



Polarion® Application Lifecycle Management Platform

Examples Supplement

Version: **2015 SR3**

Polarion Application Lifecycle Management Platform Administrator's Guide

Copyright © 2005-2015 Polarion Software

Permission is granted to use, copy, and distribute this document provided that the content is not modified and the above copyright notice and this statement appear in all copies.

Polarion is a registered trademark of Polarion Software. Polarion ALM, Polarion REQUIREMENTS, Polarion QA, Polarion VARIANTS, and Polarion PRO are trademarks or registered trademarks of Polarion Software.

Other trademarks referenced are property of their respective owners.

Updated: 17-Dec-2015

Polarion Listing & Examples Supplement

This document is a supplement to the Polarion PDF Manuals, which are generated from the online Help content. Some code listings and examples are truncated in the PDF output. This supplemental document provides these examples in full so that they can be copied if necessary. Subsequent pages are in landscape orientation.

CONTENTS

Administrator's Guide	2
Configuring Cleanup of Old Test Runs (page 92)	2
In-place Defined Metrics - distribution - Examples (page 113)	2
Referencing Page Variables and Parameters (Page 127)	3
Importing a Certificate to the Java Keystore - Step 3 (page 138)	3
Windows Example (page 138)	3
Linux Example (page 138)	4
Keytool Commands (page 138)	4
Quick Start: Ant Build - Step 2 (page 142)	4
Quick Start: Continuous Integration (page 143)	4
Build Descriptors (Definitions) - (Page 148)	5
Eclipse Projects (page 153)	6
Scheduled Builds (page 157)	6
Executing a Synchronization Pair - Example 1 (Page 166)	7
Running Scripts Via Jobs (page 175, 176)	7
Cleaning Up Temporary Folders (page 177)	7
Cleaning Up Old Builds (page 183)	7
Scheduler Configuration (page 183)	7
Configuring Polarion (page 189)	8
Installing svnserve as a Service (page 205)	8
User Guide	9
Job Configuration Syntax (page 315)	9
Using Queries in Macros (page 349)	9
Referencing Fields in Queries (page 349)	9
Basic Parameter Usage (page 351, 352)	9
Polarion Reference	11
History Indexing Job (page 6)	11
Header/Footer Properties Example (pages 23)	11
Header/Footer Properties Example (pages 24)	12
Building configuration file format (page 27)	13
Job Configuration Example (page 28)	14
Project Group Metrics (page 48, 49)	14
xUnit File Example (page 112)	14
Work Item and Index Fields (page 117)	15

Administrator's Guide

Configuring Cleanup of Old Test Runs (page 92)

```
<job cronExpression="0 0 6 ? * SUN" disabled="false" id="polarion.jobs.testruns.delete" name="Cleanup of  
Old Test Runs" scope="system">  
  <userAccountVaultKey>xUnitImporterUser</userAccountVaultKey>  
</job>
```

In-place Defined Metrics - distribution - Examples (page 113)

```
<entry>  
<key>DEFECTS-BY-RISK</key>  
  <label>Defects by Risk</label> <value type="distribution" enumeration="defect:enum:risk"  
    query="type:defect AND risk:$option$" reverseOrder="true"/>  
</entry>  
<entry>  
  <key>REQ-BY-RELEASE</key> <label>Requirements By Release</label> <value type="distribution"  
    enumeration="enum:release" query="type:requirement AND targetRelease:$option$"/>  
</entry>
```

And...

```
<entry>  
  <key>RESOLVED-RATIO</key>  
  <label>Resolved vs. Unresolved</label>  
  <description>Rate of Work Items that have resolution set.</description>  
  <value type="distribution">  
    <query id="resolved" label="Resolved">NOT resolution:##### NULL</query> <query id="not-  
resolved" label="Unresolved">resolution:##### NULL</query>  
  </value>  
</entry>
```

