



Jetspeed-2 Preferences SPI Components v.2.1.3

Project Documentation

Table of Contents

1 Jetspeed-2 Preferences Documentation	
1.1 Preferences Overview	1

1.1 Preferences Overview

Preferences Overview

The `java.util.prefs.Preferences` API provides a generic mechanism for storing user and system preferences, as well as configuration data. Jetspeed 2 relies on this API to provide a wide ranging of functionality to higher level services. Some of the Jetspeed components leveraging the `java.util.prefs.Preferences` API are:

- **Portlet Preferences:** Portlet preferences are stored in the system preferences tree. The following path structure is used to establish the location of a given portlet preference in the preferences tree: `/portlet_application/${Portlet Application Name}/portlets/${Portlet Name}/preferences/${Preference Name}/values`. The preferences values are stored under the `values` node as a map of key/values.
- **User Attributes:** User attributes are stored in the user preferences tree as follow. The following path structure is used to stored a user's attributes: `/user/${User Name}/userinfo`. The user attributes are stored as a preferences key/values map under the `userinfo` node.
- **Roles and groups hierarchy:** Documentation on how roles and groups hierarchy use the Preferences model can be found in the [security section](#) of the documentation.

Preferences Goals

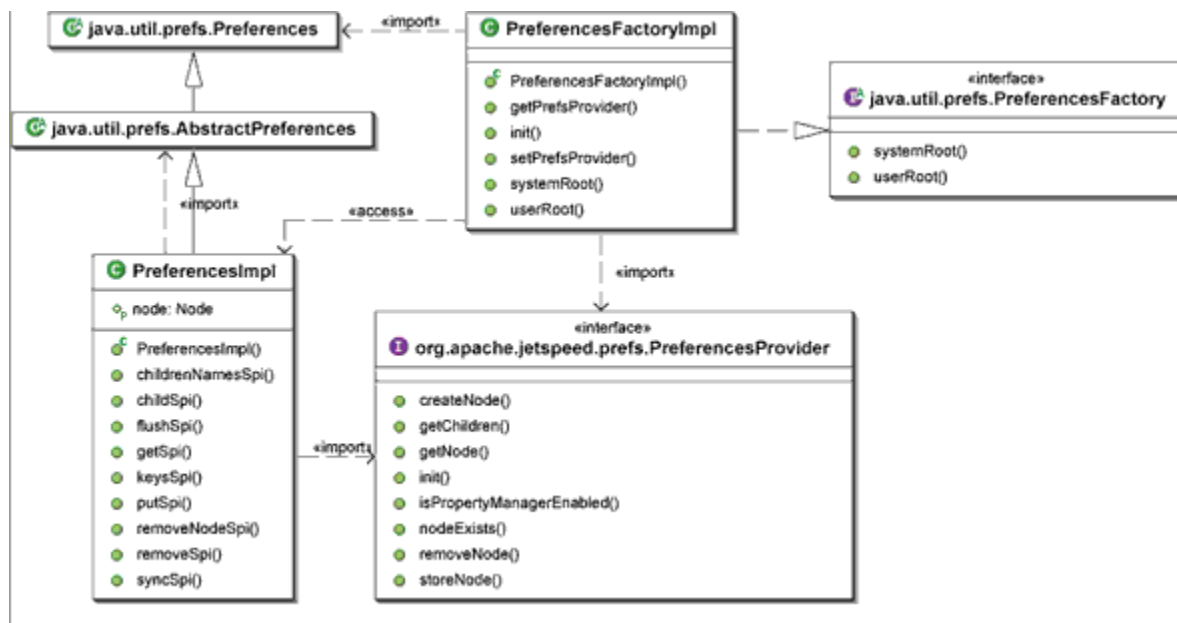
The *Preferences* API provides a generic implementation for organizing properties as illustrated below:

```
<preferences EXTERNAL_XML_VERSION="1.0">
<root type="user">
  <map />
  <node name="user">
    <map />
    <node name="principal1">
      <map />
    <node name="propertyset1">
      <map>
        <entry key="set1prop1" value="256" />
        <entry key="set1prop2" value="314" />
      </map>
    </node>
  </node>
</root>
```

A good reference article can be found at [ONJava.com](#).

Preferences Implementation Overview

Jetspeed 2 Preferences implementation leverages the following classes:



Class	Description
PreferencesFactoryImpl	Implements the <code>java.util.prefs.PreferencesFactory</code> that generates <code>Preferences</code> objects. The <code>PreferencesFactoryImpl</code> is configured through spring assembly (<code>prefs.xml</code>) to use a <code>PreferencesProvider</code> that encapsulates the backing store implementation.
PreferencesImpl	Extends the <code>java.util.prefs.AbstractPreferences</code> and implements the SPI by leveraging the <code>PreferencesProvider</code> API. Jetspeed <code>Preferences</code> implementation can be configured to validate that preferences map key/values are created for predefined keys. A <code>PropertyManager</code> enforces the checks if enabled.
PreferencesProvider	Encapsulates the preferences implementation against a specific backing store. In Jetspeed 2, the default implementation is provided by <code>org.apache.jetspeed.prefs.impl.PersistenceBrokerPreferencesProvider</code>

Preferences Database Schema

The following database schema is being used in Jetspeed 2 to store `Preferences` in a relational database model.

