Matt Holitza: Yes, sounds great, and yes, I kind of wanted to mention also on that last point that I know as a former QA Manager that you know, I had struggled you know, wherever I been measuring test by you know, by steps or by time. And that you can now weight it it really gives flexibility to your organization to do to any organization that really weight test the way you want to.

Brian Bryson: Yes.

Matt Holitza: And you know, we don't have to - we're not going to put you in a box where we say you have to do it this one way.

Brian Bryson: Yes.

Matt Holitza: Which is really a nice thing - so, yes Rational Test Lab Manager, you know, I gave a kind of you know, really quick overview of the beginning. And one thing I mentioned that's you know, really a very powerful piece of Rational Test Lab Manager is the capability of deploying using Build Forge.

So Rational Test Lab Manager you know, I don't want to confuse anybody here. But Rational Test Lab Manager also embeds Build Forge, so when you buy Rational Test Lab Manager you also get a license at Build Forge.

So Build Forge a tool which allows you to basically automate your deployments your development deployments to lab machines or even production machines.

And so what this gives the Test Lab Manager is really a powerful ability to almost environment the - or automate the environment creation. And so they can you know, step by step - what normally be a manual task you now, too install the application and make sure it's configured properly.

And you know, a lot of times in my past life you know, these were manual pieces that one person did. And you know, if they mess up on Stip 20 they have to start over.

And so being able to automate that is a huge piece of making your test teams much more streamlined and not have that manual error in your deployment. And then also the ability to deploy your test flap - you know, your - I mean your test tools.

So you know another thing was manual (unintelligible) so it would have to go down to the you know, wherever your test flab is and put the CD in or download software install the tool and that took a lot of time for somebody to do.

And you know, one of the studies we ran, was that we found that about 35% of an average testers time is spent building up or tearing down test lab machines, and so this is kind of a - a little known fact that a lot of our testers are spending a lot of time not testing.

So you know, that's kind of the purpose of Test Lab Manager is to automate a lot of those pieces - so that's one thing. The other thing is that - that we don't talk about a lot but you know, I wanted to talk about here.

Is that Rational Test Lab Manager also integrates with our Tivoli Solutions. So one of our Tivoli Solutions is called Provisioning Manager. Now, Build Forge is powerful and of itself, but what Tivoli Provisioning Manager allows you to do, is truly the Nirvana of Test Lab Management.

Where you can actually take a piece of hardware and have - you can point Tivoli Provisioning Manager on it, and it will actually install the operating system, the database server, the application server whatever tools you want on it.

And you can basically build your test lab from the ground up using Tivoli Provisioning Manager. So this is you know, a very powerful solution that really would allow you to almost completely automate your environment management of your test labs.

And so this is you know, something that you know, we don't talk about a lot but you know is a pretty cool feature that you know, if you have these two put together you can basically request your environment you want, and then the next morning it'll be ready for you to run your test on.

Those are a couple a little things of a Test Lab Manager, and - Brian you want to...

Brian Bryson: Yes - no, I was it’s funny - hearing you talk about the integration to Tivoli made me think of a couple of other things about Rational Quality that I wasn’t really planning to talk about. But - just sort of popped into my mind that you know, since we’re built on the Jazz platform we’ve had a fair amount of partners that have done some integrations already to Rational Quality Manager.

So, now sort of in the Test Lab Management domain is to simply the process of working with sort of VM or Virtual Machines. So if you’re - you know, if part of your test lab is virtual, you know you’re using VM where - virtual machines as an example.

The Quality Manager is Test Club Manager will integrate with VM Logics, and VM Logics (Insurgent) to help with the management of that - of those virtual machines.

So if you want to interact with you know, large groups of virtual machines there is a couple partners that have built in integrations already. Couple other integrations was probably you know was at CM Logic, was has an integration with the (unintelligible) changed management system and open source changed management system.

Device Anywhere which gives the ability for Rational Quality Manager to launch test on the phones - on the handsets. So they have a tool for you know, testing handsets and testing telephone or mobile phones I should say.

And you can now use Rational Quality Manager to launch the task on the mobile phone and then collect the results back into Quality Manager along with all your other test results for consolidated reporting. So you get some - a neat integration there.

For those are that are interested in sort of policy management Web Layers has an integration for design time governance sort of policy enforcement angle and some best practices there.

So you know, just hearing you talk about Tivoli made me think of those partners and I sure hope I haven't forgotten them - any of them. There is quite a few that have already developed some stuff there.

