

ALM Administration Guide

Version: 1.8.10 - Approved

Prepared by: Paul Noffke

www.provartesting.com

Contents

Introduction	3
Prerequisites.....	3
Server-side installation.....	3
Client side installation.....	6
Uninstalling ALM Plugin	8
<i>Server-side uninstall</i>	8
<i>Client side uninstall</i>	8
Provar Customisation.....	10
<i>Provar Settings</i>	10
<i>Custom Fields</i>	11
<i>Execution Requirements</i>	12
<i>Expressions</i>	12
<i>REST Field Names</i>	14

Document History

Date	Author	Version	Changes	Approved By
21 Feb 2017	PN	1.8.10	Initial version	
23 Feb 2017	PN	1.8.10	Added Adobe PDF Reader prerequisite.	
27 Feb 2017	PN	1.8.10	Clarified Custom Field impact	

Introduction

This document explains how to install and configure the Provar plug-in for HP’s Application Lifecycle Management tool.

Prerequisites

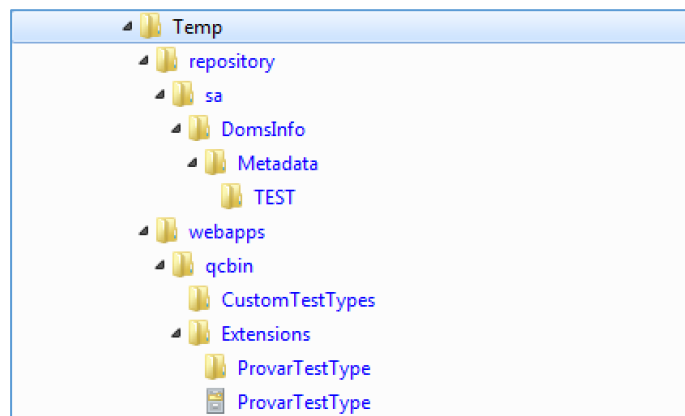
- ALM should be running.
- IE (10 and above) should be installed on both the client and the server.
- User should have Admin rights.
- DotNet framework 4.0 and above should be installed on both server and client.
- The Adobe PDF Reader plug-in must be installed in Internet Explorer on client workstations.
- On the client side the server path should be listed on IE trusted sites.

Server-side installation

Step 1. Unzip installation files:

Unzip the Provar_ALM_<alm-version>_<version>.zip to a temporary folder. This folder will be referred to as <dist-folder> in the steps below.

Assuming “Temp” is the unzipped folder, the folder structure will be as below



Step 2. Locate the ALM folder on the server:

The default location is C:\ProgramData\HP\ALM. This will be referred as <alm-folder> in the steps below.

Step 3. Copy the files:

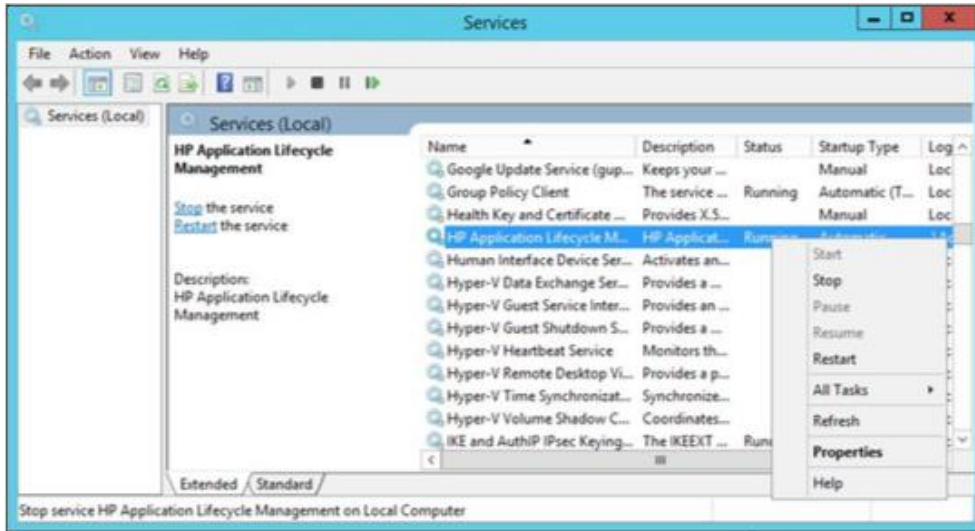
Copy the following files from <dist-folder> to the <alm-folder>

