

This project has retired. For details please refer to its [Attic page](#).

Status of this document

This document is very much a "Work in Progress". Although some sections are incomplete, I hope what is here is helpful. Comments and suggestions are welcome and encouraged. They can be set to jetspeed-user@portals.apache.org

General Configuration steps

The following steps are purely guide. Some steps are not required.

- [Install Jetspeed](#)
- [Override any properties in default property files](#)
- [Define portlets in local-portlets](#)
- [Define skins in local-skins.xreg](#)
- [Define local controls and controllers in local-managers.xreg](#)
- [Define content feeds](#)
- [Configure the layout and navigation bars](#)
- [Configure default home page](#)
- [Define users](#)
- [Configure the admin and turbine users](#)
- [Define default portal for new users](#)
- [Set portlet caching and other refresh rates](#)
- [Set Jetspeed logging](#)
- [Enable e-mail confirmation](#)
- [Logon Configuration](#)
- [Account Disabling](#)
- [Automatic Password Expiration](#)
- [Backup and Test](#)

Override any properties in default property files

Any site specific property settings overriding those `jr.props`, `tr.props`, `Torque.props`, may be placed in a custom property file (see `/WEB-INF/conf/my.properties` for an example). Using a custom property file makes future Jetspeed upgrades easier because the default property files remain unchanged.

See the [Properties Override How-To](#) for the steps required to activate such custom property file.

Define portlets in local-portlets

Any site specific portlets should be define in `<jetspeed_home>/WEB-INF/conf/local-portlets.xreg`. because this file will not be overwritten by upgrades.

See the [Portlet Catalog](#) for portlet configuration documentation.

Define skins in local-skins.xreg

Any site specific skins should be define in `<jetspeed_home>/WEB-INF/conf/local-skins.xreg`. because this file will not be overwritten by upgrades.

See the [Skin configuration guide](#) for more information.

Define local controls and controllers in local-managers.xreg

Any site specific controls and controllers should be define in `<jetspeed_home>/WEB-INF/conf/local-managers.xreg`. because this file will not be overwritten by upgrades.

Define content feeds

This section is incomplete

Configure the layout and navigation bars

This section is incomplete

The look of Jetspeed's portal is controlled by the layout and navigation templates. Below are some of the file that define that look.

A Layout Manager is used in the generation of the resulting portal. Jetspeed support 2 Layout Manager, JSP and Velocity. Both layout manager produce the similar results, which one you use is dictated by which language you prefer.

The parameter `services.TemplateService.default.extension` in `TurbineResource.properties` defines the Layout manager used.

Value	Description
jsp	The JSP layout manager will be used. This is the 1.3a1 default
vm	The vm layout manager will be used. This is the 1.3a2, 1.4b1, 1.4b2 default

Files uses by the Velocity Layout Manager

<jetspeed_home>/WEB-INF/template/vm/layouts/html/language/country/default.vm
Display the logo in the upper left hand corner and the define the top, bottom, and left navigation bars.

<jetspeed_home>/WEB-INF/template/vm/navigations/html/language/country/top.vm
Top navigation bar when the user it NOT logged in.

<jetspeed_home>/WEB-INF/template/vm/navigations/html/language/country/left.vm
Left navigation bar for all users.

<jetspeed_home>/WEB-INF/template/vm/navigations/html/language/country/bottom.vm
Bottom navigation bar for all users.

Files uses by the JSP Layout Manager

<jetspeed_home>/WEB-INF/template/jsp/layouts/html/language/country/default.jsp
Display the logo in the upper left hand corner and the define the top, bottom, and left navigation bars.

<jetspeed_home>/WEB-INF/template/jsp/navigations/html/language/country/top_default.jsp
Top navigation bar when the user is NOT logged in.

<jetspeed_home>/WEB-INF/template/jsp/navigations/html/language/country/top_loggedin.jsp
Top navigation bar for logged in users.

<jetspeed_home>/WEB-INF/template/jsp/navigations/html/language/country/left.jsp
Left navigation bar for all users.

<jetspeed_home>/WEB-INF/template/jsp/navigations/html/language/country/bottom/jsp
Bottom navigation bar for all users.

To customize the navigation bars, the following steps are suggested:

- Replace the reference to the Jetspeed logo with your logo in `default.xx`
- Add "banner adds" to `top.vm` or `top_default.jsp` and `top_loggedin.jsp`
- If no left navigation bar is desired, then remove the reference from `default.xx`
- If a left navigation bar is desired, then update `left.xx`
- Add any copyright notices, disclaimers, logos, ... to `bottom.xx`

Configure default home page

This section is incomplete

The default home page is stored in the <jetspeed_home>/WEB-INF/psml/anon directory. Currently this is not directly configurable with the customizer. A suggested procedure is to create a *HomePageUser*, do the desired configuration as that user, then copy the resulting files to the default home page.