So let me just close off we only got a few more minutes, I want to talk about a couple other things in Quality Management that are kind of cool. Maybe a less obvious one is the tagging concept, so kind of the Web 20 way of searching for things.

You know in the past whenever we organized the test plan it was generally (hierarchly) organized, you have your own top level and then things would branch off and everything sort of fit nicely into his place in a branch on a tree. And that's fine, you know there is nothing really wrong with that approach.

The only problem is and the feedback we got about that sort of hierarchal structure way of organizing test cases and requirements. Is that it's great for the person who built the tree to be able to find stuff. But it's really not that great for - if you're trying to pick up and understand where - how things are organized. It's really kind of tough to figure out where everything went.

And it's also kind of the limitation that you know, if you have a test case and you're using sort of a tree structure to organize all your test cases they can only live at one spot. Which may be fine, or may not be realistic, maybe a test case belongs effectively in multiple branches on a tree.

So instead of going the hierarchal root with quality manager, we've gone the tagging route. And the idea with tagging is whenever you have a test case, whenever you have a test plan, you tag it with key words then you could - that you can use to search and bring up and find that test case.

Now the title is an automatic tag and then you can go and add in other ones. But it gives you a much more sort of Web 20 dynamic way of finding your data.

If you used Google Gmail that's kind of the structure they use for searching across your emails.

So a little bit more modern approach to doing it, and it you know, frankly it takes a little bit of adjustment to sort of get used to this new way of doing things. But boy, there is no going back once you've - once you started with the tagging aspect.

Matt Holitza: Yes, and one thing that I would mention is that one idea that I would talk with is we also allow you to have a more structured way. You know you could have your own categories and drop downs...

Brian Bryson: Yes.

Matt Holitza: ...to measure things. If you want to do things more formally in a formal setting.

Brian Bryson: Yes.

Matt Holitza: But then if you allow your team to use the tags, that allows them to kind or organize it in their way. And so you kind of have the best of both worlds.

Brian Bryson: You could get a real hybrid there.

Matt Holitza: Yes.

Brian Bryson: And the - you know, the categories you can add as many categories and drop down classification as you want - so customizable. So you can - I mean you can classify until the cows come home as well as the tags and have both.

Matt Holitza: Yes.

Brian Bryson: Great point. Okay, let's take a quick look at the clock here, a few minutes let me hit you with a couple more things. So, probably should have talked about it before at this point, because kind of the concept of this call is to talk about things you might not pick up from the you know, first glance of the tool.

One of the things you probably would pick up that we should really talk about anyway - one of the things we should talk about is the manual testing capability. It's blatantly obvious that the tool does manual testing, you know it's a real push for the tool, one of the number one features that was requested of the tool.

But, let's spend a few second on it, because there is some interesting aspects of the manual testing portions that are kind of neat.

So, the high level description you know, quite simply the manual testing part of the tools where there is an interface provided that lets you write down all the steps of a test, and sort of capture all those steps.

And then when you go to actually execute that test, it sort of feeds them back to you one at a time and then you can tag each step as you know, just completed if it's just a you know, like quick here sort of thing it could tell you know, I did click here.

And if it's a verification point where there is actually something to verify the - you can actually go in and assign a pass or a fail or a blocked or a status to it.

So you know all very generic I guess if you will manual testing (unintelligible) and execution capabilities.

Now what are some of the neat things, well sharing steps amongst test. So there is a concept of - called key words in the tool. And the idea with the keyword is that you can group a bunch of steps together call it a keyword, and then use that keyword in multiple text.

So a common example is logging in to an application, maybe there is five or six steps required to log in.

Well you would make those five or six steps part of a keyword, and then you could attach that keyword to all of your text that required it. You know presumably almost every single one you would think.

And they would share those common test steps, you only have to write the test once. If you had to ever change the way you log in, or change the access approach, you would change it once and it would propagate to all your test.

So that's kind of step one of keywords, step two of keywords if you will, so beyond just the basic benefit of being able to share data and minimize your maintenance by sort of having the steps only once but used everywhere.

If you could automate your keywords, so again for you know, the log in example. You could using like Rational Functional Testers as an example record the steps for logging in.

And then every time that someone used that keyword they would be able to instead of actually having to do the log in steps manually, the functional tester strip would kick off, do the log in for you, and then return control back to you to complete the rest of the test.

