

# Bumblebee HP ALM Plugin



# BUMBLEBEE

Bumblebee Jenkins plugin allows seamless integration of Jenkins jobs and build test results with [HP ALM 10, 11, 12](#). The plugin processes Jenkins build data and sends it to the Bumblebee server for processing. Bumblebee automatically creates TestPlan, TestLab, TestSet, and TestRuns in [HP ALM](#). Bumblebee plugin is ideal for organizations who want to reflect the data from Jenkins into their HP ALM project. Organizations use Bumblebee to show unit tests, Selenium test, Visual Studio tests, and various other test frameworks that run in Jenkins into some HP ALM project. Bumblebee is a commercial product. For more details, please go to <http://www.agiletestware.com/bumblebee>

## Plugin Information

View Bumblebee HP ALM [on the plugin site](#) for more information.



Older versions of this plugin may not be safe to use. Please review the following warnings before using an older version:

- [Unconditionally disabled SSL/TLS certificate validation](#)

## Why use Bumblebee plugin for HP ALM

The Bumblebee Jenkins plugin provides the ability to

- Easily configure your Jenkins jobs to export their test results directly into HP ALM using the Bumblebee web service
- No need to do anything on HP ALM. The plugin will create Test Plan, Test Labs, Test Sets, Test Run on the fly based on job configuration.
- Allows unit tests, Selenium tests, or any other tests that you run in Jenkins to be reflected in HP ALM. Having all results in one place allows project managers to always get a clear picture of the project risks without having to ask everyone for status.
- Support JUnit, TestNG, TRX, Cucumber, Serenity, JBehave result formats. Can add any custom result formats as needed.
- Trigger tests in HP ALM and report results back to Jenkins
- Run local HP UFT tests
- Start and monitor HP Performance Center tests execution

## Jenkins plugin Installation

Install the Bumblebee plugin for Jenkins using the Plugin manager. Please restart Jenkins after installing or upgrade the plugin.

The screenshot shows the Jenkins Plugin Manager interface. The 'Available' tab is selected, and a search filter 'ALM' is applied. The 'Bumblebee HP ALM Plugin' is highlighted with a red box, showing its version as 3.0.2. Below the table are buttons for 'Install without restart' and 'Download now and install after restart'.

Name	Version
<a href="#">Mack Security Realm Plugin</a> Defines an insecure "security realm" only good for testing/demoing other parts of the system, such as authorization strategy.	1.1
<a href="#">Script Security Realm</a> This plugin allows you to use a user-written custom script to authenticate the username and password	1.5
<a href="#">WWWPass Authentication Plugin</a> Plugin for testing Client-side authentication via WWW-Authenticate. This plugin requires an already existing valid user defined login session.	1.1
<input checked="" type="checkbox"/> <a href="#">Bumblebee HP ALM Plugin</a> This plugin integrates Jenkins with HP Quality Center 9, 10, 11, 12, ALM using <a href="#">Agiletestware's Bumblebee</a> product. The plugin updates HP ALM TestPlan, TestLab, TestSet by simple plugin configurations.	3.0.2
<a href="#">Multijob Plugin</a> This plugin, created by <a href="#">Tikal</a> ALM team, gives the option to define complex and hierarchical jobs structure in Jenkins.	1.15

# Installing Plugins/Upgrades

## Preparation

- Checking internet connectivity
- Checking update center connectivity
- Success

Bumblebee HP ALM Plugin  Success

➡ [Go back to the top page](#)  
(you can start using the installed plugins right away)

➡ Restart Jenkins when installation is complete and no jobs are running

## Prerequisite

The Bumblebee Jenkins plugin communicates with the Bumblebee web service which in turn communicates with HP ALM. You can download the Bumblebee server from the [Agiletestware website](#)

Detailed [user guide for installing Bumblebee server](#).

## Bumblebee Global Configuration

Configure the Bumblebee URL, HP ALM URL, username, and password.

BumbleBee Global Settings	
BumbleBee URL:	<input type="text" value="http://: /bumblebee"/> ?
HP ALM URL:	<input type="text" value="http:// 8080/qcbin"/> ?
HP PC URL:	<input type="text" value="http://"/> ?
HP ALM Login:	<input type="text" value="qcuser"/> ?
HP ALM Password:	<input type="password" value="....."/> ?
UFT Batch Runner:	<input type="text" value="c:\Program Files (x86)\HP\Unified Functional Testing\bin\UFTBatchRunnerCMD.exe"/> ?
Upload Timeout:	<input type="text" value="0"/> ?
PC Timeout:	<input type="text" value="0"/> ?
Skip connectivity diagnostic	<input type="checkbox"/> ?
Trust self-signed certificates	<input checked="" type="checkbox"/> ?