Define users

This section is incomplete

Configure the admin and turbine users

This section is incomplete

Change the passwords!

Define default portal for new users

A new users home page is copied from the user TURBINE. Thus you should customized the TURBINE's user home page to reflect the desired default home page for new users.

Set portlet caching and other refresh rates

This section is incomplete

Set Jetspeed logging

As og 1.4b5, Jetspeed Log4J for logging. The logfile(s) are a valuable tool in trouble shooting problems and monitoring the general health of Jetspeed an it's portlets.

The base configuration of the logging is set in the settings in JetspeedResources.properties shown below. The details of the configuration is done in the file specified by log4j.properties. The way Jetspeed logs, you are able to controlled logging very detailed, if you should need so. Jetspeed defines a named logger for every single class that performs logging. Thus, you can configure it so that a single class logs debug to its own file. This level of detail gives you better control when debugging! The configuration shipped with Jetspeed is set up on a much coarser level initially.

```
#####
# JetspeedLogFactory Service          #
#####
# If this file has extension properties, then log4j property configuration is done.
# Otherwise DOM configuration for XML. Thus you can choose what format you like to use.
log4j.properties=/WEB-INF/conf/log4j.properties
# If true, Log4J will monitor the property file and reconfigure if changed.
log4j.configureAndWatch=true
# Number of ms between each property file check
log4j.watchInterval=60000
```

The logging level should be set according the the current needs. DEBUG logging is suggested during portlet testing, while INFO logging is appropriate for every-day activity in a stable environment. In heavy production systems, WARN is appropriate.

Detailed information on Log4J canbe found at <http://logging.apache.org/log4j/> .

Portlet usage logger

This facility is useful to generate web server like access log which captures access to individual portlets (since this cannot be accomplished via regular web server logging). See javadoc for org.apache.jetspeed.services.JetspeedPortletStatsService for more information about the internals of this service. The following are service properties as defined in JetspeedResources.properties:

Property	Description
services.PortletStats.classname = org.apache.jetspeed.services.portletstats.JetspeedPortletStatsService	Simple implementation using Apache Common Log Format (CLF)
services.PortletStats.enabled = false	This service is disabled by default
services.PortletStats.excludePortlets =	Portlets to exclude from logging (not yet implemented)
services.PortletStats.dateFormat = dd/MMM/yyyy:hh:mm:ss z	Date format to use in the log entry

NOTE: In addition, the category/logger named "org.apache.jetspeed.services.portletstats.JetspeedPortletStatsService" in the Log4J configuration controls where the actual logging goes. This must be set to log at level INFO for the portlet usage logger to be active.

Example configuration

The following configuration will log all general Jetspeed log to one file, all Jetspeed services to one, and all logging from turbine to a third. In addition, the necessary logger for the portlet usage is defined.