Referencing Page Variables and Parameters (Page 127)

```
<!-- Example referencing Velocity variable -->
#if($confidential)
    <td width="70%" align="right">${projectName} ${documentTitle} (rev. ${revision}) CONFIDENTIAL</td>
#else
    <td width="70%" align="right">${projectName} ${documentTitle} (rev. ${revision})</td>
#end

<!-- Example referencing page parameter -->
#if($pageParameters.internalDoc.equals(true))
    <td width="70%" align="right">${projectName} ${documentTitle} (rev. ${revision}) INTERNAL USE
ONLY</td>
#else
    <td width="70%" align="right">${projectName} ${documentTitle} (rev. ${revision})</td>
#end

<!-- Example referencing both -->
#if($confidential)
    <td width="70%" align="right">${projectName} ${documentTitle} (rev. ${revision}) CONFIDENTIAL</td>
#else
#if($pageParameters.internalDoc.equals(true))
    <td width="70%" align="right">
        ${projectName} ${documentTitle} (rev. ${revision}) INTERNAL USE ONLY
    </td>
#else
    <td width="70%" align="right">${projectName} ${documentTitle} (rev. ${revision})</td>
#end
#end
```

Importing a Certificate to the Java Keystore - Step 3 (page 138)

```
$JDK_HOME/bin/keytool -importcert -file $CERT -alias $ALIAS -keystore $JRE_HOME/lib/security/jssecacerts -storepass
changeit
```

Windows Example (page 138)

```
C:\Polarion\bundled\java\bin\keytool -importcert -file C:\Polarion\bundled\apache\conf\certificate.pem -
alias labs.polarion.com -keystore C:\Polarion\bundled\java\lib\security\jssecacerts -storepass changeit
```

Linux Example (page 138)

```
/usr/java/jdk1.7.0_79/bin/keytool -importcert -file /etc/pki/tls/certs/cert.pem -alias labs.polarion.com  
-keystore /usr/java/jdk1.7.0_79/jre/lib/security/jssecacerts -storepass changeit
```

Keytool Commands (page 138)

```
keytool -list -keystore %JAVA_HOME%\lib\security\jssecacerts -storepass changeit  
keytool -delete -alias mykey -keystore %JAVA_HOME%\lib\security\jssecacerts -storepass changeit  
keytool -importcert -help  
keytool -help
```

Quick Start: Ant Build - Step 2 (page 142)

```
<descriptors>  
  <descriptor>  
    <deploy>  
      <copy dir="dist" includes="*.jar" todir="results" label="jar files"/>  
    </deploy>  
  </descriptor>  
</descriptors>
```

Quick Start: Continuous Integration (page 143)

```
<job name="Integration Build of Example Project" id="master.build" cronExpression="0 0 0/1 ? * *"  
scope="project:ExampleProject">  
  <buildArtifact>com.example.project:normal-build</buildArtifact>  
  <buildDescriptorName>_default</buildDescriptorName>  
  <localDeploymentSpaceName>_default</localDeploymentSpaceName>  
</job>
```

Build Descriptors (Definitions) - (Page 148)

```
<?xml version="1.0" encoding="UTF-8"?>
<descriptors xmlns="http://polarion.com/schema/Builder/Descriptors"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://polarion.com/schema/Builder/Descriptors">
  <!-- no name and "_default" are equal -->
  <descriptor name="NAME">
    <!-- may be omitted -->
    <artifacts>
      <artifact>
        <!-- if id ends with "*" then the given prefix is searched for; default value is "*" -->
        <groupId>GROUP_ID</groupId>
        <!-- if id ends with "*" then the given prefix is searched for; default value is "*" -->
        <artifactId>ARTIFACT_ID</artifactId>
      </artifact>
      ... <artifact> ...
    </artifacts>

    <!-- may be omitted -->
    <build>
      <properties>
        <PROPERTY_NAME>PROPERTY_VALUE</PROPERTY_NAME>
        ...
      </properties>
    </build>

    <!-- may be omitted -->
    <deploy>
      <!-- includes and excludes may be omitted, default value for defaultexcludes is true -->
      <copy|move todir="TARGET_DIRECTORY" label="LABEL_FOR_BUILD_REPORT" dir="SOURCE_DIRECTORY"
includes="INCLUDES_PATTERN" excludes="EXCLUDES_PATTERN"
defaultexcludes="USE_DEFAULT_EXCLUDES_TRUE_OR_FALSE">
        ... <copy|move> ...
      <item todir="TARGET_DIRECTORY" label="LABEL_FOR_BUILD_REPORT" name="ITEM_NAME" />
      ... <item /> ...
    </deploy>
  </descriptor>

```

```

        </deploy>
    </descriptor>
    ... <descriptor> ...
</descriptors>

