

# Implementing WebDAV

## Copyright and Licensing Statement

All intellectual property rights in the SOFTWARE and associated user documentation, implementation documentation, and reference documentation are owned by Percussion Software or its suppliers and are protected by United States and Canadian copyright laws, other applicable copyright laws, and international treaty provisions. Percussion Software retains all rights, title, and interest not expressly granted. You may either (a) make one (1) copy of the SOFTWARE solely for backup or archival purposes or (b) transfer the SOFTWARE to a single hard disk provided you keep the original solely for backup or archival purposes. You must reproduce and include the copyright notice on any copy made. You may not copy the user documentation accompanying the SOFTWARE.

The information in Rhythmyx documentation is subject to change without notice and does not represent a commitment on the part of Percussion Software, Inc. This document describes proprietary trade secrets of Percussion Software, Inc. Licensees of this document must acknowledge the proprietary claims of Percussion Software, Inc., in advance of receiving this document or any software to which it refers, and must agree to hold the trade secrets in confidence for the sole use of Percussion Software, Inc.

The software contains proprietary information of Percussion Software; it is provided under a license agreement containing restrictions on use and disclosure and is also protected by copyright law. Reverse engineering of the software is prohibited.

Due to continued product development this information may change without notice. The information and intellectual property contained herein is confidential between Percussion Software and the client and remains the exclusive property of Percussion Software. If you find any problems in the documentation, please report them to us in writing. Percussion Software does not warrant that this document is error-free.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording or otherwise without the prior written permission of Percussion Software.

Copyright © 1999-2005 Percussion Software.  
All rights reserved

## Licenses and Source Code

Rhythmyx uses Mozilla's JavaScript C API. See <http://www.mozilla.org/source.html> (<http://www.mozilla.org/source.html>) for the source code. In addition, see the *Mozilla Public License* (<http://www.mozilla.org/source.html>).

Netscape Public License

Apache Software License

IBM Public License

Lesser GNU Public License

## Other Copyrights

The Rhythmyx installation application was developed using InstallShield, which is a licensed and copyrighted by InstallShield Software Corporation.

The Sprinta JDBC driver is licensed and copyrighted by I-NET Software Corporation.

The Sentry Spellingchecker Engine Software Development Kit is licensed and copyrighted by Wintertree Software.

The Java™ 2 Runtime Environment is licensed and copyrighted by Sun Microsystems, Inc.

The Oracle JDBC driver is licensed and copyrighted by Oracle Corporation.

The Sybase JDBC driver is licensed and copyrighted by Sybase, Inc.

The AS/400 driver is licensed and copyrighted by International Business Machines Corporation.

The Ephox EditLive! for Java DHTML editor is licensed and copyrighted by Ephox, Inc.

This product includes software developed by CDS Networks, Inc.

The software contains proprietary information of Percussion Software; it is provided under a license agreement containing restrictions on use and disclosure and is also protected by copyright law. Reverse engineering of the software is prohibited.

Due to continued product development this information may change without notice. The information and intellectual property contained herein is confidential between Percussion Software and the client and remains the exclusive property of Percussion Software. If you find any problems in the documentation, please report them to us in writing. Percussion Software does not warrant that this document is error-free.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording or otherwise without the prior written permission of Percussion Software.

AuthorIT™ is a trademark of Optical Systems Corporation Ltd.

Microsoft Word, Microsoft Office, Windows®, Window 95™, Window 98™, Windows NT® and MS-DOS™ are trademarks of the Microsoft Corporation.

This document was created using *AuthorIT™, Total Document Creation* (see AuthorIT Home - <http://www.author-it.com>).

Schema documentation was created using XMLSpy™.

**Percussion Software**

600 Unicorn Park Drive

Woburn, MA 01801 U.S.A

781.438.9900

Internet E-Mail: [technical\\_support@percussion.com](mailto:technical_support@percussion.com)