NOTE: If you remove the "{1}" behind the %c in the patterns in the configuration, you will get the fully qualified class name (i.e. including package name) in the log file. This might be useful.

```
# -----
#
# This file contains log4j-specific logging properties. This file is
# loaded by the logging service based on the following property
# in JetspeedResources.properties:
#
# log4j.properties = ${webappRoot}/WEB-INF/conf/log4j.properties
#
# Appender specified in log4j.category.default must be the same as
# the one specified in services.LoggingService.default property.
#
# All log4j properties should be supported. Check log4j documentation
# for more information.
#
# Note that strings containing "," (comma) characters must backslash
# escape the comma (i.e. '\,')
#
# -----
# The rootlogger means that all logging not defined otherwise, will go where it describes
log4j.rootLogger = INFO, jetspeed
# If debug = true, you will see how Log4J configures on the stdout of your webapp container
log4j.debug = true
#
# Jetspeed goes into Jetspeed Log
#
log4j.category.org.apache.jetspeed = DEBUG, jetspeed
log4j.additivity.org.apache.jetspeed = false

#
# Jetspeed services goes into Jetspeed Services Log
#
log4j.category.org.apache.jetspeed.services = DEBUG, jetspeedsrv
log4j.additivity.org.apache.jetspeed.services = false
# Setting CastorPsmManagerService to higher level, as it is quite verbose on DEBUG
log4j.category.org.apache.jetspeed.services.psmmanager.CastorPsmManagerService = INFO, jetspeedsrv
log4j.additivity.org.apache.jetspeed.services.psmmanager.CastorPsmManagerService = false

#
# Turbine goes into Turbine Log
# JetspeedLoggingService handles all logging thru TurbineLoggingService
#
log4j.category.org.apache.jetspeed.services.logging.JetspeedLoggingService = INFO, turbine
log4j.additivity.org.apache.jetspeed.services.logging.JetspeedLoggingService = false

#
# Portlet access Category
#
log4j.category.org.apache.jetspeed.services.portletstats.JetspeedPortletStatsService = INFO, access
log4j.additivity.org.apache.jetspeed.services.portletstats.JetspeedPortletStatsService = false

#
# Console output Category
#
log4j.category.stdout = INFO, stdout
```

```

log4j.additivity.stdout = false

#####
#
# Logfile definitions
#
#####

#
# jetspeed.log
#
log4j.appender.jetspeed = org.apache.log4j.FileAppender
log4j.appender.jetspeed.file = ${webappRoot}/WEB-INF/log/jetspeed.log
log4j.appender.jetspeed.layout = org.apache.log4j.PatternLayout
log4j.appender.jetspeed.layout.conversionPattern = %d [%t] %-5p %c{1} - %m%n
log4j.appender.jetspeed.append = false

log4j.appender.jetspeedsrv = org.apache.log4j.FileAppender
log4j.appender.jetspeedsrv.file = ${webappRoot}/WEB-INF/log/jetspeedservices.log
log4j.appender.jetspeedsrv.layout = org.apache.log4j.PatternLayout
log4j.appender.jetspeedsrv.layout.conversionPattern = %d [%t] %-5p %c{1} - %m%n
log4j.appender.jetspeedsrv.append = false

#log4j.appender.jetspeed = org.apache.log4j.RollingFileAppender
#log4j.appender.jetspeed.file = ${webappRoot}/WEB-INF/log/rotation.log
#log4j.appender.jetspeed.MaxFileSize = 50KB
#log4j.appender.jetspeed.MaxBackupIndex = 5
#log4j.appender.jetspeed.layout = org.apache.log4j.PatternLayout
#log4j.appender.jetspeed.layout.ConversionPattern = [%d{dd MMM yyyy HH:mm:ss} %5p] - %m%n

#
# turbine.log
#
log4j.appender.turbine = org.apache.log4j.FileAppender
log4j.appender.turbine.file = ${webappRoot}/WEB-INF/log/turbine.log
log4j.appender.turbine.layout = org.apache.log4j.PatternLayout
log4j.appender.turbine.layout.conversionPattern = %d [%t] %-5p %c{1} - %m%n
log4j.appender.turbine.append = false

#
# Portlet access Output
#
log4j.appender.access = org.apache.log4j.FileAppender
log4j.appender.access.file = ${webappRoot}/WEB-INF/log/access.log
log4j.appender.access.layout = org.apache.log4j.PatternLayout
log4j.appender.access.layout.conversionPattern = %m%n
log4j.appender.access.append = true

#
# Console Output
#
log4j.appender.stdout = org.apache.log4j.ConsoleAppender
log4j.appender.stdout.layout = org.apache.log4j.PatternLayout
log4j.appender.stdout.layout.ConversionPattern = [%d{dd MMM yyyy HH:mm:ss} %5p] - %m%n

```

The following XML configuration shows a similar configuration in XML format. You can choose your favorite format.

```

<?xml version="1.0" encoding="UTF-8" ?>
<log4j:configuration debug="true" xmlns:log4j='http://logging.apache.org/log4j/'>

    <root>
        <level value="info" />
        <appender-ref ref="jetspeed" />
    </root>

    <logger name="org.apache.jetspeed" additivity="false">
        <priority value="debug"/>
        <appender-ref ref="jetspeed"/>
    </logger>

    <logger name="org.apache.jetspeed.services" additivity="false">
        <priority value="debug"/>
        <appender-ref ref="jetspeedsrv"/>
    </logger>

    <logger name="org.apache.jetspeed.services.psmmanager.CastorPsmManagerService" additivity="false">
        <priority value="info"/>
    </logger>

