

Lab Management using IBM Rational Test Lab Manager

Extending IBM Rational Quality Manager for collaborative, comprehensive test planning and test asset management

Skill Level: Introductory

[Michael Kelly \(mike@michaeldkelly.com\)](mailto:mike@michaeldkelly.com)

Consultant

MichaelDKelly.com

28 Oct 2008

This article describes how test lab management works with IBM® Rational® Quality Manager. It also shows you how to create and view lab resources and work with lab resource reservations and requests.

IBM® Rational® Test Lab Manager is an extended component of IBM® Rational® Quality Manager, which is collaborative, Web-based, quality management software for comprehensive test planning and test asset management. Built on the Jazz™ platform, Rational Quality Manager is designed to be used by test teams of all sizes. It supports a variety of user roles, such as test manager, test architect, test lead, tester, and lab manager, as well as roles outside of the test organization. With the Rational Test Lab Manager component, your team can integrate test lab activities into your organization's quality management lifecycle so that you can centrally manage all resources -- people, artifacts, and lab machines.

This article describes how test lab management works with Rational Quality Manager. It also shows you how to create and view lab resources and work with lab resource reservations and requests.

The nature of test lab management

At the last Fortune 100 company that I worked at, we had hours and hours of

automated regression tests. Some of those were at the unit level, some were at the API and Web service levels, and others were at the GUI level. For our GUI-level and Web service regression tests, we maintained a test lab of test execution machines. At one point, I think we had more than 70 machines in the test lab.

The biggest difficulty in managing a lab that size, aside from just keeping the hardware up and running, is managing the various test configurations and keeping track of who is running what tests on which machines. Even when we replaced our hardware and migrated to virtual execution machines, the challenges of managing test configuration and coordination were costly.

Rational Quality Manager introduces the notion of *test lab management* -- functionality designed to solve the problem. You can track the resources in your lab by using the lab management tools. You can create data for physical machines and virtual images, search for resources with specific configurations, and manage requests and reservations for lab resources.

Creating and viewing lab resources

Lab management tasks are often performed by people in the roles of lab manager and tester. Therefore, to work through the exercises in this article, create two users and assign one of them to each of those roles. This is important because the role associated with each account determines the options that are available and the functions that can be accessed in the lab manager editors.

There are different types of tasks that are typically associated with these two test lab roles:

- **Lab managers** typically manage resources in test labs. These resources can include physical machines, virtual machines, and virtual images, and lab managers can work with team managers to allocate lab resources across the products under test, and to install, set up, and maintain software applications across the entire lab. The lab manager role is for a team member who deploys builds to test environments and who ensures that the lab resources are set up correctly.
- **Testers** simply run tests on lab resources.

Creating a new test lab machine

The basic building block of lab management is the lab machine. You can create physical or virtual machines, and you can create them one at a time or import multiple lab resources by using XML. For this article, you'll be adding a simple physical machine to the test lab by following these steps:

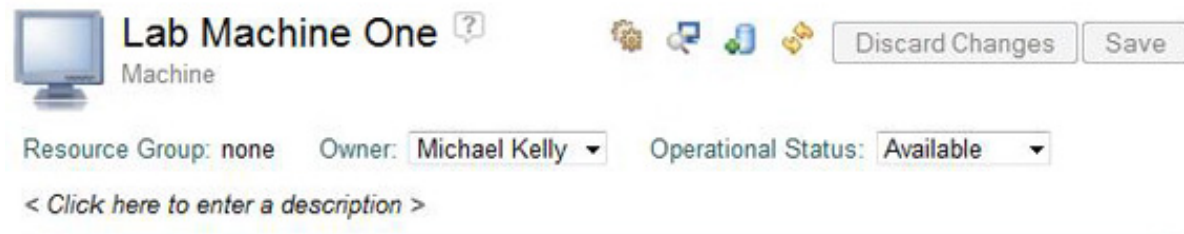
1. In the left navigation pane, click the **Lab Management** icon (two computers) and select **Create Machine** from the drop-down menu. (Figure 1)