Website: <http://www.percussion.com>



---

# Contents

<b>Introduction</b>	<b>3</b>
Architecture of WebDAV Components in Rhythmyx .....	4
Prerequisites and Recommendations .....	5
WebDAV Web Applications .....	6
Supported WebDAV Clients and Application Servers .....	7
<b>Implementing WebDAV</b>	<b>9</b>
Options for Implementing WebDAV Servlets .....	10
Implementing WebDAV in Rhythmyx's Installation of Tomcat .....	11
Adding Additional WebDAV Servlets in Rhythmyx's Installation of Tomcat .....	15
Editing a WebDAV Configuration File .....	17
Simple Example Implementation of WebDAV .....	18
Deploying the Rhythmyx WebDAV Servlet in BEA WebLogic .....	28
Deploying the Rhythmyx WebDAV Servlet in IBM WebSphere .....	32
Deploying the Rhythmyx WebDAV Servlet in Rhythmyx Express Portal .....	36
Troubleshooting WebDAV .....	39
<b>Working with WebDAV</b>	<b>41</b>
Opening Web Folders in Windows Explorer .....	42
Opening Web Folders in Internet Explorer .....	43
Opening Web Folders in Mac OS X .....	45
Working with Rhythmyx Content in Web Folders .....	46
WebDAV and Rhythmyx Workflow .....	47
<b>Appendices</b>	<b>49</b>
Default Rhythmyx WebDAV Configuration File .....	51
WebDAV Methods .....	55
<b>Index</b>	<b>57</b>

---



# Introduction

WebDAV is an extension to HTTP that lets users work with files on a remote Web Server. When content contributors use Rhythmyx with WebDAV they can create, modify, and save Rhythmyx Content Items while working in third-party applications, such as DreamWeaver and PhotoShop. Using Rhythmyx with WebDAV streamlines content creation in third-party applications because users do not have to log in to Rhythmyx in addition to opening the third-party application to create and modify content. To perform other Rhythmyx actions, such as making a Workflow Transition or viewing the Audit Trail, users must still log in to Rhythmyx.

To use Rhythmyx with WebDAV, you configure Rhythmyx Folders to associate Rhythmyx Content Types with file MIME types. Then you associate the Rhythmyx Folders with a virtual Web site, which is specified by a WebDAV URL, and point third-party applications at the same URL. The Rhythmyx Folder (and any sub-Folders) appear in the explorer dialog of the third-party applications. When a user saves a file created in a third-party application in one of these WebDAV-enabled Folders, WebDAV uploads it into Rhythmyx as the Content Type associated with its MIME type.

In addition to configuring Rhythmyx to use WebDAV, you must determine if your third-party applications require special configuration. Some applications, such as Internet Explorer, are already WebDAV-enabled, and you can use them as they are to access Rhythmyx content. Other applications, such as Adobe Photoshop and Macromedia DreamWeaver, need to be installed with a WebDAV option or may require extra configuration to use WebDAV. Check your application's documentation for information about configuring WebDAV.

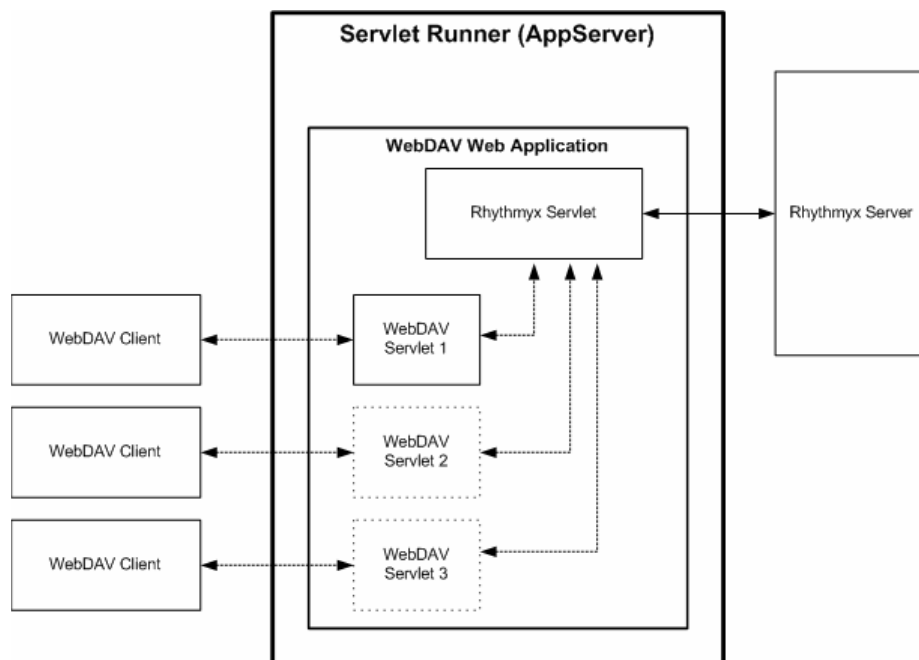
## Architecture of WebDAV Components in Rhythmyx

A WebDAV Servlet communicates with the Rhythmyx Server in the following manner:

The Rhythmyx WebDAV servlet communicates with the WebDAV client (WebDAV-enabled third-party application). It authenticates and passes requests from the client to the Rhythmyx servlet located in the WebDAV Web application. The Rhythmyx servlet passes the request to the Rhythmyx server. Responses from the Rhythmyx server pass through the Rhythmyx servlet to the WebDAV servlet, and finally to the WebDAV client.