## Uploading test framework reports to HP ALM

Add the Bumblebee post build step in your job configuration. Specify ALM TestPlan and TestLab details. Bumblebee will automatically collect, parse, and insert test results in HP ALM.

Please refer to [the documentation for details](#)

localhost:8080/job/demo-maven-job/configure

Jenkins > demo-maven-job > configuration

### Post-build Actions

#### Bumblebee HP ALM Uploader

List of HP ALM Upload Configurations

Domain	DEFAULT	?
Project	demo	?
Test Plan	Subject/testPlan_demo	?
Test Lab	Root/testLab_Demo	?
Test Set	testSet_demo	?
Format	junit	?
Result File Pattern	**/target/surefire-reports/*.xml	?
Custom Properties	<small>Fileset 'includes' setting that specifies the generated raw XML report files</small>	?
Fail build if upload unsuccessful	<input type="checkbox"/>	?

**Delete**

**Add More**

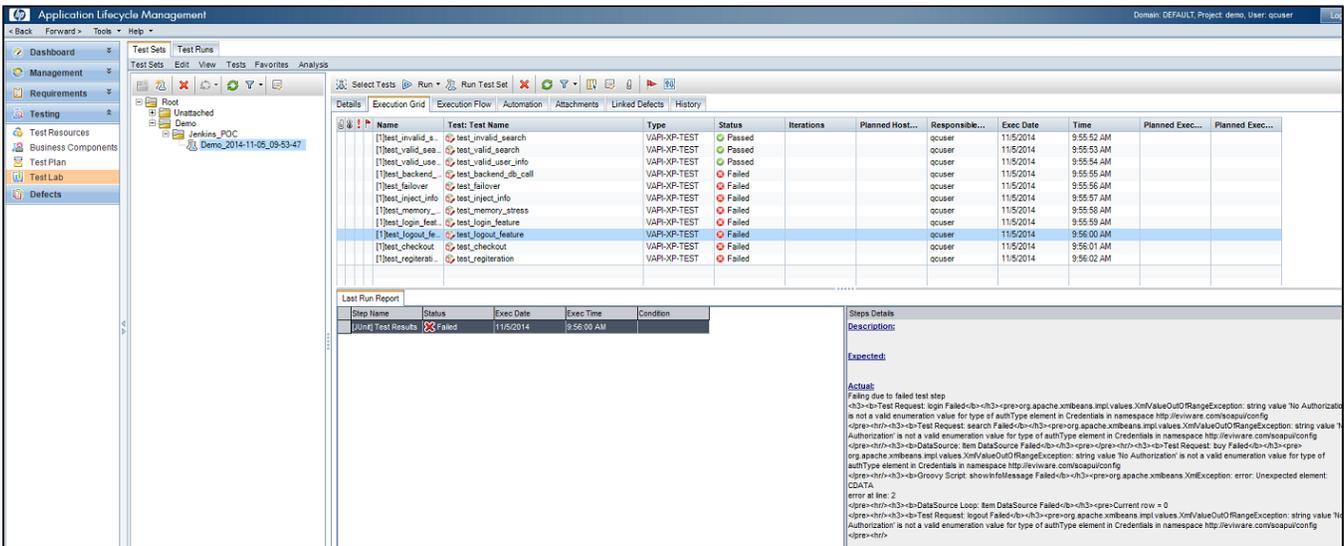
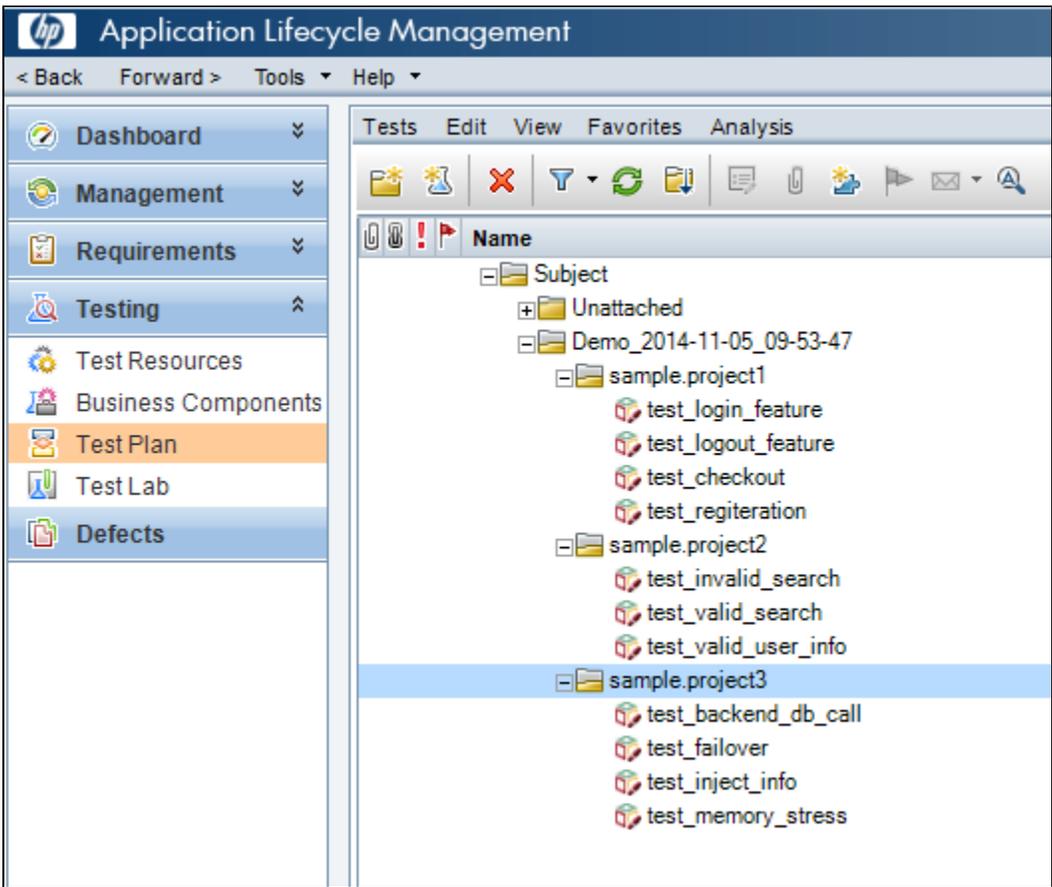
List of Upload Configurations

