

IBM Software Demos
IBM Enterprise Modernization - Manage and Develop System z
Applications with Rational Developer for System z

<0:00>

Welcome to this demonstration of Managing and Developing System z Applications. In this demonstration, we will show how Rational ClearCase, Rational ClearQuest and Rational Developer for System z can benefit your development process.

<0:16>

In this demonstration, our project lead, Lou, will set up a project work area where Alex, our developer, can make and test the changes and finally deliver them back to the project work area, where our project lead, Lou, will verify them and incorporate them into the application.

<0:32>

Our project manager, Lou, has received a request to change an application, so he logs into ClearQuest to create an activity record that we use during the change process. He is going to create an enhancement request and in that enhancement request, there are certain fields that are going to be required - they are indicated by the red labels. Also notice that each tab has a red dot until the fields that are required have been entered.

<0:59>

Once he has completed the enhancement request record and saved it, he is now ready to move on to create a project for managing the change.

<1:09>

Now Lou is going to create a project work area for this request. He will open the project explorer from the Rational ClearCase project window and within the project explorer he can see what existing projects there are. In this case there is an existing project he can reuse, a 'release 2' project that has an integration stream, a work area for the project and a developer work area called 'demo release 2'.

<1:37>

Now that the project has been created, Lou returns back to ClearQuest to find the record he has just created and begin to assign that to a developer. He locates the record, opens it up for assignment, fills in the required fields plus some additional information about which project he used, goes to the analysis tab because there are some required fields there, makes the assignment based on priority, and an owner which will be Alex, and apply that record. Now this record is available to Alex and it will be in his inbox.

IBM Software Demos
IBM Enterprise Modernization - Manage and Develop System z
Applications with Rational Developer for System z

<2:18>

Now let's observe Alex, the developer, as he makes the requested change. He opens up Rational Developer for System z, his integrated development environment. Within Rational Developer, he opens up a ClearQuest perspective so he can work with the activity record that has been assigned to him. He logs in to the ClearQuest database, and then displays a list of open and assigned activity records. Notice that record 216 is the one that Lou assigned as a high priority item. So Alex is ready to work on it - he opens it and changes its state to 'open' to indicate that he is ready to work on it now.

<3:07>

With the activity record open, Alex is now ready to create a project within RDz to assist with his changes to the code. He creates a local zOS project that will enable him to do local syntax checking against the Cobol code. He creates a project named 'Madrid' and maps it to the work area that Lou indicated would be used for this project – the 'demo release 2' view that is on the V drive. He will further map this to a specific folder within that view, to isolate only the code that he is interested in changing.

<3:47>

Using the integrated 3270 host connect facility, Alex will take a look at the application that needs some change. He double-clicks on the prerecorded macro that logs him into CICS and starts the application, and then double-clicks on the blue tab to open the window full screen. He enters in a date and then makes a selection, and he notices that the 'Days' field currently displays '888', which needs to be changed.

<4:20>

Using the integration with ClearCase, Alex now checks out the file to indicate that he is ready to begin working on it. Also, the integration with ClearQuest will prompt him for an activity record to associate with that change. He will browse the list of open requests and choose the activity record number 216 that applies to this change request.

IBM Software Demos
IBM Enterprise Modernization - Manage and Develop System z
Applications with Rational Developer for System z

<4:54>

By double-clicking on the filename he can open the file 'Madl02' in the editor, and by using the outline view, he can quickly navigate to the section of code he is interested in.

<5:08>

Expanding 'procedure division', he can click on 'calculate day difference' to have the editor move the current line to that section of the code. He notices that '888' is the reason for the field showing up in the test and that should have been 'w-day-difference', so he will make that change. He will delete '888' and he could copy in this field here with cut and paste, or in this case he could use content assist in which RDz will display a list of declared variables for the program from which he can select the one that applies.

<5:47>

So he makes that change, saves the file and then does a local syntax check. This performs a compile on his workstation, which he can use to verify that the code is correct before he moves it up to the mainframe to do testing there.

<6:03>

He clicks on the tab 'Remote Error List' to see what errors if any, and notices that they are all information messages, so he is ready now to test. Now that he's made his changes, Alex is ready to move the file to the host. So using the remote synchronization feature, the file is copied from ClearCase up to the mainframe and placed in a PDS. By double-clicking on the program name in the 'Remote Systems' view, he can verify in fact that the program is up there with the changes that he made.

<6:39>

Using RDz's build support, Alex will generate the JCL necessary to compile and link this into an executable and then submit the job for execution. To verify that the build job completed successfully, Alex uses another feature of RDz to look at the JES spool. He refreshes the list, opens it, and then selects the build job '8095'. Double-clicking opens it up in the editor area, where he can view the status by looking at the return codes and finally purge it from the list since it is not needed anymore. With the build successful, Alex is now ready to test. He

IBM Software Demos
IBM Enterprise Modernization - Manage and Develop System z
Applications with Rational Developer for System z

expands the CICS window, tells CICS that there is a new program that needs to be loaded and enters the transaction for this application. He puts in a new date and now observes that the 'Days' field reflects the new calculation, and so the test is successful.

<7:44>

With a successful test, Alex is now ready to deliver his changes to the project integration area. So he uses the Deliver Stream function of ClearCase and selects his work area which is 'demo release 2' and the delivery function will move his changes from 'demo release 2' to the project integration stream. Once all the changes have been moved and he's ready to confirm that integration, he will hit the 'Complete' button to finalize it.

<8:25>

Finally, Alex will update the Activity record. He switches to the ClearQuest perspective and changes the state from 'opened' to 'closed' and indicates that he has resolved this enhancement request. He applies the change and now that record will be updated as 'closed' indicating that Alex has completed his assignment.

<8:58>

Now that Lou has heard from Alex that he has completed his task, and Lou has had a chance to verify that in fact the application changes have been made, Lou goes back into ClearQuest to verify that the activity record has been closed. So using the 'All Enhancement Requests' view, he brings up a list of all known enhancement records, sorts them and looks for Alex's record, noting that it has been closed.

<9:24>

The final thing Lou will do is to go into the ClearCase project – the 'release 2 integration' stream - and make the changes that Alex has done a part of the new product by integrating them into the current baseline.

<9:46>

Once they have been made part of the latest baseline, Lou will also recommend that that is the recommended baseline for others to pick up when new changes need to be made.

END