Figure 1. Create a lab machine



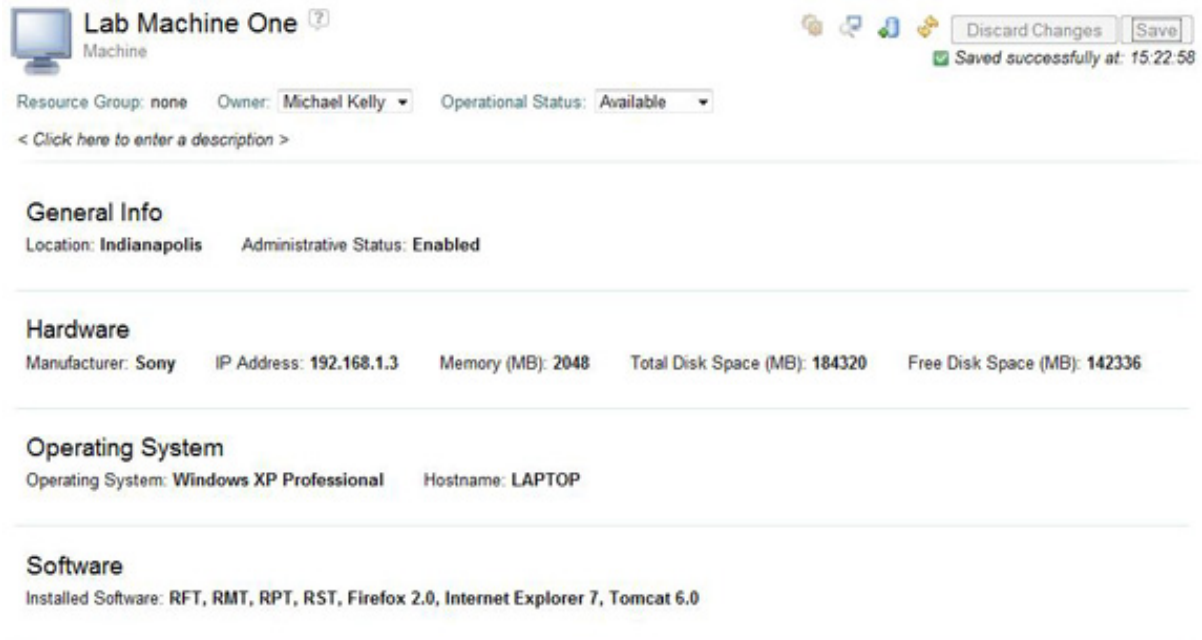
2. This should open the Create Machine editor. You need to enter a name for the machine and then assign an **Owner** and an **Operational Status** (Figure 2).

Figure 2. Select owner and status from options



3. Next, you'll enter more information about the host machine. You can enter as much or as little of this information as you like. However, the more the better. Figure 3 shows the summary information for the test machine used for this article, including location, administrative status, operational status, and details about hardware, software, and the operating system (OS).

Figure 3. Test machine details



4. In **Operational Status**, make sure that you select **Available**, and in **Administrative Status**, select **Enabled**.
5. After you have finished entering your lab machine's information, click **Save**. If you get error messages the first time that you save (as I did), hover your cursor over the error message to get more details about the problem. When you have saved successfully (which may take a minute), you will see a "Saved successfully at..." confirmation message.
6. If you want to make sure that your machine is entered correctly, you can click the **Lab Management** icon in the left navigation area and select **All Lab Resources**. You should see the computer added to the list (Figure 4).

Figure 4. Verifying that the machine is added correctly

Show All - Items per page		Previous 1 - 1 of 1 Next			
<input type="checkbox"/>	Name	Resource Group	Operating System	IP Address	Reserved
<input type="checkbox"/>	Lab Machine One		Windows XP Professional	192.168.1.3	

Previous | 1 - 1 of 1 | Next

Managing your lab resources

After you have added a lab resource, you have a variety of methods of managing that resource. This article does not cover setting up each method, but they are important to cover briefly so that you are familiar with the test environment, virtual collection, and lab resource group features. You can access these groups from the

Lab Management icon.

Test environments

One way to manage lab resources is to create test environments. In Rational Quality Manager, you can create a test environment that can be stored and reused in different lab management tasks. Environments are useful in fulfilling lab resource requests and searching for lab resources with common criteria. Think of a test environment as simply a set of attributes: available memory, operating system version, or specific installed software.