**Delete**

**Add post-build action**

**Save** **Apply**

HP ALM Results



## Running HP ALM tests from Jenkins

Bumblebee's Jenkins build step allows you to run HP ALM TestSets directly from Jenkins and view the results in both Jenkins and HP ALM.

### Pre-Requisites for running HP ALM tests from Jenkins

- Job must run on the Jenkins slave installed on Windows machine
- Jenkins slave must have launch method: Launch slave agents via Java Web Start
- Jenkins slave must NOT run as windows service
- Appropriate version of HP ALM Connectivity Tool must be installed on Jenkins slave machine. Tool is available at [http://your\\_alm\\_server\\_and\\_port/qcbin/Plugins/TDCConnectivity/TDCConnect.exe](http://your_alm_server_and_port/qcbin/Plugins/TDCConnectivity/TDCConnect.exe)
- Appropriate version of HP ALM Client must be installed on Jenkins slave machine. Available at [http://your\\_alm\\_server\\_and\\_port/qcbin/start\\_a.jsp?common=true](http://your_alm_server_and_port/qcbin/start_a.jsp?common=true)

To run test set from Jenkins you need to add `Bumblebee HP ALM Test Set Runner` build step to your build configuration

### Bumblebee HP ALM Test Set Runner

Domain  ?

Project  ?

Run Mode  ?

Run Host

Test Sets  ?

JUnit Results Directory  ?

**Required**

Execution Timeout  ?

Field	Description
Domain	The name of HP ALM domain
Project	The name of HP ALM project
Test Sets	A list of test sets to execute. Each test set path must start with new,line. Patch must start with Root and contains full path to the test set,in HP ALM TestLab. E.g. Root\folder1\testset1, where Root\folder1 is,TestLab folder and folder1 is the name of the test set to execute
Run Mode	How to run test sets. Possible values: LOCAL - run all tests on agent's,machine, SCHEDULED - run tests on planned host, REMOTE - run on remote,host
Run Host	The name of host on which tests shall be run. May be blank if Run Mode is LOCAL or SCHEDULED
JUnit Results Directory	Directory where JUnit-like execution reports will be placed. If it does not exist, Bumblebee will create it
Timeout	The number of minutes to wait for test sets execution. 0 means wait indefinitely.

When Jenkins runs `Bumblebee HP ALM Test Set Runner` step it connects to HP ALM server and triggers specified test sets.