You can create multiple WebDAV servlets in a single WebDAV Web application. You may also create multiple WebDAV applications. Since each WebDAV application has its own security constraints, using multiple applications lets you specify different security constraints for different servlets.

The following graphic illustrates the architecture of the WebDAV components in Rhythmyx



*Figure 1: WebDAV architecture in Rhythmyx*

Each instance of the WebDAV servlet supports a combination of a Rhythmyx Folder (and any sub-Folders of the specified Folder) with a set of Content Types. You will need several WebDAV servlets if you want to access multiple root-level Folders.

The WebDAV Web application must have security constraints; otherwise, it may not function.

**NOTE:** If you change your Rhythmyx Server and Port, you must change the values for server and port in all instances of your Rhythmyx Servlet.



---

## Prerequisites and Recommendations

- Each Content Type that you associate with a WebDAV-enabled folder must include fields for storing the following:
  - user with lock on Content Item;
  - size of uploaded file;
  - MIME type of uploaded file;
  - content of uploaded file.

These fields are mapped to WebDAV values in the WebDAV configuration file.

See the topic *Editing a WebDAV Configuration File* (on page 8) for information about how these fields are formatted.

- Content Types may validate fields that WebDAV does not upload. To avoid errors, remove validation from the Content Types that you associate with WebDAV folders if they process fields that are not uploaded (or set in other ways, for example, by a pre-exit). The WebDAV validator returns a warning if it finds <FieldRules> (elements that store validation) for fields that are not uploaded. See the topic “Adding Data Validation to a Content Editor” in the *Implementing Content Editors* document for help locating code that performs data validation.
- Rhythmyx WebDAV requires that a Web Server Security Provider be defined in Rhythmyx. Current versions of Rhythmyx include a Web Server Security Provider by default, but previous versions do not. If you are upgrading from a previous version, you must define a Web Server Security Provider. For details about implementing a Web Server Security Provider, see the topic “Web Server” in the *Rhythmyx Server Administrator Online Help*. NOTE: If you are not using other security providers included in Rhythmyx, remove them to avoid possible errors.
- To avoid unwanted behavior, put WebDAV Servlets that point to the same Rhythmyx Server in the same application server, and put WebDAV Servlets that point to different Rhythmyx Servers into different application servers.

---

## WebDAV Web Applications

Rhythmyx is installed with WebDAV Web applications for the following Web application servers:

- **Tomcat Version 4.1.27** (see "[Implementing WebDAV in Rhythmyx's Installation of Tomcat](#)" on page 8) in Rhythmyx 5.5; **Tomcat Version 5.5.7** (see "[Implementing WebDAV in Rhythmyx's Installation of Tomcat](#)" on page 8) in Rhythmyx 5.6; **Tomcat Version 4.1.24** in Rhythmyx Express Portal.
- **BEA WebLogic 8.1** (see "[Deploying the Rhythmyx WebDAV Servlet in BEA WebLogic](#)" on page 8);
- **IBM Websphere 5.1** (see "[Deploying the Rhythmyx WebDAV Servlet in IBM WebSphere](#)" on page 8) (Rhythmyx does not support WebDAV in WebSphere 5.0 and lower).

The .war or .ear files for these applications and their directories are:

```
<Rhythmyxroot>\InstallableApps\ibm\ws\rxwebdav.ear  
<Rhythmyxroot>\InstallableApps\bea\wls\rxwebdav.war  
<Rhythmyxroot>\InstallableApps\apache\tomcat\rxwebdav.war
```

Each of these directories contains a subdirectory for each Web application server on which WebDAV is supported. The subdirectory contains a WebDAV Web application. In other words, the WebDAV Web application for Tomcat is in the directory

```
<Rhythmyxroot>\InstallableApps\apache\Tomcat, which contains the file  
rxwebdav.war.
```

The rxwebdav Web application is automatically installed into the Tomcat application server that is delivered with the Rhythmyx Publisher and Rhythmyx Express Portal. You must deploy the rxwebdav Web application into the other application servers and other instances of Tomcat.