From	To
webapps\qcbin\CustomTestTypes\testtypes.xml	webapps\qcbin\CustomTestTypes\testtypes.xml*
webapps\qcbin\Extensions\ProvarTestType.cab	webapps\qcbin\Extensions\ProvarTestType.cab
webapps\qcbin\Extensions\ProvarTestType*	webapps\qcbin\Extensions\ProvarTestType*
repository\sa\DomsInfo\Metadata\TEST*	repository\sa\DomsInfo\Metadata\TEST*

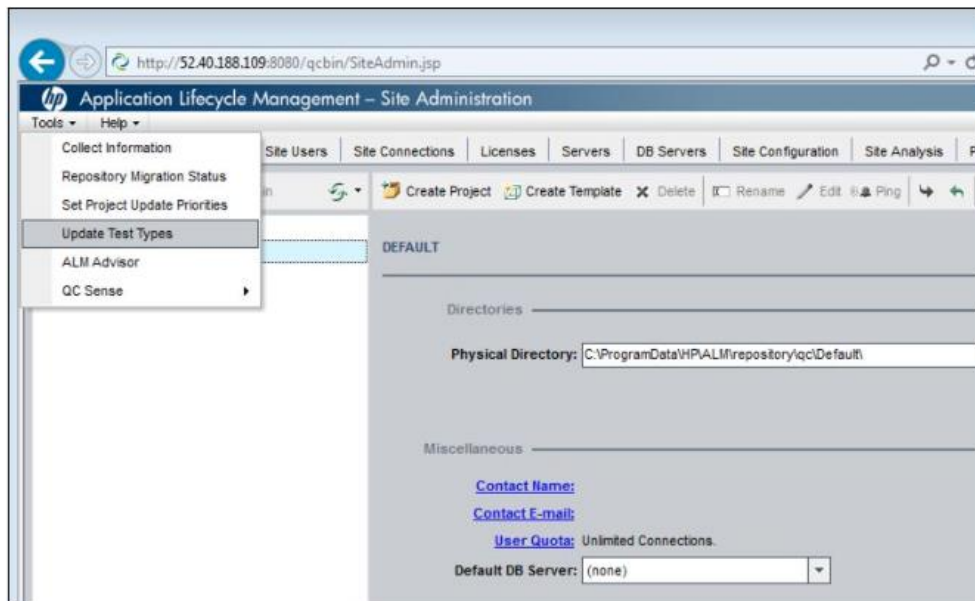
* **Note:** The contents of CustomTestTypes\testtypes.xml must be merged if the file already exists.

Step 4. Restart ALM:

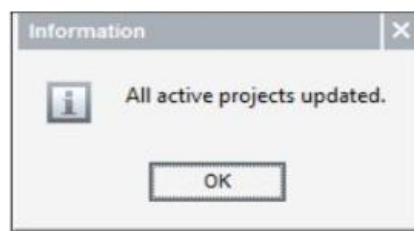
From the services control panel entry (services.msc), restart the ALM service.



Step 5. Update the Test Types via the ALM site Admin.



Choose YES at the confirmation dialog and you should see:

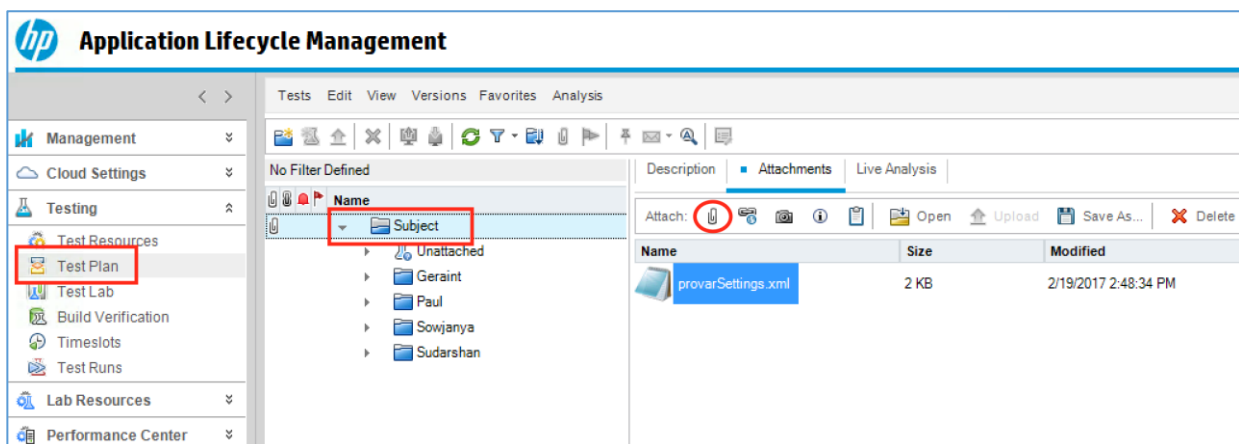


Step 6. Enable Provar for ALM Projects

Carry out the following procedure for each ALM project that Provar must be enabled for:

Customise the “provarSettings.xml” from the Provar ALM Installation zip folder to suit your requirements. See

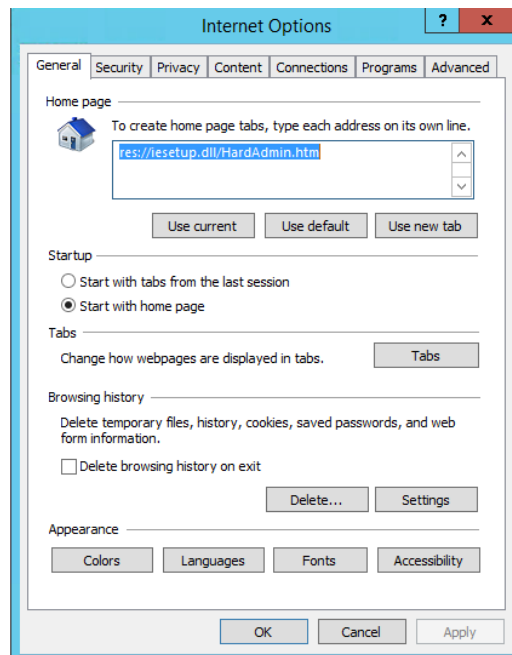
1. Provar Customisation on page 10 for details.
2. Logon to the ALM project via the ALM client (qcbn/start_a.jsp).
3. Open the Test Plan tab via the left-hand navigation panel and select the top-level ‘Subject’ node in the tree. Use the paperclip tool item on the Attachments tab to attach your customized provarParams.xml file to the Subject node.



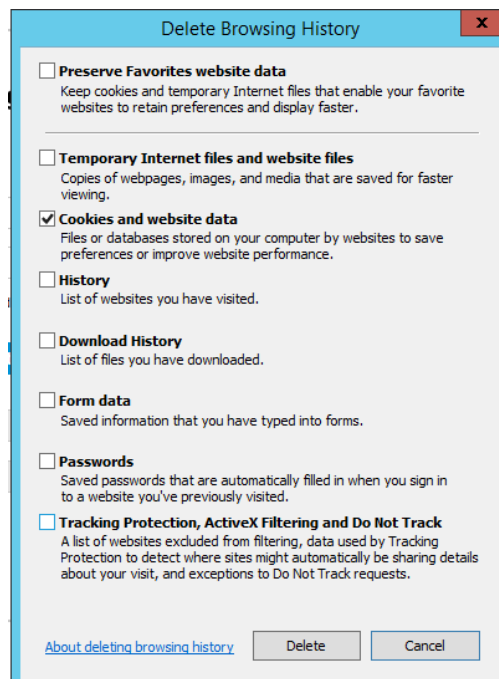
Client side installation

Step 1. Clear Internet explorer cache

For this click on IE tools on the top right corner of IE screen and select Internet options. The following screen is shown:



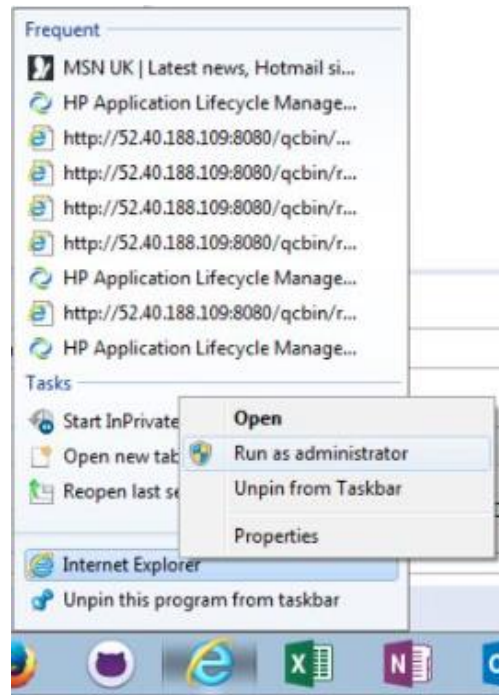
In the Browsing history section, click Delete button and the following screen displays



Check mark 'Cookies and website data' and click the Delete button.

Step 2. Update ALM client

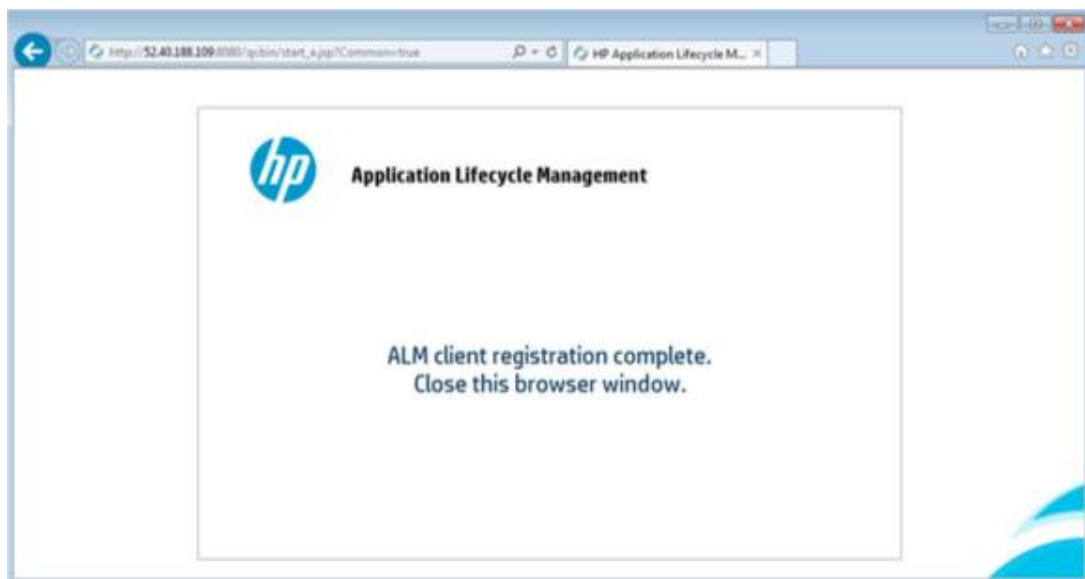
Start Internet Explorer in 'Run as Administrator' mode.



Navigate to the following URL:

`http://<alm-server>\qcbn\start_a.jsp?Common=true`

This will download ALM client components from the server and display the following when finished. Close the browser and open IE and enter the ALM server path.



If prompted, agree to install any component published by Make Positive Provar Limited.