Here is an example of build log produced by Bumblebee:

```

Building remotely on slave in workspace C:\temp\Jenkins\slave1\workspace\Bumblebee_plugin_demo\TestSet_Runner_Test
Bumblebee: running test sets
[bumblebee] $ C:\temp\Jenkins\slave1\bumblebee\testsetrunner.bat -project C:\temp\Jenkins\slave1\bumblebee\work\project4727232378091170142.xml -output
C:\temp\Jenkins\slave1\workspace\Bumblebee_plugin_demo\TestSet_Runner_Test\bumblebee_junit -timeout 0
Project XML file: C:\temp\Jenkins\slave1\bumblebee\work\project4727232378091170142.xml
Connecting to: http://[redacted]:8080/qcbin, User: qcuser, Domain: DEFAULT, Project: demo
Running test set: Root\Jenkins\testrunner
Start execution of test set: Root\Jenkins\testrunner
Run mode: LOCAL
Host name: 111-TOSH
Test: [1]run me, Execution status: WAITING, Message: Waiting...
Test: [1]run me, Execution status: UNKNOWN, Message: Host connected
Test: [1]run me, Execution status: RUNNING, Message: Running...
Test: [1]run me, Execution status: PASSED, Message: Passed
Execution has finished. Processing execution results
Test name: [1]run me, Message: Passed, Status: FinishedPassed
Writing C:\temp\Jenkins\slave1\workspace\Bumblebee_plugin_demo\TestSet_Runner_Test\bumblebee_junit\Root_Jenkins_testrunner_1467987908760.xml
Running test set: Root\Jenkins\testrunner1
Start execution of test set: Root\Jenkins\testrunner1
Run mode: LOCAL
Host name: 111-TOSH
Test: [1]fail, Execution status: WAITING, Message: Waiting...
Test: [1]fail, Execution status: CONNECTING, Message: Connecting...
Test: [1]fail, Execution status: UNKNOWN, Message: Host connected
Test: [1]fail, Execution status: RUNNING, Message: Running...
Test: [1]fail, Execution status: FAILED, Message: Failed
Test: [1]run me, Execution status: CONNECTING, Message: Connecting...
Test: [1]run me, Execution status: UNKNOWN, Message: Host connected
Test: [1]run me, Execution status: RUNNING, Message: Running...
Test: [1]run me, Execution status: PASSED, Message: Passed
Execution has finished. Processing execution results
Test name: [1]fail, Message: Failed, Status: FinishedFailed
Test name: [1]run me, Message: Passed, Status: FinishedPassed
Writing C:\temp\Jenkins\slave1\workspace\Bumblebee_plugin_demo\TestSet_Runner_Test\bumblebee_junit\Root_Jenkins_testrunner1_1467987960000.xml
BumblebeeTestSetRunner finished successfully
Some of tests failed, check JUnit reports
Return code: 1
ERROR: Test set execution failed. See logs for details.
Recording test results
Finished: FAILURE

```

Bumblebee HP ALM Test Set Runner puts results of test execution as a simple JUnit report which can be then published using Publish JUnit test result report step

## Test Result : (root)

1 failures (+1)

3 tests (+2)

Took 0 ms.

 [add description](#)

### All Failed Tests

Test Name	Duration	Age
<a href="#">testrunner1.[1]fail</a>	0 ms	1

### All Tests

Class	Duration	Fail	(diff) Skip	(diff) Pass	(diff) Total	(diff)
<a href="#">testrunner</a>	0 ms	0	0	1	1	
<a href="#">testrunner1</a>	0 ms	1	+1	0	1 +1	2 +2

## Running local HP UFT tests directly in Jenkins

Bumblebee allows you to run local [HP Unified Functional Testing](#) tests directly from Jenkins and reports results back to Jenkins.

### Prerequisites

- Bumblebee server v4.1.0 or higher

- Jenkins slave runs on Windows machine and have appropriate [HP Unified Functional Testing](#) version installed. Please see UFT requirements for a particular version of OS and other software.
- Jenkins slave must run as a console application (not as a windows service)
- UFT Batch Runner property of Global Configuration or UFT\_RUNNER environment variable on slave must be set

To override path to UFT Batch Runner, defined in the Global Configuration, you need to set a UFT\_RUNNER environment variable on a Jenkins slave.

To set a value to UFT\_RUNNER environment variable of Jenkins slave:

- Open Jenkins slave configuration page
- Check "Environment variables" checkbox
- Add a new variable and type "UFT\_RUNNER" as "Name" and path to UFT Batch Runner on that slave as a "Value"
- Click on "Save" button

Jenkins > Nodes > slave

[Back to List](#)  
[Status](#)  
[Delete Agent](#)  
[Configure](#)  
[Build History](#)  
[Load Statistics](#)  
[Log](#)

Build Executor Status

Name: slave

Description:

# of executors: 1

Remote root directory: C:\temp\Jenkins\slave

Labels: slave

Usage: Only build jobs with label expressions matching this node

Launch method: Launch agent via Java Web Start

Advanced...