---

## Supported WebDAV Clients and Application Servers

Rhythmyx WebDAV supports the following WebDAV clients:

- Microsoft's Word 2002, Excel 2002, PowerPoint 2002, FrontPage 2002, and Project 2003. These products use Windows Explorer Web folders to access WebDAV in Rhythmyx.
- Adobe Acrobat Reader 6.0 uses Windows Explorer Web folders to access WebDAV in Rhythmyx.
- Macromedia Dreamweaver MX 2004 communicates with WebDAV in Rhythmyx directly; it does not require Windows Explorer Web folders.
- Adobe Photoshop 7.0.1 communicates with WebDAV in Rhythmyx directly; it does not require Windows Explorer Web folders.
- Windows Explorer on Windows XP and Windows 2000.
- MacOS WebDAV client that is shipped with the OS.

Rhythmyx WebDAV supports the following application servers:

- Tomcat 4.1.27 for Rhythmyx 5.5; Tomcat 5.5.7 for Rhythmyx 5.6; Tomcat 4.1.24 for Rhythmyx Express Portal
- BEA WebLogic 8.1
- IBM WebSphere 5.1

Tomcat 4.1.27 is shipped with Rhythmyx 5.5. Tomcat 5.5.7 is shipped with Rhythmyx 5.6. Tomcat 4.1.24 is shipped with Rhythmyx Express Portal. The Rhythmyx BEA WebLogic 8.1 Portal Connector allows users to run Rhythmyx through the BEA WebLogic 8.1 application server. The Rhythmyx IBM WebSphere Portal Connector does not yet allow users to run Rhythmyx through the IBM WebSphere 5.1 application server.

Rhythmyx WebDAV does not work with Xerces 2.0.2 and lower. (Rhythmyx 5.5 uses Xerces 2.5.0.)



## CHAPTER 2

# Implementing WebDAV

This chapter describes how to implement WebDAV in Rhythmyx. It includes a discussion of the different implementation options available then describes how to implement WebDAV in the default Tomcat Web application server installed with Rhythmyx, including a simple example of WebDAV implementation. A discussion of the deployment options for supported portals is also included. The final section provides troubleshooting hints for your implementation.

---

## Options for Implementing WebDAV Servlets

The following options are available when implementing WebDAV servlets:

- (Recommended option) You can implement all WebDAV servlets in one WebDAV Web application. The URL for the Web location of the Rhythmyx folder is: `http://host:port/WebDAVApplicationName/ServletURLPattern`, where `WebDAVApplicationName` is the name of the WebDAV servlet that makes the desired Rhythmyx folder available and `ServletURLPattern` is the complete path of the Rhythmyx folder in the Rhythmyx navigation pane. This approach simplifies implementation somewhat, since you only need to configure one Web application.
- You can implement one WebDAV Web application for each WebDAV servlet you implement. If you choose this option, the URL for the Web location of each Rhythmyx folder may be simpler. The drawback to this option is that you may have a large number of WebDAV Web applications.
- You can implement a combination of the two approaches.

If you implement WebDAV in the installation of Tomcat in Rhythmyx, you can edit the WebDAV files manually to add additional servlets to the default WebDAV Web application and further modify it. If you implement it in IBM Websphere or BEA WebLogic or an external version of Tomcat, see the specific application's documentation for the correct methods of editing the deployment descriptor and configuration file and implementing those edits.

---

# Implementing WebDAV in Rhythmyx's Installation of Tomcat

Rhythmyx includes a default WebDAV Web application. You must customize the files in the default application to implement your system's WebDAV setup.

This topic lists the procedure for implementing WebDAV in Rhythmyx's installation of Tomcat. For specific details about performing each step, see *Simple Example Implementation of WebDAV* (on page 8).

---

## NOTES:

The default implementation of Rhythmyx's WebDAV specifies Folders installed with FastForward. If you have installed both FastForward and Tomcat, you can access the default Web Folders with the users and passwords *rxuser/demo* and *admin1/demo*.

The sample WebDAV configuration file `<Rhythmyx root>\AppServer\webapps\rxwebdav\RxWebdavConfig_for_OldSample.xml` refers to the previous Rhythmyx samples which can be installed with the archive file `<Rhythmyx root>/Samples/DeprecatedSamples.pda`. If you choose to use this file, you must modify the RxWebDAV application deployment descriptor, `web.xml`, to refer to the Servlet that it configures.