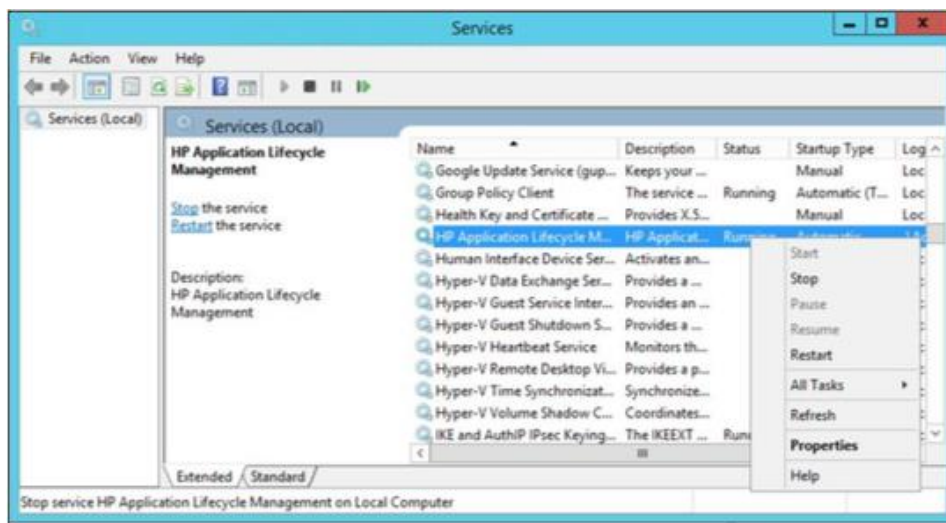
Uninstalling ALM Plugin

Server-side uninstall

Step 1. Delete the following.

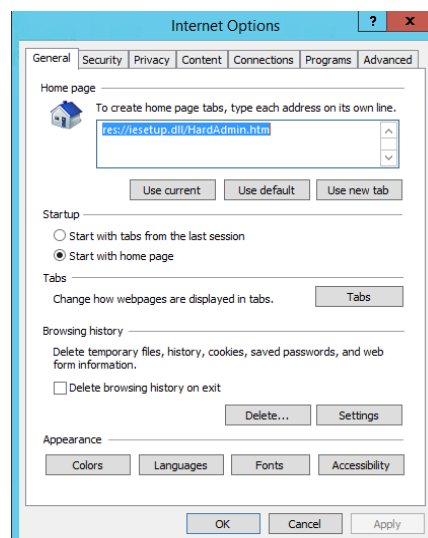
```
webapps\qcbn\CustomTestTypes(delete folder)
webapps\qcbn\Extensions\ProvarTestType.cab
webapps\qcbn\Extensions\ProvarTestType(delete folder)
repository\sa\DomsInfo\Metadata\TEST\ProvarTestType.xml
```

Step 2. Restart ALM server.

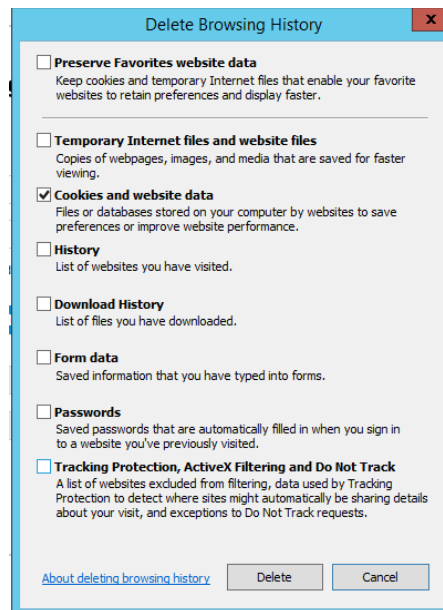


Client side uninstall

Step 1. Clear Internet explorer cache. For this click on IE tools on the top right corner of IE screen and select Internet options. The following screen is shown:



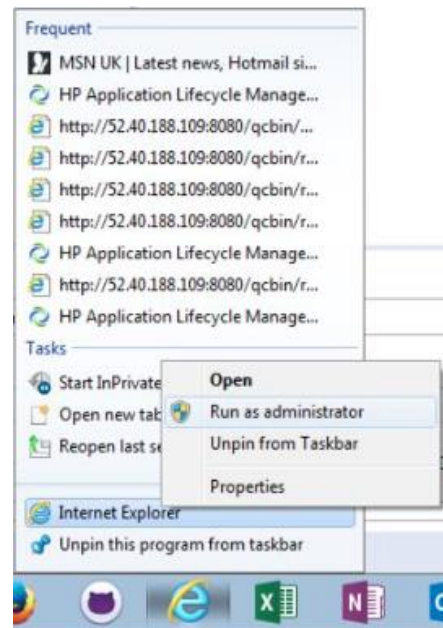
In the Browsing history section, click Delete button and the following screen displays



Check mark 'Cookies and website data' and click the Delete button.