Availability: Keep this agent online as much as possible

Node Properties

Environment variables

List of variables

Name	UFT_RUNNER
Value	c:\Program Files (x86)\HP\Unified Functional Testing\bin\UFTBatchRunnerCMD.exe

[Delete](#)

[Add](#)

Tool Locations

[Save](#)

## Adding and configuring "Bumblebee Local UFT Test Runner" build step

To add a new "Bumblebee Local UFT Test Runner" build step, just add a new build step in Jenkins build configuration with name "**Bumblebee: Run local UFT tests**".

"Bumblebee Local UFT Test Runner" build step has the following configuration parameters:

- Test Path - the path to a test folder or test batch file (.mtb) which shall be executed
- Results Directory - directory inside your project where Bumblebee put JUnit-like execution reports. If it does not exist, Bumblebee will create it automatically.

Bumblebee Local UFT Test Runner

Test path: test.mtb

JUnit Results Directory: uft-results

**Note: If you use .mtb file from GIT repository, you need to make sure paths to tests are correct and point to tests in build directory. You can use windows batch script for this.**

Bumblebee UFT step puts results of test execution as a simple JUnit report into folder defined by "Results Directory". These reports can be then published using Publish JUnit test result report Post-build step. Please note that it shall be configured to scan the output directory of Bumblebee UFT task.

## Build

**Bumblebee Local UFT Test Runner** X

Test path  ?

JUnit Results Directory  ?

Add build step ▾

## Post-build Actions

**Publish JUnit test result report** X ?

Test report XMLs  ?

Fileset 'includes' setting that specifies the generated raw XML report files, such as 'myproject/target/test-reports/\*.xml'. Basedir of the fileset is [the workspace root](#).

Retain long standard output/error ?

Health report amplification factor  ?

1% failing tests scores as 99% health. 5% failing tests scores as 95% health

Allow empty results  Do not fail the build on empty test results ?

Add post-build action ▾

When Jenkins runs Bumblebee UFT step, it will trigger local HP UFT Batch runner and record its output:

```

Bumblebee: Run local UFT test
Downloading license from bumblebee server
License has been installed
Downloading proxy settings from bumblebee server
Proxy settings file has been downloaded
[bumblebee] $ C:\temp\Jenkins\slave\bumblebee\testsetrunner.bat -mode uft -uftRunner "c:\Program Files (x86)\HP\Unified Functional Testing\bin\UFTBatchRunnerCMD.exe" -testPath
"C:\temp\Jenkins\slave\workspace\UFT runner test\" -outputDirPath "C:\temp\Jenkins\slave\workspace\UFT runner test\uft-results" -timeout 0
C:\temp\Jenkins\slave\workspace\UFT runner test\APITest1\APITest1.st
C:\temp\Jenkins\slave\workspace\UFT runner test\UITest1\Test.tsp

-----APITest1-----
Test is executing
Log Message:Step 'Start' : Step 'Start' started
Step StartActivity1 is executing
Step StartActivity1 is completed
Log Message:Step 'Start' ended successfully
Log Message:Step 'Test Flow' : Step 'Test Flow' started
Step Loop2 is executing
Log Message:Step 'Iteration 1' : Step 'Iteration 1' started
Step Sequence4 is executing
Step Sequence4 is completed
Log Message:Step 'Iteration 1' ended successfully
Step Loop2 is completed
Log Message:Step 'Test Flow' ended successfully
Log Message:Step 'End' : Step 'End' started
Step EndActivity3 is executing
Step EndActivity3 is completed
Log Message:Step 'End' ended successfully
Test is completed

Run results can be found at 'C:\temp\Jenkins\slave\workspace\UFT runner test\APITest1\Report'

-----GUITest1-----
The "Flights 1" Link object was not found in the Object Repository.
Check the Object Repository to confirm that the object exists or to find the correct name for the object.
The "Flights 1" Link object was not found in the Object Repository.
Check the Object Repository to confirm that the object exists or to find the correct name for the object.
Run results can be found at 'C:\temp\Jenkins\slave\workspace\UFT runner test\GUITest1\Report\Report'
Return code: 0
Recording test results
Build step 'Publish JUnit test result report' changed build result to UNSTABLE
Archiving artifacts
Finished: UNSTABLE

```

## Checking build report

Bumblebee UFT step captures results of test execution and produces a simple JUnit report which are then attached to the build report and can be seen on "Test Results" page:

## Test Result

1 failures

2 tests  
[Took 2 min 28 sec.](#)  
[add description](#)