For information about installing and using FastForward, see the FastForward documentation.

---

To implement WebDAV in Rhythmyx's installation of Tomcat:

- 1 To function in Rhythmyx, WebDAV requires a Web Server Security Provider. New installations of Rhythmyx include a Web Server Security Provider automatically. If you are upgrading from a previous version of Rhythmyx and have not already implemented a Web Server Security Provider, you must define one. For details about implementing a Web Server Security Provider, see the topic "Web Server" in the *Rhythmyx Server Administrator Online Help*. NOTE: If you are not using other security providers included in Rhythmyx, remove them to avoid possible errors.
- 2 The Roles that you want to have access to your WebDAV servlet must be defined in Rhythmyx, the rxwebdav deployment descriptor (`<Rhythmyx root>\AppServer\webapps\rxwebdav\WEB-INF\web.xml`), and Tomcat.
  - a) If the Roles are not already defined in Rhythmyx, add them. Associate the Roles with the Community that you want to have access to the WebDAV-enabled Folder. In addition add them to the Workflows assigned to your Content Types. Assign each Role to all States in its Workflows as an assignee.
  - b) Back up the default rxwebdav Web application deployment descriptor, `<Rhythmyx root>\AppServer\webapps\rxwebdav\WEB-INF\web.xml`. In your current copy, add these Roles if they are not already included in the `<SecurityConstraint>` element.
  - c) In the file `<Rhythmyx root>\AppServer\conf\tomcat-users.xml`, define the Roles if they are not already defined and associate them with the usernames and passwords that you want to use with WebDAV.

- 3 Create the folders that you want to associate with WebDAV under the Sites and Folders nodes in the Content Explorer navigation pane. In the Create Folder dialog **Folder Community** field, choose the value of the Community that you want to have access to your Servlet.
  - a) Choose *All Communities* or a specific Community (which must be the same as the Community entered in the rxwebdav configuration file).
  - b) Give the Roles that you want to have access to the Folder in Rhythmyx Write or Admin permission to the Folder.
- 4 Create a WebDAV Web application deployment descriptor, and configure the default WebDAV servlet and add additional WebDAV servlets:
  - c) Back up the original deployment descriptor, `<Rhythmyx root>\AppServer\webapps\rxwebdav\WEB-INF\web.xml` and open the current copy. There are four `<servlet>` nodes that contain the default WebDAV servlets, *CorporateInvestmentImages*, *CorporateInvestmentFiles*, *EnterpriseInvestmentImages* and *EnterpriseInvestmentFiles*.
  - d) In one of the default `<servlet>` nodes, change `<servlet-name>` to a custom name for your servlet, and change the name in the `<servlet-mapping>` element to the custom name. Change the name of the config file in `<init-param><param-value>` to a custom xml filename for your servlet. Be sure to change the name of the physical config file, (either `RxWebDAVConfigEIIImages.xml`, `RxWebDAVConfigEIFiles.xml`, `RxWebDAVConfigCIIImages.xml`, or `RxWebDAVConfigCIFiles.xml` to match the new value. Delete servlets and servlet mappings that you are not using.
  - e) If you are not using the default Rhythmyx Port, 9992, enter the correct value in the port `init-param`.

To add additional WebDAV servlets, see *Adding Additional WebDAV Servlets* (see ["Adding Additional WebDAV Servlets in Rhythmyx's Installation of Tomcat"](#) on page 12).

- 5 Configure a WebDAV configuration file for each WebDAV servlet. The configuration files for the default WebDAV servlets are:

```
<Rhythmyx
root>\AppServer\webapps\rxwebdav\RxWebdavConfigEIIImages.xml
<Rhythmyx
root>\AppServer\webapps\rxwebdav\RxWebdavConfigEIFiles.xml
<Rhythmyx
root>\AppServer\webapps\rxwebdav\RxWebdavConfigCIIImages.xml
and
<Rhythmyx
root>\AppServer\webapps\rxwebdav\RxWebdavConfigCIFiles.xml
```

Back up one of these files. Rename the current copy with the configuration file name you specified in your deployment descriptor. Edit this file to specify the Rhythmyx Folder and Community you want to make available through your WebDAV servlet, and specify the Content Types you want to manage using this servlet.



For each additional WebDAV servlet in the WebDAV application, make a copy of the configuration file in the same location, give it the name you specified in the `<init param><param value>` element for the servlet, and modify the Rhythmyx Folder, Community and Content Types.