So you'd have a way of doing hybrid manual and automated testing. You'd be able to mix the two up, which is really kind of a neat concept, and really for teams that are just getting into test automation, a great way to start out where you automate small little chunks that are common that are used frequently across your application, turn them into keywords and make them part of your manual test to save you time an effort.

Matt Holitza: And I'll just jump in.

Brian Bryson: Yes.

Matt Holitza: And another application of this that I have talked to customers about is actually since it's a Web based application - Quality Manager is, you can use it for user acceptance testing.

So, you automate the pieces of the test that you don't want the user to have to worry about, like log in, navigate to a certain place. Do their verifications and you do manually the verification piece. and then it automate to log out.

And so the user only has to - or the stakeholder only has to worry about the piece that they are concerned about. And not all the ancillary tasks.

Brian Bryson: All the stuff right?

Matt Holitza: Yes.

Brian Bryson: All the plumbing and stuff (unintelligible).

Matt Holitza: Mm-hm.

Brian Bryson: Yes - no, it's great. Two other quick things - as I look at the clock again. Two other quick things about manual testing that I'll jump in with. When you are - common issue when you're doing manual testing is - it's very simple if your test is basically click here, click here, click here - verify. You know those are fine.

But if your test is instead, click here, click here, click here, type in all sorts of data and then verify. Well that can be A, tedious to have to retype data over and over and over every time you run the test.

And it can be somewhat error-prone especially if you're - you're doing like financial applications where you're pumping lots of numbers in and - or, you know, you're testing something like license keys where you've got these you know, cryptic strings of characters and numbers and getting those right is often tricky.

So one of the things you can do as part of offering a manual test, when you actually write your manual test is you can provide the test data at that time when you're writing the test step so that when you actually go to execute the test, what Quality Manager will do is it'll pop up a dialogue that says here is your test data, it's now on your clipboard.

And then you go to your application you could paste it into your application instead of having to type it, and then you save the effort of sort of you know, repetitively typing in that data, and you know, potentially making in sort of typos.

So you got that sort of assisted data entry I think it's being called where you know, it does the data entry for you.

Now the flip side of that is if you have to compare you know complex data return values in your system. If that's the case, what the tool will do for you is if you copy the data off your screen into the clipboard, you can paste it into Quality Manager and it'll compare that to the baseline that you know, recorded when you wrote the test. And it'll actually do the comparison for you and highlight any differences.

So for manual testing that's a real time saver, especially if you got you know, complex large strings of numbers and data that you need to compare to have it do the entry in the comparison for you is a really powerful thing.

So, probably something that you would have caught just on first use of the tool kind of deviating a little bit from a cool thing. But I wanted to you know, spend a little bit of time on it.

So I think we've got two minutes left. I think we'll kind of wrap it up here. Matt, did you have anything else that you wanted to throw out or talk about before we call it day?

Matt Holitza: I don’t think so, I was just wondering if there is any other questions out there, please feel free to get in a last minute question.

Brian Bryson: Yes.

Operator: Ladies and gentlemen at this time you would to ask a question, please press star 1 on your telephone keypad.

Brian Bryson: So as we - we’ll give a few second for anything to - any last minute things to come in. A couple of resources to talk about you know, since the (unintelligible) is so new I’m not really surprised there is not a lot of questions out there today since you know, we’ve only been out there a couple of weeks.

A couple of resources just to get you going on Quality Manager, so first of all, there is a demo on IBM.com/Rational if you go there you can actually see Quality Manager in action, just a five or six minute demo, take a look at the tool, have a look at it and see how it works.

You can also read the product data sheet which will sort of outline some of the key features - not all of them, but you know the main things about the tool and what it does.

Now, once you kind of get that basic foundation knowledge down, from then I would head over to IBM.com/DevelopWorks, and what you’ll find on Developer Works is two things, one is the actual product. So you can download it from there get a free 60 day - so two month trial evaluation that you can try out with an unlimited number of users.

So if you need - you know, should spread out with teams of hundreds of people you can all access at the same time.

And you'll see a couple interesting articles, one is up there now and one should be up there in a day or two from an Author named (Mike Kelly), who is one of the first users of Quality Manager and he's written - really call it a tutorial, its sort of a guide - a first use guide of the tool. And it'll get you through creating your first project, creating your first test plan, writing your first manual task, executing your first test, looking at your first results.