### All Failed Tests

Test Name	Duration	Age
<a href="#">+ GUITest1.GUITest1</a>	2 min 28 sec	1

### All Tests

Package	Duration	Fail	(diff) Skip	(diff) Pass	(diff) Total	(diff)
<a href="#">(root)</a>	2 min 28 sec	1	+1	0	1 +1	2 +2

For failed tests, report contains an error message reported by UFT:

# Failed

GUITest1.GUITest1 (from uft.GUITest1)

Failing for the past 1 build (Since  #2 )

[Took 2 min 28 sec](#)

 [add description](#)

## Stacktrace

Errors:

Step: The "Home 1" Link object was not found in the Object Repository., Error: The "Home 1" Link object was not found in the Object Repository.

Check the Object Repository to confirm that the object exists or to find the correct name for the object.

Step: The "Flights 1" Link object was not found in the Object Repository., Error: The "Flights 1" Link object was not found in the Object Repository.

Check the Object Repository to confirm that the object exists or to find the correct name for the object.

UFT also produces detailed reports with description of all steps, screenshots, etc...

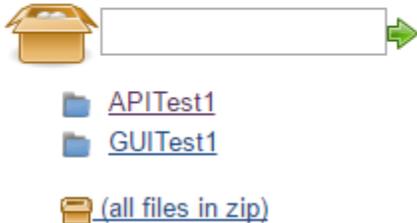
You can setup Jenkins to capture those results as build artifacts and attach them to a build results.

To do that, you just need to add "Archive artifacts" Post-build step to your build configuration and define appropriate value for "Files to archive".  
e.g.:



After build has finished, artifacts are displayed on "Artifacts" tab:

## Artifacts of UFT runner test #2



## Running HP Performance Tests from Jenkins

*HP Performance Center is a powerful set of tools for composing and running performance tests which is used by many companies.*

*Bumblebee offers Jenkins users ability to easy trigger Performance Center tests and report results back to Jenkins.*

When the task starts it triggers a new test run in PC and then polls run status from time to time. When run reaches some terminal state or timeout is reached then task is finished.

Terminal states are:

- Finished
- Before Collating Results (if Post Run Action = Do Not Collate)
- Before Creating Analysis Data (if Post Run Action = Collate Results)
- Canceled
- Run Failure
- Aborted
- Failed Collating Results
- Failed Creating Analysis Data

If run finished successfully, all test results are downloaded into specified folder in the build working directory.

## Prerequisites

- Bumblebee server version 4.1.0 or higher

## Global configuration

To start working with the new task the following [Bumblebee Global Settings](#) should be set:

- Bumblebee URL - URL of Bumblebee server
- HP ALM URL - URL of HP ALM
- PC URL - URL of a Performance Center
- HP ALM user name - user name to connect to HP ALM and Performance Center
- HP ALM password - password for HP ALM and Performance Center
- PC timeout (optional) - the number of minutes to wait for the PC test to finish. 0 means wait indefinitely.

## Adding and configuring of "Bumblebee HP PC Test Runner" task

To add a new "Bumblebee HP PC Test Runner" build step, just add a new build step in Jenkins build configuration with name "**Bumblebee HP PC Test Runner**".

"**Bumblebee HP PC Test Runner**" build step has the following configuration parameters:

Parameter name	Description
Domain	Domain name in HP ALM.
Project	Project name in HP ALM.
Results Directory	Directory to which test result files will be saved.
Path To Test	Path to a test in HP ALM TestPlan, e.g. "Subject\folder1\test", where "Subject\folder" is a path to a test folder and "test" is the name of a test to run.
Test Set	Path to a test set in HP ALM TestLab, containing correspondent test instance, e.g. "Root\folder1\testSet", where "Root\folder1" is a path to a test lab folder and "testSet" is the name of a test set. If test set does not exist or test is not assigned to it, Bumblebee task will try to create a new test set and assign a test to it.
Post Run Action	Defines what PC should do after a test run. Available options: Collate And Analyze, Collate Results and Do Not Collate.
Time Slot Duration	Time to allot for the test (PC parameter). It cannot be less than 30 minutes (limitation by PC).
Use VUD Licenses	If true, the test consumes Virtual User Day (VUD) licenses.
Timeout	Overrides a global PC timeout value and represents the number of minutes to wait for the Performance Center test to finish. 0 means wait indefinitely.
Retry Attempts	Number of retry attempts, before task completely fails.
Retry Interval	Number of seconds to wait between retry attempts.
Interval Increase Factor	Increase factor for retry interval. E.g. if it is set to 2, then each subsequent wait interval between attempts will be twice bigger than the previous one.
Polling Interval	The number of minutes between two test state requests.
Fail Build If Task Fails	If true and task has failed (or timeout has reached), then the whole build will be failed. If false, then build will not be failed even if task has failed.