For more information about editing the configuration file, see *Editing a WebDAV Configuration File* (on page 13).

- 6 To create an additional WebDAV application in Tomcat, you must copy and rename the `rxwebdav.war` used for your Web application server and deploy it to a different folder in `<Rhythmyx root>\AppServer\webapps`. Then you must make the changes listed in the above steps to the deployment descriptor and WebDAV config file(s).
- 7 After you make changes to your WebDAV configuration, restart your Web application server to apply them.
- 8 To confirm that implementation is correct, restart the Tomcat Server. In your Web browser:
  - Enter `http://hostname:9980/rxwebdav/[ServletURLPattern]?sys_command=getConfig`.  
If the WebDAV servlet is configured correctly in the application server the browser displays the contents of the WebDAV configuration file.

```
<?xml version="1.0" encoding="UTF-8" ?>
- <!--
  The WebDAV configuration information

  "root" - the virtual root path of the Rhythmyx for all resource
  specified in the request. It must start from "Site" or "Folder"
  For example, root="//Site/mysite", root="//Folder/myfolder".
  It must be an existing folder.

  "communityid" - the name of the community, which is used to communicate
  with Rhythmyx Server.

  "communityname" - the name of the community, this is only a reference
  to the communityid, but is not used by the WebDAV.

  "locale" - the locale that is used for the created items

  "deleteas" - an optional attribute, indicating the behavior of the
  DELETE operation (in DELETE, COPY and MOVE method).
  The possible values are:

  "purge": indicate the DELETE operation will purge the
  target items and/or folders. The purged items
  will not be able to recovered afterwards.

  "remove": indicate the DELETE operation will remove the
  folder relationships with the target items. The
  removed items can be recovered afterwards. This
  is the default behavior if this attribute is not
  specified.
```

Figure 2: Get Configuration screen

- Enter `http://hostname:9980/rxwebdav/[ServletURLPattern]?sys_command=validateConfig`.  
If the WebDAV servlet's references to Rhythmyx elements are correct, the screen displays the message "WebDAV configuration successfully validated".

The warnings displayed on the following screen appear if WebDAV is uploading a file into a Content Type with a required displaytitle field to avoid confusion over which field WebDAV displays as the item's title. If displaytitle is required, Rhythmyx WebDAV does the following when it uploads a file into Rhythmyx:

- It sets the Rhythmyx Content Item's displaytitle field to the file name.
- It sets the Rhythmyx Content Item's sys\_title field to the file name.

During processing after upload, WebDAV:

- Does not use the Rhythmyx Content Item's displaytitle field.
- Maps the Rhythmyx Content Item's sys\_title field to WebDAV's displayname property. When the value in one changes, the other sets itself to the new value.

Therefore, changing the Content Item's displaytitle field does not change the displayname that WebDAV shows to users. Explain this to your users who work with Rhythmyx WebDAV.

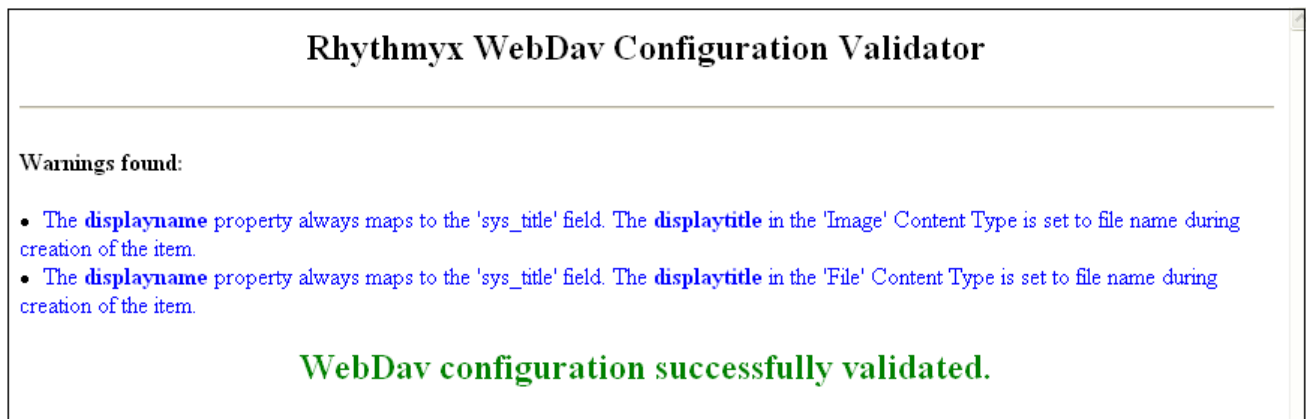


Figure 3: Validation screen

Note: In the WebDAV configuration file, the servlet URL pattern is the value of the *root* parameter in the <PSXWebdavConfigDef> element.