Virtual collections

Virtual collections are logical groupings of virtual images or virtual machines that represent application topologies. A virtual collection could consist of a virtual image for a Web server and another virtual image for a database server when both are used together. If your systems is integrated with an external provider that manages virtual images and machines, you can view, deploy, and assign test environments to virtual collections in Rational Quality Manager.

Lab resource groups

Creating a lab resource group enables you to track and manage multiple resources together. You can also associate teams of people with the group. A common use for this might be automated test execution machines or any group of machines that are dedicated to a particular group of testers. The team always has exclusive access to the group of lab resources that you define. You can view all lab resource groups by team area to see who has access to what resources.

Working with lab resource reservations and requests

So why, you may ask, do you need to set up and enter all of these computers? What's the point? The advantage? On past projects, I've managed the status of the lab machines by using spreadsheets. From software versions to who had a machine "checked out", everyone was supposed to keep the information in the spreadsheets up to date. How well do you think that worked?

Not very well. My experience is that the information about the software installed quickly becomes out of date. The information about the hardware (typically RAM for physical machines) becomes outdated. The status of who is using which computer daily is inaccurate, and few people check in their machines early if they finish their testing early, if they check them back in at all. It ends up taking a lot of manual effort, e-mailing people to update their status, and shared frustration across the team.

In Rational Quality Manager, one of the core ideas is that you can manage your resources across the team or even across teams by using those lab resources groups. You can reserve lab resources for immediate use or for specified periods of time. You can also request a lab resource that may not even exist yet, thereby giving

the lab manager one place to look for the list of new configurations to create.

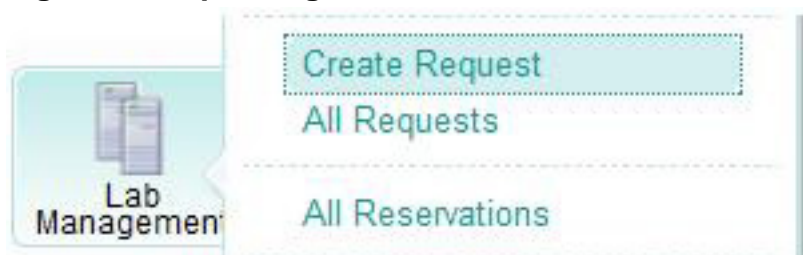
If you are a lab manager, you can track and manage lab resource requests by viewing the status of all requests, assigning requests to lab personnel to fulfill, or fulfilling them directly. When you read "lab personnel," don't just think of a group of people who just manage lab equipment (few teams are *that* big). Instead, think of different roles: someone who has access to the room with the physical machines, someone who has administrative access to the virtual images, someone else who manages licensing for testing tools, someone else who has root access for specific servers, and so on. Different people on the team, even if they are all testers, might have different lab roles. For some teams, assigning and tracking requests is, largely, effectively coordinating how all of those people work together.

Creating a request for a lab resource

The whole process gets kicked off by requesting a lab resource. It's the basic building block of the lab manager's workflow. To request a new resources, try the following steps:

1. In the left navigation pane, click the **Lab Management** icon and select **Create Request** (Figure 5).

Figure 5. Requesting a new lab resource



2. This opens the Create Request editor (Figure 6). Select a **Priority** for your request, a **Respond by** date (if applicable), and a **Summary** of your request (required).

Figure 6. Create Request editor



Request <09:12:21>  Cancel Request Discard Save Submit
Request New / Not Yet Saved

Created by: Michael Kelly Status: Draft Priority: Normal ▾

Assigned to: Unassigned

Respond by: 

Summary: * I'd like a computer to run the regression testing for build 2.3.2.

3. Next, enter the date you want the resource by in the **Reservation**. For this example, assume that you want the resource over the weekend. Use the down arrows to select **From** and **To** dates (Figure 7).

Figure 7. Specify dates to reserve the resource



Reservation 

Enter the date range for the requested resources.