**Bumblebee HP PC Test Runner** X

Domain	<input type="text" value="default"/>	?
Project	<input type="text" value="demo"/>	?
Results Directory	<input type="text" value="pc-results"/>	?
Path To Test	<input type="text" value="Subject\pc\test"/>	?
Test Set	<input type="text" value="Root\pc\test"/>	?
Post Run Action	<input type="text" value="Collate And Analyze"/>	?
Time Slot Duration	<input type="text" value="30"/>	?
Use VUD Licenses	<input type="checkbox"/>	?
Timeout	<input type="text" value="0"/>	?
Retry Attempts	<input type="text" value="2"/>	?
Retry Interval	<input type="text" value="5"/>	?
Interval Increase Factor	<input type="text" value="2,0"/>	?
<input checked="" type="checkbox"/> Retry Collate/Analyze		?
Collate/Analyze retry attempts	<input type="text" value="1"/>	?
Collate/Analyze retry interval	<input type="text" value="5"/>	?
Polling Interval	<input type="text" value="1"/>	?
Fail Build If Task Fails	<input checked="" type="checkbox"/>	?

## Attaching PC results as Jenkins build artifacts

Since, Performance Center produces some test reports, Bumblebee task downloads them from the PC server and stores into Results Directory, defined in a build configuration.

To see those reports on Jenkins build page, they need to be attached as build artifact, so before running the build, Jenkins should be configured to capture and archive required artifacts.

To do that, you just need to add "Archive artifacts" Post-build step to your build configuration and define appropriate value for "Files to archive".

Results Directory	<input type="text" value="pc-results"/>	?
Path To Test	<input type="text" value="Subject\pc\test"/>	?
Test Set	<input type="text" value="Root\pc\test"/>	?
Post Run Action	<input type="text" value="Collate And Analyze"/>	?
Time Slot Duration	<input type="text" value="30"/>	?
Use VUD Licenses	<input type="checkbox"/>	?
Timeout	<input type="text" value="0"/>	?
Retry Attempts	<input type="text" value="2"/>	?
Retry Interval	<input type="text" value="5"/>	?
Interval Increase Factor	<input type="text" value="2,0"/>	?
<input checked="" type="checkbox"/> Retry Collate/Analyze		?
Collate/Analyze retry attempts	<input type="text" value="1"/>	?
Collate/Analyze retry interval	<input type="text" value="5"/>	?
Polling Interval	<input type="text" value="1"/>	?
Fail Build If Task Fails	<input checked="" type="checkbox"/>	?

## Post-build Actions

**Archive the artifacts** X

Files to archive	<input type="text" value="pc-results/*"/>	?
------------------	---	---

## Running the task

When task is triggered it starts a new run in HP Performance Center for a test specified by "Path To Test" and "Test Set" properties of the task. The following parameters affect test execution in PC:

- Post Run Action
- Time Slot Duration
- Use VUD Licenses

Please refer to HP Performance Center documentation for detailed description of these parameters.

After test is started, task waits for it to finish and polls run state from time to time ("Polling Interval"). If test reaches one of the following states, Bumblebee assumes that test has passed:

- Finished
- Before Collating Results (if Post Run Action = Do Not Collate)
- Before Creating Analysis Data (if Post Run Action = Collate Results)

If test reaches one of the following states or timeout has occurred, Bumblebee treats test as failed:

- Canceled
- Run Failure
- Aborted
- Failed Collating Results
- Failed Creating Analysis Data

If test has failed Bumblebee makes a decision on whether build shall be failed or not based of value of "Fail Build If Task Fails" property. If it is true, then the whole build is failed. If it is false, then Jenkins does not fail and proceeds with the next task.

If an error occurs during fetching runs status from PC, Bumblebee will try to retry failed action according to the retry settings defined for a task.