For example, in the sample WebDAV configuration file, the <PSXWebdavConfigDef> element appears as:

```
<PSXWebdavConfigDef root="//Sites/EnterpriseInvestments/Files"
communityname="Enterprise Investments" communityid="1002"
locale="en-us" deleteas="remove" publicValidTokens="y"
QEValidTokens="i">
```

Also, in the WebDAV deployment descriptor,

<RhythmyxRoot>\AppServer\webapps\rxwebdav\WEB-INF\web.xml, the servlet URL pattern is specified in the <servlet-mapping> element for the servlet in the <url-pattern> sub-element.

To use the getConfig command, enter:

```
http://hostname:9980/rxwebdav/Sites/EnterpriseInvestments/Files
?sys_command=getConfig.
```

- 9 Open the folders in *Windows Explorer* (see "Opening Web Folders in Windows Explorer" on page 14) or *Internet Explorer* (see "Opening Web Folders in Internet Explorer" on page 16), and if you plan to access the folders through a third-party application, configure the application as specified in its documentation.

WebDAV is now configured in your Rhythmyx system. You may now create Rhythmyx Content Items in Windows Explorer and your other third-party applications.

## Adding Additional WebDAV Servlets in Rhythmyx's Installation of Tomcat

One WebDAV Web application can support multiple WebDAV servlets. To add an additional servlet to a WebDAV Web application:

- 1 Back up one of the default configuration files. Rename the new file for the servlet you are creating. For example, create a servlet named *USMarketing* with a configuration file named `RxWebdavConfigMarketing.xml`.
- 2 Open the deployment descriptor (`<RhythmyxRoot>\AppServer\webapps\rxwebdav\WEB-INF\web.xml`) of the WebDAV Web application.

- 3 Find the servlet definition:

```
<servlet>
  <servlet-name>EnterpriseInvestmentImages</servlet-name>
  <display-name>Rx WebDAV for EnterpriseInvestment Images
folder</display-name>
  <description>Rhythmyx WebDAV for EnterpriseInvestment Images
folder</description>
  <servlet-
class>com.percussion.webdav.PSWebdavServlet</servlet-class>
  <init-param>
    <param-name>RxWebDAVConfig</param-name>
    <param-value>/RxWebDAVConfigEImages.xml</param-value>
    <description>The webdav configuration file path, which
is relative to the current web application</description>
  </init-param>
</servlet>
```

Copy this code and paste the copy immediately below the original.

- 4 Change `<servlet-name>EnterpriseInvestmentImages</servlet-name>` to `<servlet-name>USMarketing</servlet-name>`
- 5 Change the value of the `RxWebDAVConfig` parameter to the name of the WebDAV configuration file you want to implement with this servlet. For example:

```
<servlet>
  <servlet-name>USMarketing</servlet-name>
  <display-name>Rx WebDAV for Marketing folder</display-name>
  <description>Rx WebDAV for Marketing folder</description>
  <servlet-class>com.percussion.webdav.PSWebdavServlet</servlet-class>
  <init-param>
    <param-name>RxWebDAVConfig</param-name>
    <param-value>/RxWebdavConfigMarketing.xml</param-value>
```

```
    <description>The webdav configuration file path, which is relative
to the current web application</description>
  </init-param>
</servlet>
```

**6** Find the `<servlet-mapping>` elements:

```
  <servlet-mapping>
    <servlet-name>CorporateInvestmentImages</servlet-name>
    <url-pattern>/Sites/CorporateInvestments/Images/*</url-
pattern>
  </servlet-mapping>
  <servlet-mapping>
    <servlet-name>CorporateInvestmentFiles</servlet-name>
    <url-pattern>/Sites/CorporateInvestments/Files/*</url-
pattern>
  </servlet-mapping>
  <servlet-mapping>
    <servlet-name>EnterpriseInvestmentImages</servlet-name>
    <url-pattern>/Sites/EnterpriseInvestments/Images/*</url-
pattern>
  </servlet-mapping>
  <servlet-mapping>
    <servlet-name>EnterpriseInvestmentFiles</servlet-name>
    <url-pattern>/Sites/EnterpriseInvestments/Files/*</url-
pattern>
  </servlet-mapping>
  <servlet-mapping>
    <servlet-name>Rhythmyx</servlet-name>
    <url-pattern>/Rhythmyx/*</url-pattern>
  </servlet-mapping>
```

and add one for the new servlet:

```
  <servlet-mapping>
<servlet-name>USMarketing</servlet-name>
<url-pattern>/USMarketing/*</url-pattern>
  </servlet-mapping>
```