```

```

        <appender-ref ref="jetspeedsrv"/>
    </logger>

    <logger name="org.apache.jetspeed.services.logging.JetspeedLoggingService" additivity="false">
        <priority value="info"/>
        <appender-ref ref="turbine"/>
    </logger>

    <logger name="org.apache.jetspeed.services.portletstats.JetspeedPortletStatsService." additivity="false">
        <priority value="info"/>
        <appender-ref ref="access"/>
    </logger>

    <appender name="jetspeed" class="org.apache.log4j.FileAppender">
        <param name="File" value="${webappRoot}/WEB-INF/log/jetspeedX.log" />
        <layout class="org.apache.log4j.PatternLayout">
            <param name="ConversionPattern"
                value="%d [%t] %-5p %c{1} - %m%n"/>
        </layout>
    </appender>

    <appender name="jetspeedsrv" class="org.apache.log4j.FileAppender">
        <param name="File" value="${webappRoot}/WEB-INF/log/jetspeedservicesX.log" />
        <layout class="org.apache.log4j.PatternLayout">
            <param name="ConversionPattern"
                value="%d [%t] %-5p %c{1} - %m%n"/>
        </layout>
    </appender>

    <appender name="turbine" class="org.apache.log4j.FileAppender">
        <param name="File" value="${webappRoot}/WEB-INF/log/turbineX.log" />
        <layout class="org.apache.log4j.PatternLayout">
            <param name="ConversionPattern"
                value="%d [%t] %-5p %c{1} - %m%n"/>
        </layout>
    </appender>

    <appender name="access" class="org.apache.log4j.FileAppender">
        <param name="File" value="${webappRoot}/WEB-INF/log/access.log" />
        <layout class="org.apache.log4j.PatternLayout">
            <param name="ConversionPattern"
                value="%d [%t] %-5p %c{1} - %m%n"/>
        </layout>
    </appender>

</log4j:configuration>

```

Enable e-mail confirmation

This section is incomplete

Logon Configuration

Case Insensitive Logon

Jetspeed can be configured to automatically force all usernames and passwords to upper or lower case. By setting the "caseinsensitive" property for the username or password to "true", then the "caseinsensitive.upper" property can be applied. When it is true, the username or password is converted to uppercase. When false, username or password are converted to lowercase. The default setting is false, where the username and password will be case-sensitive and the "caseinsensitive.upper" property is ignored.

```

services.JetspeedSecurity.caseinsensitive.username=false
services.JetspeedSecurity.caseinsensitive.password=false
services.JetspeedSecurity.caseinsensitive.upper=true

```

Logon Rules

When a new user is created, the PSML resources for the new user are copied from another user's account. This account is configurable with the property:

```
services.Profiler.newuser.template=turbine
```

Setting it to nothing configures role-based psml as the default method for new users.

```
services.Profiler.newuser.template=
```

Role Based PSML

Some configuration will want to share PSML resources among users. In this case, users will not have their own PSML resources. Enable Role Based PSML with this property set to true:

```
services.Profiler.rolefallback=true
```

Remember to set the "services.Profiler.newuser.template" property to nothing.

In other configurations, it may be useful to create initial user PSML based on the roles the user is assigned. The role profile merging feature aggregates PSML from each role to create a single tabbed profile. To enable this feature the following settings have to be set to true:

```
services.Profiler.rolefallback=true  
services.Profiler.rolemerge=true
```

Assigning Roles, Media Type

When a new user is created, you can configure which roles are automatically signed to that new user. You may also configure which media types that new user will support. Use commas to separate multiple entries.

```
services.JetspeedSecurity.newuser.roles=user  
services.Profiler.newuser.media_types=html,wml
```

Account Disabling

Accounts can be auto-disabled with a failed-logon tracking feature. To turn on this feature, set this property:

```
services.JetspeedSecurity.logon.auto.disable=true
```

Account Disabling is controlled by the parameters: count, interval, and max. These parameters are applied over the lifetime of the server. When the server is shutdown, all counts are reset. However, the 'disabled' status is persisted in the database. With the parameters below, the user is given 3 failed attempts over 300 seconds (5 minutes). After 5 minutes or a successful logon, the strike count is reset. After 3 failed attempts within 5 minutes, the account is disabled and the "logon.disabled.form" is displayed whenever the user attempts to logon again. The 'max' parameter counts all failed logon attempts over the entire lifetime of the server. After 10 failed attempts over the lifetime of the server, the account is disabled. Accounts can be enabled from the Jetspeed Admin User Maintenance screen.

```
# 3 logon strikes per 300 seconds and your out  
services.JetspeedSecurity.logon.strike.count=3  
services.JetspeedSecurity.logon.strike.interval=300  
# dont allow more than 10 over any time period  
services.JetspeedSecurity.logon.strike.max=10  
logon.disabled.form=AccountDisabled.vm
```

Automatic Password Expiration

Password can automatically expire with this feature. When password expires, the user is presented with a change password screen. The user must change the password in order to continue. When new user is created, a temporary password is assigned. On first login, the user is then prompted to change

the password. The administrator can force password expiration by setting column PASSWORD_CHANGED to null in TURBINE_USER for particular account(s). To turn on this feature, set this property to a value greater than 0:

```
services.JetspeedSecurity.password.expiration.period=[number of days before the password expires]
```

Backup and Test

This section is incomplete

Backup early and often!