Step 2. Update ALM client

Start Internet Explorer in 'Run as Administrator' mode.



Navigate to the following URL:

`http:\<alm-server>\qcbn\start_a.jsp?Common=true`

This will download existing ALM client components from the server and display the following when finished. Close the browser and open IE and enter the ALM server path

Provar Customisation

Provar's ALM plug-in has several options and settings that can be set via the provarParams.xml file.

This file is attached to the Subject node in ALM's Test Plan, allowing settings to be customized for each ALM Project.

A sample provarParams.xml file is included in the .zip file and can be used as a starting point.

Provar Settings

Setting	Values	Notes
allowCheckedOutExecution	true false	Whether provarParams should allow Tests that are not checked-in to be executed against ALM.
checkInAfterExport	true false	Whether newly added Tests should be checked-in in ALM. Defaults to true.
customField	See below	One or more <customField> child elements. These are used to specify the initial values for any mandatory custom fields that are already present in ALM.
disableCheckOut	true false	If true then "Check-out files" checkbox in the Download from ALM dialog is disabled in Provar. Intended for customers with ALM customisations that require checkouts to be done from inside ALM. Defaults to false.
disableCheckIn	true false	If true then "Check-in files" checkbox in the Upload to ALM dialog is disabled in Provar. Intended for customers with ALM customisations that require checkins to be done from inside ALM. Defaults to false.
executionRequirement	See below	One or more <executionRequirement> child elements. These specify requirements that Tests and/or Test Sets need to meet in ALM before test execution is allowed.
initialExportStatus	As per ALM's Status field	The Status value that newly added Tests should have. Defaults to Imported.
testType	ProvarTestType MANUAL	The Test Type that should be used when adding tests to ALM. Defaults to ProvarTestType

Custom Fields

Custom Fields allow you specify default values for fields in your ALM entity schema They are required if:

- You have added mandatory custom fields to one or more entities.
- You have changed a system field from optional to mandatory.
- You would like to default the value of a field (regardless of whether it is mandatory or not).

NOTE: Provar does not add or modify fields in your ALM entity schema. Custom Fields support customizations that have already be made.

Setting	Values	Notes
entity	Test Resource TestSet	The type of ALM entity that the field belongs to. Defaults to Test.
name		The internal name of the custom field which must already exist inside ALM. See REST Field Names below for details of how to derive the correct value from the ALM entity schema.
initialValue		The value that Provar should populate. May contain Expressions (see below).

The following will default the name of the TS_USER_01 field in ALM schema:

```
<customField name="user-01" entity="Test" initialValue="Y"/>
```

Execution Requirements

Setting	Values	Notes
testQuery	ALM query language expression	The query that is run against the Tests that have been selected for execution.* The Test IDs of the selected Tests are automatically appended to this query. Optional
testSetQuery	ALM query language expression	The query that is run against the Test Set that the user selects.* Optional
disqualifying	true false	If true then the execution is disallowed if the query returns any matches. If false then the execution is only allowed if all the selected Tests are returned by the query.
message	text	The message that is displayed to the user is the execution is not allowed.
remedy	text	A suggested remedy that is displayed to the user if the execution is not allowed.

* See [List Filtering in the ALM REST Guide](#) for the query syntax.

The following example will prevent execution if the Test Set has a user-01 value of 'N' and any of the selected Tests has a status that is not Ready:

```
<executionRequirement testSetQuery="user-01['N']" testQuery="status[NOT Ready]"
disqualifying="true" message="..." remedy="..."/>
```

Expressions

The initialValue attributes in <customField> elements can contain Provar expressions to allow values to be calculated or sourced from related ALM Entities.