```

Eclipse Projects (page 153)

```

<artifact>
    <type>maven2</type>
    <location>trunk</location>
    <resources>
        <resource source="folderA/plugin1" target="plugins/plugin1" recurse="true"/>
        <resource source="folderA/plugin2" target="plugins/plugin2" recurse="true"/>
        <resource source="folderB/plugin3" target="plugins/plugin3" recurse="true"/>
        <resource source="folderA/feature1" target="features/feature1" recurse="true"/>
        <resource source="folderB/feature2" target="features/feature2" recurse="true"/>
        <resource source="pom.xml" target=""/>
    </resources>
</artifact>

```

Scheduled Builds (page 157)

```

<job name="ARBITRARY_NAME" id="master.build" cronExpression="CRON_EXPRESSION"
scope="project:PROJECT_NAME">
    <buildArtifact>GROUP_ID:ARTIFACT_ID</buildArtifact>
    <!-- use "_default" for default build -->
    <buildDescriptorName>BUILD_DESCRIPTOR_NAME</buildDescriptorName>
    <!-- use for integration build, omit for standalone build -->
    <localDeploymentSpaceName>_default</localDeploymentSpaceName>
    <!-- may be omitted; specifies additional build properties -->
    <properties>
        <PROPERTY_NAME>PROPERTY_VALUE</PROPERTY_NAME>
        ...
    </properties>
</job>

```


Executing a Synchronization Pair - Example 1 (Page 166)

```
<job id="synchronizer" name="Connector Sync Job" scope="project:MyProjectID" cronexpression="0019?*MON-FRI">
  <syncPair>[SYNC PAIR ID]</syncPair>
</job>
```

Running Scripts Via Jobs (page 175, 176)

```
<job name="Script Job" id="script.job" cronExpression="" disabled="true" scope="system">
  <scriptName>test.js</scriptName>
  <scriptEngine>js</scriptEngine>
  <properties>
    <myProperty>myPropertyValue</myProperty>
  </properties>
</job>
```

Cleaning Up Temporary Folders (page 177)

```
<job cronExpression="0 55 * ? * *" id="jobs.cleanup" name="Cleanup of Temporary Files" scope="system">
  <preserveAgeHours>36
</preserveAgeHours>
</job>
```

Cleaning Up Old Builds (page 183)

```
<job name="Builds Cleanup" id="build.clean" cronExpression="0 0 1 ? * MON-SAT" scope="system">
  <localDeploymentSpaceName>_default</localDeploymentSpaceName>
</job>
```

Scheduler Configuration (page 183)

```
<job name="_ARBITRARY_JOB_NAME_" id="logs.cleanup" cronExpression="" scope="system"
disabled="true">
  <maxFileAgeDays>_MAX_AGE_OF_FILE_IN_DAYS_</maxFileAgeDays> <filePatterns>
    <filePattern>_FOLDER_|_PATTERN_</filePattern>
    ... <filePattern> ...
  </filePatterns>
</job>
```

Configuring Polarion (page 189)

```
com.polarion.portal.tomcat.usesProxySSO=true  
com.polarion.portal.tomcat.RequestHeadersCredentialsProvider.userNameHeader=userheader  
com.polarion.portal.tomcat.RequestHeadersCredentialsProvider.passwordHeader=pwheader
```

And...

```
login.proxySSOLoginFailed=Message for end users in case authentication fails or is misconfigured.
```

Installing svnserve as a Service (page 205)

“WINDOWS” section. The following is a single command, which should be entered on a single command line:

```
sc create SvnServe2Polarion binpath= "\"C:\Polarion\bundled\svn\bin\svnserve.exe\" --service -R  
-r C:\Polarion\data\svn\repo --listen-host 127.0.0.1" displayname= "SvnServe2Polarion" depend=  
Tcpip start= auto
```

User Guide

Job Configuration Syntax (page 315)

```
<job cronExpression="0 * * * * ?" disabled="false" id="xUnitFileImport" name="xUnitFileImport"
scope="system">
  <path>c:\Temp\xunit</path>
  <project>playground</project>
  <idPrefix>TR</idPrefix>
  <userAccountVaultKey>xUnitFileImportUser</userAccountVaultKey>
</job>
```

Using Queries in Macros (page 349)

```
{workitems: query=NOT HAS_VALUE:resolution AND type:defect AND (severity:blocker OR severity:critical)}
```

Referencing Fields in Queries (page 349)

```
{workitems: query=resolution:fixed AND type:defect | fields=id, title, severity}
```

And...

```
{workitems: query=resolution:fixed AND type:defect | fields=id, title, severity | sortBy=severity}
```

Basic Parameter Usage (page 351, 352)

```
Updated from {parameter-editor:dateFrom} to {parameter-editor:dateTo}
```

...