Here is an example of the execution log:

```
Started by user anonymous
Building on master in workspace C:\Program Files (x86)\Jenkins\workspace\Performance Center tests
Start Performance Center test
Parameters:
Bumblebee URL: http://[redacted]/bumblebee
HP ALM URL: http://[redacted]/qcbn
HP ALM user: qcuser
HP ALM Domain: DEFAULT
HP ALM Project: demo
Task timeout: 5 minute(s)
Test path: Subject\PC\test
Test Set path: Root\PC\test set
Post Run Action: Collate Results
Timeslot Duration: 30 minute(s)
Use VUD licenses: false
Fail build if task fails: true
PC polling interval: 40 second(s)
Starting PC test: test path: Subject\PC\test, test set path: Root\PC\test set
Run: ID: 42 has started
Waiting for run completion. Run ID: 42
Fetching run state from Performance Center
Run state: Running
Fetching run state from Performance Center
Run state: Finished
Run has finished
Fetching test results from Performance Center. Run ID: 42
Fetching result file: Name: non-junit1496425093108.xml, type: NOT_JUNIT
Result file was stored to: C:\Program Files (x86)\Jenkins\workspace\Performance Center tests\pc-results\non-junit1496425093108.xml
Fetching result file: Name: junit1496425093108.xml, type: JUNIT
Result file was stored to: C:\Program Files (x86)\Jenkins\workspace\Performance Center tests\pc-results\junit1496425093108.xml
Run with ID: 42 has finished with state: Finished
Run duration: 2 minutes
Archiving artifacts
Finished: SUCCESS
```

## Pulling test results from HP ALM

If you want to pull test results from Jenkins and display them as JUnit report of your build, you can use "Bumblebee: Import HP ALM Test Results" step.

## Prerequisites

- Bumblebee server version 4.1.5 and higher

## Configure Import HP ALM Test Results step

Parameter name	Description
Domain	Domain name in HP ALM
Project	Project name in HP ALM
Login	User name in HP ALM. If it is set, it will override global settings
Password	Password in HP ALM. If it is set, it will override global settings
Results Directory	Path to the directory where to put JUnit-like reports containing results of tests in HP ALM
Test Set Path	Path to a TestSet in HP ALM TestLab to pull results from it

The screenshot shows the configuration window for 'Bumblebee: Import HP ALM Test Results'. It contains several input fields and a list of Test Set Paths.

- Domain:** DEFAULT
- Project:** project
- Login:** (empty)
- Password:** (empty)
- Results Directory:** alm-results
- Test Results Configuration:**
  - Test Set Path: Root\folder1\folder2\testSet1
  - Test Set Path: Root\folder1\folder2\testSet2
  - Buttons: Add More, X, ?

## Execution

During the execution of "Bumblebee: Import HP ALM Test Results" test step, Bumblebee searches for a Test Set by path given in "Test Set Path" parameter, creates JUnit XML report file and puts it into "Results Directory" folder. This folder can be used by JUnit publisher to build test trends.

## Changelog

### Version 4.1.4 (released September, 2019)

- Add "Trust to self-signed certificates" checkbox
- Replace Jersey library with Apache HTTP Client

### Version 4.1.3 (released July, 2019)

- Fix bug with failure on save of "Import HP ALM Test Results" build step

### Version 4.1.2 (released February, 2019)

- Update dependencies

### Version 4.1.1 (released January, 2019)

- Add support of Serenity reports
- Add support of JBehave reports
- Add ability to skip connectivity diagnostic
- Add ability to override PC user/password

### **Version 4.1.0 (released April, 2018)**

- Support protractor-jasmine reports
- Add retry settings for collate/analyze phase
- Improve logging
- Bugfixes

### **Version 4.0.9 (released February, 2018)**

- Pull results from HP ALM
- Improve logging

### **Version 4.0.8 (released October, 2017)**

- Fix possible memory leak

### **Version 4.0.7 (released September, 2017)**

- Bug fixes

### **Version 4.0.6 (released July, 2017)**

- Add new "Bumblebee: Add Test to Test Set" step for creating TestSet instances in HP ALM TestLab and adding tests from TestPlan to them

### **Version 4.0.5 (released June, 2017)**

- Running local HP UFT tests from Jenkins
- Running HP Performance Center tests from Jenkins

### **Version 4.0.4 (released August, 2016)**

- Use Jenkins proxy settings to communicate with Bumblebee and HP ALM server
- Bug fixes

### **Version 4.0.3 (released July, 2016)**

- Bug fixes

### **Version 4.0.2 (released July, 2016)**

- Support for Bumblebee server version 4.0.4
- Asynchronous processing of test reports ([docs](#))
- Running HP ALM tests from Jenkins ([docs](#))
- Add support of Cucumber reports
- Add support of FitNesse reports

### **Version 4.0.1 (released February, 2016)**

- Downgrade to Java 1.6 to support older versions of Jenkins

### **Version 4.0.0 (released October, 2015)**

- Support for Bumblebee server version 4.0
- Usability improvements

### **Version 3.0.2 (released November, 2014)**

- Updated Jenkins dependency version
- Fixed minor bug with license checker logic.

### **Version 3.0.0 (released November, 2014)**

- Bumblebee releases version 3.0