

PLT.17 User Information Configuration

by Ate Douma

Table of contents

1	2
2	2
3	4

1.

The Portlet Specification defines how Portlet Applications can use User Attributes. The attributes must be defined in the portlet.xml like (see PLT.17.1):

```
<portlet-app version="1.0"
xmlns="http://java.sun.com/xml/ns/portlet/portlet-app_1_0.xsd">
  <user-attribute>
    <description>User Given Name</description>
    <name>user.name.given</name>
  </user-attribute>
  <user-attribute>
    <description>User Last Name</description>
    <name>user.name.family</name>
  </user-attribute>
  <user-attribute>
    <description>User eMail</description>
    <name>user.home-info.online.email</name>
  </user-attribute>
  ...
</portlet-app>
```

Once attributes are defined like this, a portlet can access the current values for the logged on user as an unmodifiable Map from the PortletRequest using the USER_INFO constant defined in the PortletRequest interface (see PLT.17.2):

```
Map userInfo = (Map)request.getAttribute(PortletRequest.USER_INFO);
String givenName = (userInfo!=null) ?
(String)userInfo.get("user.name.given") : "";
String lastName = (userInfo!=null) ?
(String)userInfo.get("user.name.family") : "";
String email = (userInfo!=null) ?
(String)userInfo.get("user.home-info.online.email") : "";
```

What is not defined by the Portlet Specification is *how* the Portal must map the defined User Attributes to concrete attributes of a user.

2.

Jetspeed-2 provides a very flexible way to define concrete User attributes and defining access to them.

Concrete User attributes are stored using User Preferences for which Jetspeed-2 provides its own database back end for storage (which is customizable by the way like almost any component of Jetspeed-2)

The concrete User attributes are stored under a specific node in the User preferences and can contain any named attribute at will.

These concrete User attributes can be mapped to the defined User Attributes in the portlet.xml in two ways:

1. Using an exact match of attribute names
2. Using a custom mapping definition in a jetspeed-portlet.xml

If you write new Portlet Applications with Jetspeed-2 as target Portal, defining User Attributes which match the concrete User attributes in the Portal usually will be quite straightforward

But, if you have an existing Portlet Application which you want to deploy on Jetspeed-2, there might be a mismatch between the attribute names needed by the Portlet Application and the concrete attribute names as stored in the Jetspeed-2 User Preferences.

Note: The Portlet Specification defines a set of attribute names which are recommended to be used in Appendix PLT.D.

Portlet Applications using these attribute names and Portals storing the concrete User attributes also using these names won't need any custom attribute mapping as will be described below.

Although Jetspeed-2 allows a fully free definition of the concrete User attributes, it is recommended to use these predefined attributes names as much as possible.

The jetspeed-portlet.xml allows jetspeed specific configurations and customizations to be specified.

This deployment document isn't required, but will be processed if found within the WEB-INF folder of a Portlet Application war.

Jetspeed specific configurations must be defined using the "http://portals.apache.org/jetspeed" namespace.

User attribute mapping is defined using an "user-attribute-ref" element containing a "name" element defining the custom user attribute name and a "name-link" element defining the concrete attribute name to which it is mapped:

```
<portlet-app version="1.0"
xmlns="http://java.sun.com/xml/ns/portlet/portlet-app_1_0.xsd"
xmlns:js="http://portals.apache.org/jetspeed">
  <js:user-attribute-ref>
    <js:name>user-name-given</js:name>
    <js:name-link>user.name.given</js:name-link>
  </js:user-attribute>
  <js:user-attribute-ref>
    <js:name>user-name-family</js:name>
    <js:name-link>user.name.family</js:name-link>
  </js:user-attribute>
  ...
</portlet-app>
```

Using the above custom mapping as an example, the Portlet can now access the user

attributes as follows:

```
Map userInfo =
(Map)request.getAttribute(PortletRequest.USER_INFO);
String givenName = (userInfo!=null) ?
(String)userInfo.get("user-name-given") : "";
String lastName = (userInfo!=null) ?
(String)userInfo.get("user-name-family") : "";
String email = (userInfo!=null) ?
(String)userInfo.get("user.home-info.online.email") : "";
```

Note that the email attribute for which no custom mapping was defined still can be access using exact name matching (provided the concrete attribute is defined for the logged on user).

3.

Jetspeed-2 is provided with several Administrative Portlets, including for User Management. Using the User Management Portlets, it is very easy to define or modify concrete attributes for a user:

The User Info Test demo Portlet, default deployed in Jetspeed-2 and displayed on the start page, uses the above example User Attribute definitions and displays the values for the logged on user (also showing that these can be accessed from both the PortletRequest as well as the HttpServletRequest from within a servlet):