From:  ▾ ▾ ▾

To:  ▾ ▾ ▾


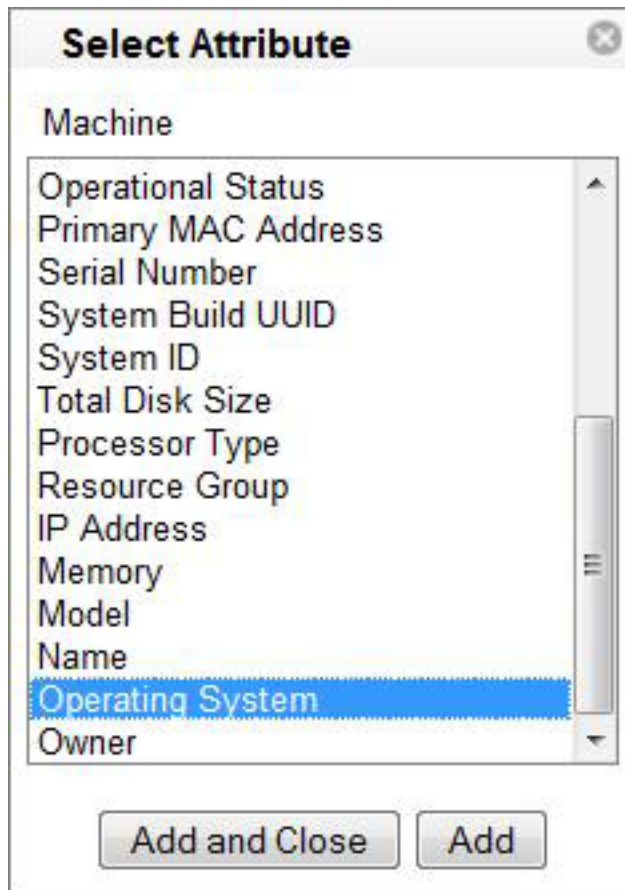
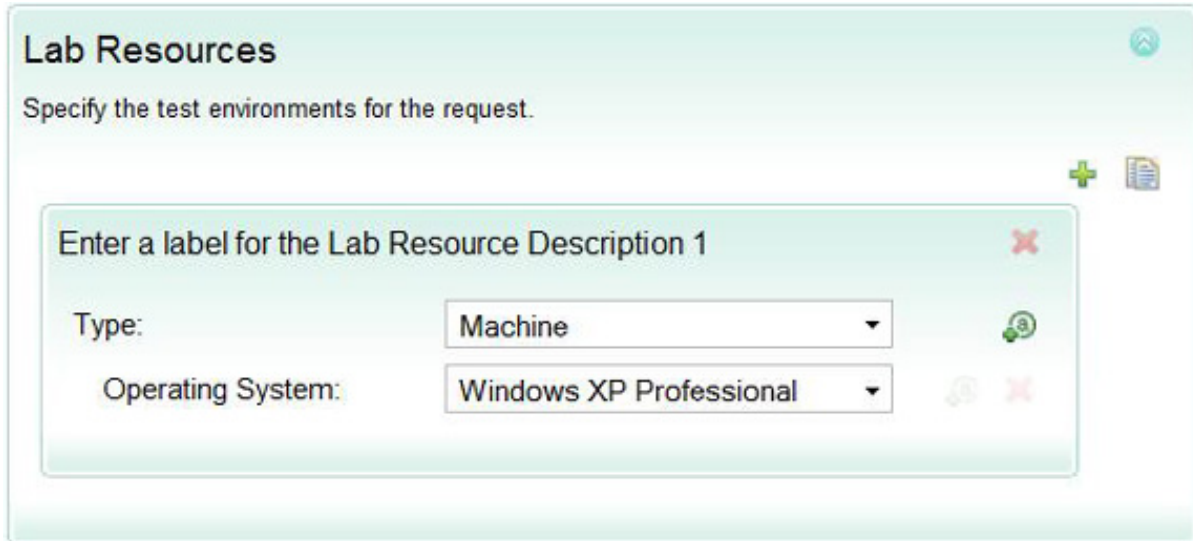
4. In the **Lab Resources** section, specify the test environments that you want to request. To start, click the **Add Criteria** icon  (lowercase letter A in a green circle).
5. In the resulting **Select Attribute** dialog (Figure 8), select **Operating System** and then click **Add and Close**.

Figure 8. Select Attribute dialog



6. In the **Lab Resources** section (Figure 9), you will see the Operating System attribute added with a drop-down menu beside it. In that drop-down menu, select the operating system that you specified for your resource earlier. "Windows XP Professional" (or whichever OS you gave your resource in the first part of the article). You can repeat this for as many attributes as you like.

