

## **JSR 362 TCK Assertions**

---

**Version 0.1**

Send comments about this document to: [issues@portletspec3.java.net](mailto:issues@portletspec3.java.net)

This document is the portion of the JSR 362 Portlet Specification 3.0 that describes the assertions covered by the JSR 362 Technology Compatibility Kit.

Martin Scott Nicklous (Scott.Nicklous@de.ibm.com)  
IBM Corporation

IBM Corporation (the “Spec Lead”) for the Portlet (JSR 362) Specification 3.0 (the “Specification”) hereby grants permission to copy and display the Specification, in any medium without fee or royalty, provided that you include the following on ALL copies, or portions thereof, that you make:

1. A link or URL to the Specification at this location: <http://jcp.org/en/jsr/detail?id=362>.
2. The copyright notice as shown herein.

The Spec Lead offers to grant a perpetual, non-exclusive, worldwide, fully paid-up, royalty free, irrevocable license under its licensable copyrights and patent claims for which there is no technically feasible way of avoiding infringement in the course of implementing the Specification, provided that you:

- (a) fully implement the Specification including all its required interfaces and functionality.
- (b) do not modify, subset, superset or otherwise extend the Specification's name space;
- (c) pass the TCK for this Specification; and
- (d) grant a reciprocal license to any of your patents necessary to implement required portions of the Specification on terms consistent with the provisions of Section 6.A of the Java Specification Participation Agreement.

THE SPECIFICATION IS PROVIDED "AS IS," AND THE SPEC LEAD AND ANY OTHER AUTHORS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT, OR TITLE; THAT THE CONTENTS OF THE SPECIFICATION ARE SUITABLE FOR ANY PURPOSE; NOR THAT THE IMPLEMENTATION OF SUCH CONTENTS WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS. THE SPEC LEAD AND ANY OTHER AUTHORS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SPECIFICATION OR THE PERFORMANCE OR IMPLEMENTATION OF THE CONTENTS THEREOF.

The name and trademarks of the Spec Lead or any other Authors may NOT be used in any manner, including advertising or publicity pertaining to the Specification or its contents without specific, written prior permission. Title to copyright in the Specification will at all times remain with the Authors.

No other rights are granted by implication, estoppel or otherwise.

# **1 Change History**

- 20140626 – Initial version to test the document format.

## **2 Version 3.0 Assertions**

Tbd

### 3 JSR 286 Portlet Specification 2.0 Assertions

The assertions in this section serve to demonstrate backward compatibility with the JSR 286 Portlet Specification 2.0. The first part of this section documents assertions described in the JSR 286 specification itself. The second part documents assertions gleaned from the JSR 286 API documentation. There may be overlap between the two sections.

#### 3.1 JSR 286 Specification Assertions

Section	Title	Name	Test	Description
PLT.2.5	Compatibility	deployJSR168-1	TRUE	A portlet with the JSR 168 deployment descriptor can be deployed
PLT.2.5	Compatibility	deployJSR168-2	TRUE	A portlet implementing only the JSR 168 API can be deployed
PLT.5.1	Number of Portlet Instances	instantiate1	TRUE	In a single-VM environment, the portlet container must instantiate only one portlet object per portlet definition
PLT.5.1	Number of Portlet Instances	instantiate2	TRUE	When the portlet application is marked as distributable in a multiple-VM environment, the portlet container must instantiate only one portlet object per portlet definition per virtual machine

#### 3.2 JSR 286 Apidoc Assertions

Class	Method	Name	Test	Description
ActionFilter		extendsPortletFilter	TRUE	ActionFilter extends PortletFilter
ActionFilter		canBeConfigured1	TRUE	An ActionFilter can be configured in the portlet descriptor
ActionFilter		canBeConfigured2	TRUE	Multiple ActionFilter classes can be configured in the portlet descriptor
ActionFilter	doFilter	hasDoFilter	TRUE	ActionFilter implements a doFilter method with the appropriate signature
ActionFilter	doFilter	doFilterIsCalled	TRUE	The doFilter method is called before the processAction method for the portlet

<b>Class</b>	<b>Method</b>	<b>Name</b>	<b>Test</b>	<b>Description</b>
ActionFilter	doFilter	doFilterProcessAction 1	TRUE	After the method has successfully completed and invokes the next filter, the processActionMethod is called
ActionFilter	doFilter	doFilterProcessAction 2	TRUE	After the method has successfully completed and invokes the next filter, the next filter in the chain is called if multiple filters are defined
ActionFilter	doFilter	doFilterBlock	TRUE	If the method does not invoke the next filter, processAction is not called
ActionFilter	doFilter	doFilterException1	TRUE	If the method throws an UnavailableException, processAction is not called
ActionFilter	doFilter	doFilterException2	TRUE	If the method throws an UnavailableException, no further filter is called
ActionFilter	doFilter	doFilterExamine	TRUE	After the next filter has been successfully invoked, the ActionResponse may be examined