So if you're sort of just trying to get ramped up, that's the path I would recommend. IBM.com to see the demo and data sheet, and IBM.com/Developer Works to get the code and to go through this sort of getting started guide by (Michael Kelly) and author that's you know, published this article for us.

Matt Holitza: And I'd also mention that same places you can find information about Rational Test Bud Manager, and if you download the evaluation copy of Rational Quality Manager, it also includes the trial of Rational Test Bud Manager as well.

Brian Bryson: Perfect, yes exactly. Which is confusing for some people the first time they look for the Test Bud Manager Trial, and it's Quality Manger, and it's like wait a second - it's all in one.

Matt Holitza: Yes, all in one.

Angelique Matheny: So (Cassandra) any last minute questions?

Operator: We do have a question...

Man: (Unintelligible).

Operator: From the line of Rob Smith.

Brian Bryson: Hey Rob.

Rob Smith: So I had a question regarding manual test within RQM. You guys - or have you done anything to a comparison between and that the existing Rational Manual Tester, just so we know and customers know if there are any differences what they are?

Brian Bryson: So one of the designs principles of the tools is that it s a complete functional replacement for the current manual tester tool. We didn’t really even talk about that today, so it’s a good question.

Rational Manual Test was a previous tool that we had for manual testing that effectively was rolled into quality management.

All of the functionality that was in manual test is in Quality Manager that was one of the - one of the design principles that we had for the tool. So it’s all there, there is also the ability to import your manual test into Quality Manager.

So again just like Test Manager and Clear Quest Test Manger you’re not losing any data. It’s all imports, and if you wanted to today if you had a manual test project you could dump it all into Quality Manager and move forward.

The only real difference and its one of the limitations of being web based versus the former eclipse base. Is the original manual test had a kind of a neat feature that let you adjust the transparency of the manual test dialogue.

So the idea was when you ran the test, you could address the transparency of the manual tester application so that you can see behind it, and get a little more screen real estate for whatever it was you were testing.

Because we're now browser based, I think that's one of the few things that dropped off, it's not the only thing that dropped from the original manual test that you won't find in Quality Manager since we couldn't - there is no way to adjust the transparency of the browser we don't have control over that from Quality Manager.

Matt Holitza: But I'm sure we have researchers on it.

Brian Bryson: I wouldn't be surprised. There probably even a Firefox Plug In that does it to now that I think about it. (Unintelligible) and Firefox.

Angelique Mattheny: Any other questions (Cassandra).

Operator: Yes your next question comes from Lillian.

Brian Bryson: Hey Lillian, how's it going?

Lillian Bunch: Good, (Caroline Galya) has joined me as she has a question. Is this product available that cold be installed at a client's location?

Brian Bryson: Is it available and could be installed at a client's location?

Lillian Bunch: Yes, because I'm being told it's not supposed to be available until sometime the middle of next year.

Brian Bryson: No product - October 28th has been available since October 28th, is available now. Go grab a coffee of a - I mean Google for it just Google Rational Quality Manager Trial you'll get it first hit I'm sure.

If not you can go to IBM.com/DeveloperWorks.

Lillian Bunch: What is the application - what is the implications to someone who has ClearQuest and (unintelligible) Manager, PQTM and (Repro) installed today.

Brian Bryson: So if you have ClearQuest Test Manager or Test Manager let me start with those two, all of that data can be imported into Quality Manager and so you an take what you've got today, role it into Quality Manager and move forward.

Lillian Bunch: Okay.

Brian Bryson: For Requisite Pro quality Manager has the same integration to Requisite Pro - actually let me back up. So Requisite Pro for the initiated is a tool for the requirements management from IBM Rational.

Rational Quality Manager has an integration to Requisite Pro, so if you're using Requisite Pro you can have Quality Manager link to your requirements in Requisite Pro and use those as a foundation for the test plan. So it's a live link whenever requirements changes - Quality Manager is aware etcetera.

Lillian Bunch: So I don't have - it doesn't matter that I'm not on the latest version of Req Pro?

Brian Bryson: Oh, that's a good one, that one I'm going to have to check out. Do you know that one Matt, do we need the...

Matt Holitza: I don’t know that one, I am guessing it’s probably the most recent one but I can’t say definitively.

Lillian Bunch: Because I know (unintelligible).

Matt Holitza: But we could find out for sure.

Lillian Bunch: I know there was a new release of that supposedly in the same time frame.

Brian Bryson: There was.

Lillian Bunch: What about (unintelligible).