Figure 9. Lab Resources view to specify test environments



Lab Resources

Specify the test environments for the request.

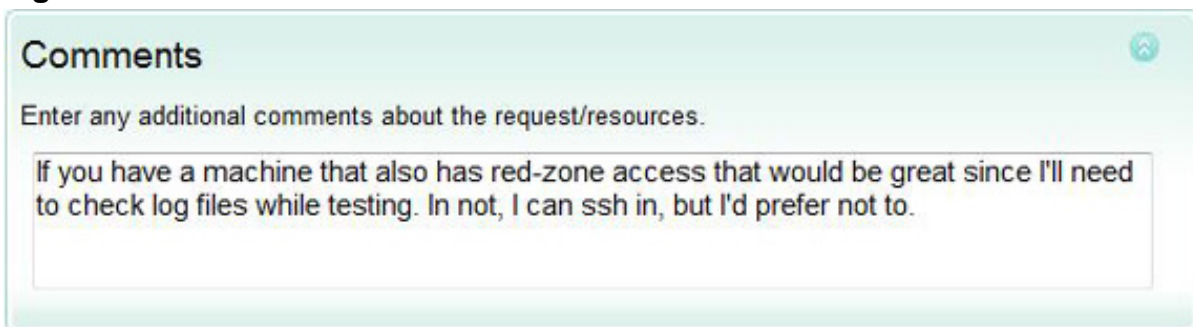
Enter a label for the Lab Resource Description 1

Type: Machine

Operating System: Windows XP Professional

7. Finally, if you have specific setup or configuration details that you require for your testing, you can add a details about that in the **Comments** section (Figure 10).

Figure 10. Comments section



Comments

Enter any additional comments about the request/resources.

If you have a machine that also has red-zone access that would be great since I'll need to check log files while testing. In not, I can ssh in, but I'd prefer not to.

8. When you are finished, click **Submit**.

You have just created your first request. Next, you'll see how you find and fulfill requests.

Viewing requests

Now that there is a request waiting for fulfillment, let's look at the methods that you can use to find it. There are two basic ways to monitor requests.

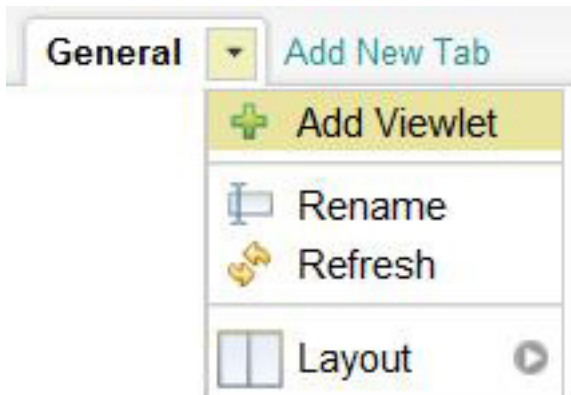
- The first is the brute force method, which is to look at all of the requests and then filter based on different criteria.
- The second method is similar but a bit more elegant: you can add a Viewlet to your dashboard.

Here, you'll add a Viewlet, but you can get to the same files by clicking the **Lab Management** icon and selecting **All Requests**.

To create a Requests Viewlet for your dashboard, perform these steps:

