

# ALM Integration

You can use this plugin to run ALM tests sets and use ALM Lab Management with the Jenkins continuous integration.

If you are working with Quality Center 10.00 or earlier, and QuickTest Professional 9.x or 10.x, use the [Quality Center Plugin](#).

If you are working with Quality Center 10.00 or earlier, and QuickTest Professional 9.x or 10.x, use the [Quality Center Plugin](#).

- [Configure the connection to your ALM Server](#)
- [Running Test Sets from ALM](#)
- [Running Server Side Tests Using ALM Lab Management](#)
- [Create/Update the AUT Environment Configuration Using ALM Lab Management](#)
- [Upload Test Results to ALM](#)

## Configure the connection to your ALM Server

1. Go to the Jenkins Server home page.
2. Click the **Manage Jenkins** link in the left pane.
3. In the Manage Jenkins Page click **Configure System**.
4. In the Configuration tab, scroll down to the **Application Lifecycle Management** section.
5. Click **Add ALM server**, if no server is currently configured.
6. Specify a meaningful server name and URL. When specifying an ALM Server URL, use the full string: <http://myserver.mydomain/qcbin>.

## Running Test Sets from ALM

### Set up a job

1. Go to the Jenkins Server home page.
2. Click the **New Job** link or select an existing job.
3. Enter a Job name (for a new job).
4. Select **Build a free-style software project** and click **OK**.
5. In the Project Configuration section scroll down to the **Build** section.
6. Expand the **Add build step** drop-down and select **Execute functional tests from ALM**.
7. Select one of the ALM servers that you configured in the previous step.
8. Enter the server credentials, project and domain. For ALM 14, use the API Key client ID and secret as the username and password. **Note:** If you are using the ALM scheduler, it will run under the Jenkins agent user. For example, if Jenkins is running as a **System** user, the scheduler will run the tests as a **System** user. This will not affect test execution.
9. Add the test set folders or specific test sets that you want to include, using the ALM path. To add a specific test, add the test name after the test set path. To add multiple entries, click the down arrow on the right of the field and enter each item on a separate line. See the example below.
10. Optionally, indicate a timeout in seconds after which the job will fail.
11. Click **Advanced** to indicate a Run mode (local, remote, or planned host) If you specify a remote host mode, specify a host name. This must be a machine with a valid installation of the testing tool.
12. Click **Apply** to save your changes and continue with more build steps. Click **Save** when you are finished adding build steps.

```
Root\testfolder1\testset_a
Root\testfolder1\testset_b
Root\testlab_folder
Root\testlab_folder\testset_a\test-name
```

### Set up the Post Build actions

1. In the **Post-build Actions** section, expand the **Add post-build action** drop-down and select **Publish test result**.  
**Run the job**
2. Run or schedule the job as you would with any standard Jenkins job.

### Review the results

1. From the dashboard, click on the job.
  2. Click the Console link to view the ALM information.
  3. Copy the ALM link to your Internet Explorer browser and view the Test Set results from within ALM.
-

# Running Server Side Tests Using ALM Lab Management

If you have Lab Management activated in ALM, you can run server-side tests from functional test sets and build verification suites. After setting up the test sets and build verification suites, you can configure a Jenkins build step to execute your tests.

## Set up a job

1. Go to the Jenkins Server home page.
2. Click the **New Job** link or select an existing job.
3. Enter a Job name (for a new job).
4. Select **Build a free-style software project** and click **OK**.
5. In the Project Configuration section scroll down to the **Build** section.
6. Expand the **Add build step** drop-down and select **Execute tests using ALM Lab Management**.
7. Select one of the ALM servers that you configured in the previous step.
8. Enter the server credentials, project, and domain.
9. If your ALM server version is 12.60 or higher, enter the **Client type**.
10. Select a Run Type from the drop down menu (functional test set or build verification suite).
11. Enter the ID of your run entity (either the test set ID or the build verification suite ID).
12. **Optional:** Enter a description of the build step.
13. Enter a duration (in minutes) for the timeslot. The minimum time is 30 minutes.
14. **Optional:** If you have defined an AUT environment configuration in ALM, you can enter the ID here in order to execute your timeslot with specific AUT parameters.  
If you have CDA configured in ALM and want to implement it for this time slot, select the **Use CDA for provisioning and deployment** checkbox and enter your CDA details.  
**Set up the Post Build actions**
15. In the **Post-build Actions** section, expand the **Add post-build action** drop-down and select **Publish test result**.