This is useful in the following situations:

- Where a value needs to be sourced from a related entity (e.g. Test or Test Set).
- Where a date need to be included. For instance, including the execution start time in the Run name.

Expressions are enclosed in curly braces as per the normal convention in Provar. The following expression will derive the Run's Name from the Test's name and execution start time.:

```
<customField name="Name" entity="Run"
initialValue="Run_{Test.Name}_{DateFormat(ExecutionItem.startedDate, 'yyyy-MM-dd')}'/>
```

NOTE: Expressions are currently only supported for the Run entity. **No substitutions are done for other entity types.**

ExecutionItem

Exposes information about the Test execution:

Field	Format	Description
startedDate	Date/Time	The time that the Test execution started. Can be formatted using the DateFormat function.
endedDate	Date/Time	The time that the Test execution ended. Can be formatted using the DateFormat function.

Test, TestSet

Exposes fields from the related Test or Test Set

Field	Format	Description
Name	String	The name of the Test or Test Set
*	*	Any other field from the Test or Test Set entities in ALM, including and custom fields. Must be specified in REST format (see below). If the name contains a hyphen, then the following syntax must be used: <code>entity['field-name']</code> For example, for the TS_USER_01 custom field: <code>Test['user-01']</code>

DateFormat(date, format, timeZone)

Parameter	Format	Description
date	Date/Time	The date/time to be formatted, or NOW for the current date/time.
format	String	A valid Java date format string. See https://docs.oracle.com/javase/7/docs/api/java/text/SimpleDateFormat.html .
timeZone	String	A valid Java time zone. See https://docs.oracle.com/javase/7/docs/api/java/util/TimeZone.html Defaults to GMT.

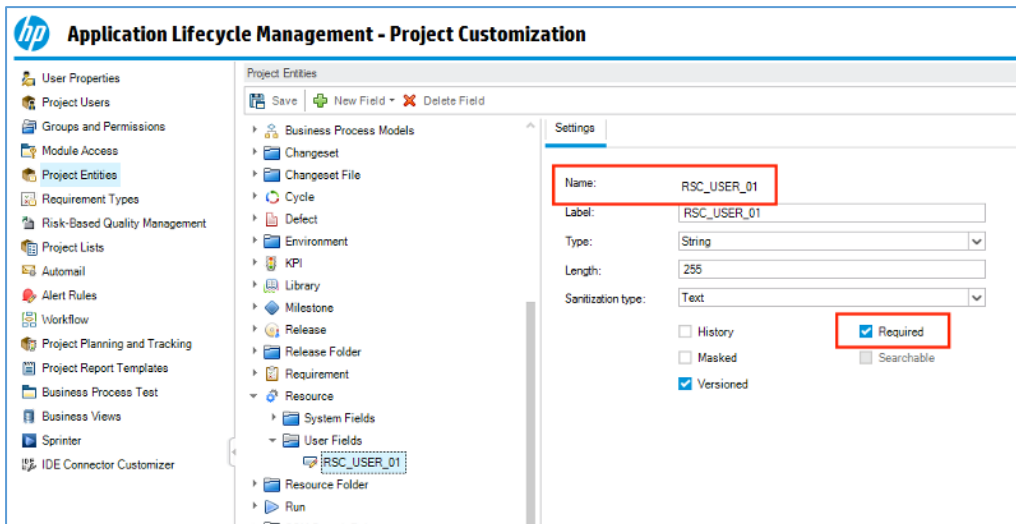
The following example formats the Execution start date/time to the Warsaw time zone:

```
{DateFormat(ExecutionItem.startedDate, 'yyyy-MM-dd_HH:mm', 'Europe/Warsaw')}
```

REST Field Names

The <customField> and <executionRequirement> elements in provarParams.xml use REST format names for ALM fields. This section outlines the procedure for determining these names from ALM.

The ALM names of the fields can be found in the Project Entities panel in ALM's Project Customisation screens:



Given an ALM field name:

1. Remove the RSC_, TS_, CY_ or RN_ prefix from the field name (depending on the entity type)
2. Change all letters to lower case and all underscores (_) to dashes (-).

For example, "RSC_USER_01" becomes "user-01".