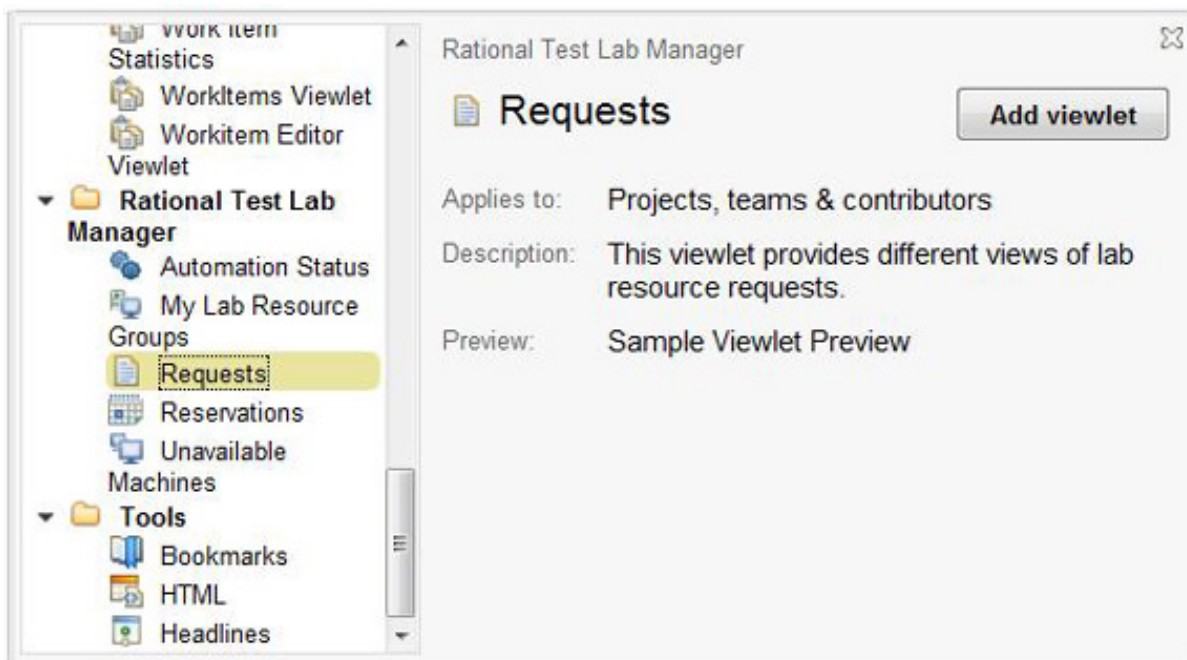
1. On your dashboard (the **Home** tab), click the down arrow on the **General** tab and select **Add Viewlet** (see Figure 11).

Figure 11. Dashboard view



2. In the resulting Viewlet explorer (Figure 12), select **Requests** under **Rational Test Lab Manager**, and click **Add viewlet**.

Figure 12. Viewlet explorer



- This will add the Requests Viewlet to your dashboard (Figure 13), probably at the top of the screen. You can move it around, but wherever it is, you should see the request that you just entered.

Figure 13. Requests Viewlet

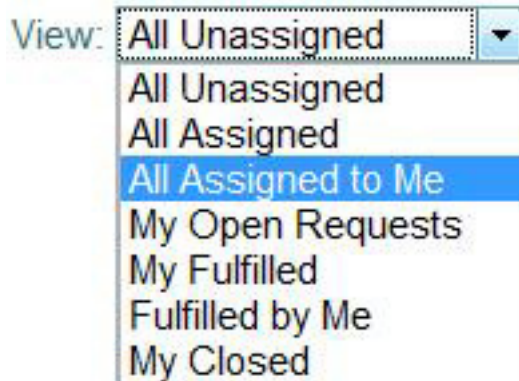
The screenshot shows a 'Requests' viewlet window. At the top, there is a 'View:' dropdown menu set to 'All Unassigned' and a search box labeled 'Type Filter Text'. Below this is a 'Show All' dropdown and 'Items per page' controls, along with 'Previous' and 'Next' navigation buttons and a page indicator '| 1 - 1 of 1 |'. The main content is a table with the following columns: ID, Summary, Created, Respond By, Assigned To, Priority, and Status. A single request is listed with ID 3, a summary 'I'd like a computer to run the regression testing for build 2.3.2.', created 26 minutes ago, responded to yesterday at 12:00:00 AM, and has a priority icon and a status icon.

ID	Summary	Created	Respond By	Assigned To	Priority	Status
3	I'd like a computer to run the regression testing for build 2.3.2.	26 minutes ago	Yesterday at 12:00:00 AM			

Navigation: Previous | 1 - 1 of 1 | Next

- Assuming that someday you'll have a thriving test team with hundreds of requests, there are several filtering options available to you. The default is "All Unassigned." Assume that you're a lab manager or one of the team members in that role, and you want to select "All Assigned to Me" (Figure 14).

Figure 14. "View" field with drop-down menu to select



Even better, you can create multiple Viewlets with different views.

Working with requests

Now that you can see your request, you can start work on it:


1. In the Viewlet, click the link to the request that you created to open the **Request** editor. It looks a lot like the Create Request editor, but there are subtle differences:
 - A. First, you can assign the request to someone on the team (for example, a system administrator may need to provide the specific access requested in the Comments section), as this image shows.