THE FOLLOWING IS A SINGLE LINE:

```
{workitems:query=$pageParameters.query $!queryOpen AND
assignee.id:($pageParameters.getAsString($pageParameters.user, " OR ")) AND
updated:[$pageParameters.getDateForQuery("dateFrom") TO $pageParameters.getDateForQuery("dateTo")]}
```

##Parameter Definitions

```

{parameter:team|type=string|value=teamX|allowed-values=teamX,teamY}
{parameter:query|type=string|value=type: defect} {parameter:justOpen|type=boolean|value=true}
{parameter:teamX|type=user|value=mTest,aSeller|multi=true}
{parameter:teamY|type=user|value=sDeveloper,pGUI|multi=true}
{parameter:dateFrom|type=date|value=2011-05-05} {parameter:dateTo|type=date|value=@current}

```

1 Work Items By Team

```

{parameter-form}
Select Team: {parameter-editor:team} \
Query: {parameter-editor:query} {parameter-form-submit} \
Not Resolved: {parameter-editor:justOpen} \
Updated from {parameter-editor:dateFrom} to {parameter-editor:dateTo} \
{parameter-form}

```

Team members:

```

$pageParameters.render($pageParameters.team," ", " )

```

```

#if ($pageParameters.justOpen)
#set($queryOpen = " AND NOT HAS_VALUE:resolution")
#end

```

THE FOLLOWING IS A SINGLE LINE:

```

{workitems:query=$pageParameters.query $!queryOpen AND
assignee.id:($pageParameters.getAsString($pageParameters.team, " OR ")) AND
updated:[$pageParameters.getDateForQuery("dateFrom") TO $pageParameters.getDateForQuery("dateTo")]}

```

Polarion Reference

History Indexing Job (page 6)

```
<job cronExpression="0 0 1 ? * *" id="polarion.jobs.db.history" name="DB History Creator"
scope="system" />
```

And...

```
<job cronExpression="0 0 6 ? * *" id="polarion.jobs.db.history.killer" name="DB History Creator
Killer" scope="system" />
```

Header/Footer Properties Example (pages 23)

```
<export>
  <pdf-compare marginLeft="10" marginTop="10" marginRight="10" marginBottom="10">
    <header height="35">
      <![CDATA[
        <table width="100%" style="color:rgb(128,128,128)">
          <tr>
            <td width="5%"></td>
            <td width="35%">Polarion Software<br><a style="color:rgb(128,128,128)"
href="http://www.polarion.com">http://www.polarion.com</a></td>
            <td width="70%" align="right">${projectName} ${documentTitle} (compare rev.
${revision1} and ${revision2})</td>
          </tr>
        </table>
      ]]>
    </header>
    <footer height="25">
      <![CDATA[
        <table width="100%" style="border-top:1px solid rgb(128,128,128);color:rgb(128,128,128)">
          <tr>
```

```

        <td width="20%"><span style="color:rgb(0,0,0)">${page} |</span> Page</td>
        <td width="60%" align="center"><a style="color:rgb(128,128,128) "
href="http://www.polarion.com">${productName} ${productVersion}</a></td>
        <td width="20%" align="right">${generated}</td>
    </tr>
</table>
]]>
</footer>
</pdf-compare>
</export>

```

Header/Footer Properties Example (pages 24)

```

<export>
<pdf-compare-documents marginLeft="10" marginTop="10" marginRight="10" marginBottom="10">
<header height="35"><![CDATA[
    <table width="100%" style="color:rgb(128,128,128); font-size: 8px !important; margin-bottom: 20px;">
    <tr>
        <td width="5%"></td>
        <td width="35%">Polarion Software<br><a style="color:rgb(128,128,128); font-size: 8px !important;"
href="http://www.polarion.com">http://www.polarion.com</a></td>
        <td width="70%" align="right" style="font-size: 9px !important;">${projectName1} ${documentName1}
rev. ${revision1}
        compare ${projectName2} ${documentName2} rev. ${revision2}</td>
    </tr>
    </table>]]>
</header>
<footer height="25"><![CDATA[
    <table width="100%" style="border-top:1px solid rgb(128,128,128);color:rgb(128,128,128);font-size:
8px !important;">
    <tr>
        <td width="20%" style="font-size: 9px !important;"><span style="color:rgb(0,0,0)">${page} |</span>
Page</td>
        <td width="60%" align="center"><a style="color:rgb(128,128,128) "