## Run the job

Run or schedule the job as you would with any standard Jenkins job.

## Review the results

1. From the dashboard, click on the job.
2. Click the Console link to view the ALM information.
3. Copy the ALM link to your Internet Explorer browser and view the Test Set results from within ALM.

**Note:** For environments with Internet Explorer 9 and ALM version 12.20 or later: The report, normally accessible through a link in the console log, will not be available.

---

# Create/Update the AUT Environment Configuration Using ALM Lab Management

If you have Lab Management activated in ALM, you can create/update an AUT Environment Configuration for an existing AUT Environment in ALM.

## Set up a job

1. Go to the Jenkins Server home page.
2. Click the **New Job** link or select an existing job.
3. Enter a Job name (for a new job).
4. Select **Build a free-style software project** and click **OK**.
5. In the Project Configuration section scroll down to the **Build** section.
6. Expand the **Add build step** drop-down and select **Execute AUT Environment preparation using ALM Lab Management**.
7. Select one of the ALM servers that you configured in the **Configure the connection to your ALM server** step.
8. Enter the server credentials, project name, and domain.
9. Enter the ID of the environment for which you want to create/update a configuration.
10. Select one of the following options to Indicate whether or not to create a new AUT Environment Configuration or update an existing one.
  - a. For **Create a new configuration named**, enter a name for the new configuration.
  - b. For **Use an existing configuration with ID**, enter the ID of your AUT Environment Configuration in ALM.
11. **Optional:** Enter a path for a JSON file that contains values for the AUT Environment parameters for the relevant configuration.
12. **Optional:** Enter a name of a build environment parameter in order to save the ID of the created/updated configuration for future use.
13. Add the AUT Environment parameters that you want to update for the created/updated configuration. For each parameter:
  - a. Select the type of the parameter from the drop-down menu (Manual, Environment, From JSON).
  - b. Enter the full path of the parameter as it appears in ALM.
  - c. Enter the value you want to assign to this parameter.

## Run the job

Run or schedule the job as you would with any standard Jenkins job.

---

# Upload Test Results to ALM

You can upload tests running in JUnit, NUnit, or TestNG frameworks to ALM as part of the run job or as a separate job. ALM will create tests and test sets (of External type) corresponding to the executed tests.

## Set up a job

1. Go to the Jenkins Server home page.
2. Click the **New Job** link or select an existing job.
3. Enter a Job name (for a new job).
4. Select **Build a free-style software project** and click **OK**.

## Set up the Post Build actions

1. In the Project Configuration section, scroll down to the **Post-build Actions** section.
2. Expand the **Add post-build action** drop-down and select **Upload test result to ALM**.
3. Select one of the ALM servers that you configured in the **Configure the connection to your ALM server** step.
4. Enter the server credentials, project name, and domain.
5. Select a testing framework from the drop-down list: JUnit, NUnit, or TestNG.
6. **Optional:** Enter the name of the Testing Tool name, to be used in the corresponding entity in ALM.
7. Enter the ALM Test folder path, in which to store the external tests. Do not include the Root test folder (Subject).
8. Enter the ALM Test Set folder path, in which to store the external test sets. Do not include the Root test sets folder.
9. Enter the Testing result file condition, relative to the root path of the job. For example, use `**/junitResult.xml` for Junit Plugin results.
10. **Optional:** Enter the Jenkins server URL.

## Run the job

1. Run or schedule the job as you would with any standard Jenkins job.

## Review the results

1. From the dashboard, click on the job.
2. Click the **Console** link to view the ALM information.
3. Copy the ALM link to Internet Explorer and view the tests, test sets, and test runs that were created within ALM.