 - B. Second, you'll see **Fulfill** and **Return** buttons at the top (Figure 15). **Fulfill** is a command that pushes the request through the workflow for request tickets.
 - C. Third, under **Lab Resources**, you'll see a **Find Machine(s)** section (Figure 16).

Figure 15. Options for managing the request



Figure 16. Find Machines section



2. To assign the request to a machine (or set of machines), select **Find Machines**.
3. After the list of matching and available machines loads, select the check box next to the machines that you want to reserve. For this example, select only the one that you added earlier (Figure 17).

Figure 17. Assign the request to a machine or set of machines




4. When you are finished, click the **Reserve** icon  (spreadsheet with a blue cell). This will update the ticket to show the machine reserved.

Figure 18. Request status updated

Reserved Machine:

Machine Name	Start Date	End Date
Lab Machine One	Oct 10, 2008 12:00:00 AM	Oct 12, 2008 11:59:00 PM

5. When you are finished, click **Fulfill**.


Viewing reservations

Now that you have a reservation to process, you can view it in the same ways that you viewed your requests. This time, rather than using a Viewlet, use the **Lab Management** icon:

1. Click the **Lab Management** icon and select **All Reservations**.
2. Again, you have the option to change your view. You can see the status of each request and the associated details (see Figure 19).

Figure 19. Select which reservations to view



3. If you select a reservation, you can cancel the reservation by using the **Cancel Selected Reservation icon** , which has a red X in the upper-left corner.

Next steps

Lab management using Rational Test Lab Manager with Rational Quality Manager is about bringing order to chaos. This article merely scratched the surface by looking at the basic building block (the machine) and the basic workflow. The next steps for setting up Rational Quality Manager are to configure the test environments, virtual collections, and lab resource groups for your team. You will also start integrating other Rational tools into the workflow (IBM® Rational® Functional Tester and (IBM® Rational® Performance Tester, for example) and learn how the lab management structure helps you manage those tests. Watch developerWorks for articles on intermediate and advanced Rational Quality Manager topics.

Resources

Learn

- Read author Mike Kelly's related article on developerWorks: [Managing your first project with IBM Rational Quality Manager](#).
- To find more information about IBM Rational Quality Manager and related products, check the [Software testing and quality management](#) page on IBM.com.
- See the [Rational Quality Manager product roadmap](#) on developerWorks - a resource roadmap for all users.
- For tips, tricks, and updates, track the [IBM Rational Quality Manager blog](#) for technical resources and best practices for Rational Software Delivery Platform products.
- Visit the [Rational software area on developerWorks](#) for technical resources and best practices for Rational Software Delivery Platform products.
- Explore [Rational computer-based, Web-based, and instructor-led online courses](#). Hone your skills and learn more about Rational tools with these courses, which range from introductory to advanced. The courses on this catalog are available for purchase through computer-based training or Web-based training. Additionally, some "Getting Started" courses are available free of charge.
- Subscribe to the [Rational Edge newsletter](#) for articles on the concepts behind effective software development.
- Subscribe to the [IBM developerWorks newsletter](#), a weekly update on the best of developerWorks tutorials, articles, downloads, community activities, webcasts and events.
- Browse the [technology bookstore](#) for books on these and other technical topics.

Get products and technologies

- To find more information about IBM Rational Test Lab Manager and related products, see the [Rational Test Lab Manager](#) page on developerWorks.
- To find more information about IBM Rational Quality Manager, see the [Rational Quality Manager](#) page on developerWorks.
- Download [trial versions of IBM Rational software](#).
- Download these [IBM product evaluation versions](#) and get your hands on application development tools and middleware products from DB2®, Lotus®, Tivoli®, and WebSphere®.

Discuss

- Find answers to your questions and share solutions in the [Rational Quality Manager](#) forum on developerWorks.
- See also the [Rational TestManager](#) forum on developerWorks.
- Check out [developerWorks blogs](#) and get involved in the [developerWorks community](#).

About the author

Michael Kelly

Mike Kelly is an independent consultant located in the Midwest region of the United States. Mike also writes and speaks about topics in software testing. You can find most of his articles and his blog on his Web site, www.MichaelDKelly.com.