```

```

href="http://www.polarion.com">${productName} ${productVersion}</a></td>
    <td width="20%" align="right" style=";font-size: 9px !important; ">${generated}</td>
</tr>
</table>]]>
</footer>
</pdf-compare-documents>
</export>

```

Building configuration file format (page 27)

```

<?xml version="1.0" encoding="UTF-8"?>
<artifacts xmlns="http://polarion.com/schema/Builder/Artifacts"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://polarion.com/schema/Builder/Artifacts"
    <!-- default value is "false" (the other is "true") -->
    auto-recognition="AUTO_RECOGNITION_FLAG">
  <artifact>
    <type>TYPE </type>
    <!-- location may be omitted in which case it is the project's location -->
    <location>BASE_LOCATION_RELATIVE_TO_PROJECT'S_REPOSITORY_LOCATION </location>
    <!-- resources may be omitted in which case the whole base location is used as a resource -->
    <resources>
      <!-- default value of recurse is "true" (other value is "false") -->
      <resource source="REPOSITORY_LOCATION_RELATIVE_TO_BASE_LOCATION"
target="TARGET_LOCATION_IN_LOCAL_DEPLOYMENT_SPACE"
        recurse="WHETHER_TO_USE_FOLDER_CONTENTS_AS_A_RESOURCE"/>
      ... <resource> ...
    </resources>
    ... additional type-specific elements ...
  </artifact>
  <artifact>
    ...
  </artifact>
</artifacts>

```

Job Configuration Example (page 28)

```
<job name="Work Report" id="tracker.workreport" cronExpression="0 0 0 ? * *" scope="system">
  <fileName>scheduled_work_report_{yyyy-MM-dd}.xls</fileName>
  <overwrite>true</overwrite>
  <title>Work Report</title>
  <scale>week</scale>
  <grouping>project,user</grouping>
  <query></query>
  <periodStart></periodStart>
  <periodFinish></periodFinish>
</job>
```

Project Group Metrics (page 48, 49)

```
{line-chart:report-path=charts/workitems-trend.xml|items=workitems-trend-data:RESOLVED-
UNRESOLVED/*|tags=14|width=100%|height=100%}
```

And...

```
{line-chart:report-path=charts/workitems-trend.xml|items=workitems-trend-data:#RESOLVED-
UNRESOLVED.UNRESOLVED|tags=14|width=100%|height=100%}
```

xUnit File Example (page 112)

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- This example shows only tags, attributes and content which is currently read by Polarion
      See http://windyroad.org/dl/Open%20Source/JUnit.xsd -->

<testsuites>
  <!--Name of the root tag does not matter, but it must not be same as the ones below -->

  <testsuite timestamp="2011-10-17T23:05:06">
    <!-- testsuite tags can be nested, timestamp is not required and format is "yyyy-MM-dd'T'HH:mm:ssZ" -->
    <testsuite>
      <testcase name="someMethod" classname="SomeClass" time="0.285"/>
      <!-- test case id is "SomeClass.someMethod", time is not required and is in seconds -->
    </testsuite>
  </testsuite>
</testsuites>
```



```

<testcase name="otherMethod" classname="SomeClass" time="0.001">
  <failure message="failure message" type="package.Exception">
    <!-- message and type are not required, all text content is added to the created defect -->
    Failure details
  </failure>
</testcase>

<testcase name="otherMethod" classname="OtherClass" time="0.002">
  <error message="error message" type="package.OtherException">
    Error Details
  </error>
</testcase>
</testsuite>
</testsuites>

```

Work Item and Index Fields (page 117)

```

{workitems: timePoint.time:[20120101 TO 20120212] AND assignee.id:$me | fields=id, title, severity,
assignee, timePoint | sortby=timePoint created}

```