Brian Bryson: Actually this month yes sometimes - yes.

Lillian Bunch: What about integration into ClearQuest.

Brian Bryson: So yes, with ClearQuest, ClearQuest again for the (unintelligible) issues are defect and change tracking system. Integrate by the Jazz platform to Quality Manager. So there is something called the ClearQuest Connector which will essentially link and let ClearQuest plug into Rational Quality Manager.

So that when you’re in Quality Manager and you create a defect I wouldn’t even talk about this, but the Quality Manager has the capability to track it’s own defects.

It will synchronize those defects with ClearQuest so that you can use ClearQuest for your defect tracking.

Lillian Bunch: So the Jazz component everything is working today?

Brian Bryson: Yes, all good.

Lillian Bunch: And you have clients that are currently using this?

Brian Bryson: Yes, yes in fact October 28th the day of release is also the day of first sale. We had a long beta program for this tool - the tool went probably six or seven months in beta probably since June, so June, July, August, September, October I guess if you want to count that.

So five months of open public data, where anybody can just download it and use it and we have a quite extensive program there.

So we've had very quick uptick on the - on Quality Manager.

Lillian Bunch: So other than the adapter for CQ do I need to be at the latest version on CQ as well?

Brian Bryson: That I don't know, I'll have to check but I don't think so. My guess would be that you do need the latest Requisite Pro because that's the only - I know when I've done my integrations I could have probably use the latest.

Matt Holitza: 7.1 - yes.

Brian Bryson: With 7.1 yes. With ClearQuest I'm not sure.

Matt Holitza: ClearQuest I don't think it's as necessary, but I can't say for sure either.

Brian Bryson: Yes.

Lillian Bunch: Is that adapter available for with any version of ClearQuest?

Brian Bryson: It actually comes with Jazz - it's a Jazz connector.

Lillian Bunch: Okay.

Brian Bryson: Actually come with RQM?

Lillian Bunch: How extensive is the migration from current to the new world?

Brian Bryson: That is something we've worked on a lot with customers, and it's been pretty battle tested. And I think it's very robust now. I - it probably really depends on the number of records were looking and you know, we've done it first for our beta customers for you know, several tens of thousands of records.

Matt Holitza: And just to give you - I mean it's a wizard based type...

Brian Bryson: Yes.

Matt Holitza: ...point to one, point to the other.

Brian Bryson: Because we understood that you know, introducing a new test management tool that we need to have some really good robust tools to migrate data over.

Lillian Bunch: All right, thank you.

Brian Bryson: Good, oh thank you.

Angelique Matheny: Brian and Matt I think we've run out of time.

Brian Bryson: Oh you know, we should have pumped - we should have pressured more of questions (unintelligible).

Matt Holitza: I know.

Angelique Matheny: Yes.

Brian Bryson: I just expected (unintelligible) but that’s too bad.

Angelique Matheny: As always very valuable session. Well do you want to plug your next teleconference for Thursday?

Brian Bryson: Yes actually so we’re doing another one of these on Thursday. Same time, I don’t have the number handy, maybe Angelique if you could look it up while we’re talking about it for a second.

Angelique Matheny: Oh it’s the same number.

Brian Bryson: Oh, same number, perfect. Well were going talk again a little bit about Quality Manager, but a little bit more about - at the same time that we release Quality Manager we also released updates to Rational Functional Tester, Rational Performance Tester, and Rational Service Tester which are our test automation tools.

And we were going to spend that call talking about sort of what’s new with those applications.

So you know, we’re going to spend a little bit of time on (unintelligible) Manager, a little bit of time on Quality Manager there.

And if you know, if we did live any questions dangling lets pick them up on Thursday, same time, same number.

Angelique Matheny: And sure you can always send them to the AskUsNow@US.IBM.com...

Brian Bryson: Ask Us Now - yes.

Angelique Matheny: ...email as well. So thank you so much, if anyone would like to listen to this conference again, or share it with your colleagues this will be available for replay in MP3 format in about a week or so, on its Rational Talk View site, that's IBM.com/Rational/Talk.

Again I'd like to thank our guest, Brian Bryson and Matt Holitza for being with us to talk about Quality in Action Using Rational Quality Manager.

We also like to thank you our audience for your interest in IBM, we hope to see you back for another one of our events in the near future. Thank you very much, talk to you soon.

Operator: This concludes today's conference call. You may now disconnect. Speakers hold the line please.

END