**7** Repeat steps 3 to 5 for each servlet you want to implement.

**8** Restart the Web application.

**9** The URL for accessing the new Servlet is  
`http://host:port/rxwebdav/USMarketing`.

## Editing a WebDAV Configuration File

A WebDAV configuration file specifies the Rhythmyx Folder and Community the WebDAV servlet makes available, and the Content Types that users can manage through WebDAV. The WebDAV configuration file DTD is `RxWebDAVconfig.dtd`, and the default WebDAV configuration files in Rhythmyx are `RxWebDAVConfigEIIImages.xml`, `RxWebDAVConfigEIFiles.xml`, `RxWebDAVConfigCIImages.xml` and `RxWebDavConfigCIFiles.xml`. They are located in `<Rhythmyx root>\AppServer\webapps\rxwebdav`.

A WebDAV configuration file consists of the following nodes:

`PSXWebdavConfigDef` - specifies the Rhythmyx Folder and Community the WebDAV servlet makes available. This node includes the following attributes:

`root` - specifies the Rhythmyx Folder. The value of this attribute must begin with either `“//Sites/”` (if the Folder you want to make available is a Site Folder) or `“//Folders/”` (if the Folder you want to make available is a child of the Folders node).

`communityname` - specifies the name of the Community you want to make available. NOTE: The Community property of the root folder can be either *All Communities* or the Community you specify here.

`communityid` - specifies the ID of the Community (Rhythmyx assigns the ID when you save the Community; you can find it in parentheses next to the Community Name on the Communities Editor on the System tab of Content Explorer).

`locale` – Locale for Content Items created in this Folder.

`deleteas` – How the delete function for items and Folders behaves. Possible values are *remove* and *purge*. Note that if a Content Item is in either Public or Quick Edit State, Rhythmyx does not perform any action on that item. If the Content Item is in any other state, Rhythmyx executes the action normally.

- `remove` – Default. If a WebDAV-enabled Folder is deleted in Rhythmyx or a third-party application, permanently delete the Folder in Rhythmyx and remove the Content Items from the Folder, but save them in Rhythmyx.
- `Purge` – If a WebDAV-enabled Folder is deleted in Rhythmyx or a third-party application, permanently delete the Folder and the Content Items in Rhythmyx.

`publicValidTokens` - specifies a list of flags for Public States in the Workflow. Used with the attribute `QEVvalidTokens` to implement automatic checkout of Content Items in the WebDAV Folder. Default value is *y*.

`QEVvalidTokens` - specifies a list of flags for Quick Edit States in the Workflow. Used with the `publicValidTokens` attribute to implement automatic checkout of Content Items in the WebDAV Folder. Default value is *i*.

`PSXWebdavContentType` – One or more of these nodes may be present. Each node specifies a Content Type that users can maintain in the Folder the servlet makes available.

`id` - specifies the ID of the Content Type; you can find it in parentheses next to the Content Type Name on the Content Types Editor on the System tab of Content Explorer.

`name` - specifies the name of the Content Type.

contentfield - specifies the Content Type field that stores the data file for the Content Type.

ownerfield - specifies the Content Type field that stores the lock owner data from WebDAV.

default - specifies whether the Content Type is the default Content Type for the servlet. The default Content Type is the Content Type that processes any MIME Types not specified for another Content Type in the configuration. Only one Content Type in the configuration can be specified as the default Content Type. Values are *true* or *false*.

MIMETypes - specifies the MIME types of files that will be processed by the Content Type. Each MIME type is specified in a separate Mime Type child node.

PropertyMap - stores the mappings of WebDAV properties to the Rhythmyx fields. The name attribute of the PSXPropertyFieldNameMapping element specifies the WebDAV property. The FieldName child of the PSXPropertyFieldNameMapping specifies the Rhythmyx Content Type field that stores the data for the specified WebDAV property.

PSXPropertyFieldNameMapping name="displayname" - Required for FastForward Content Types. Specifies the display title of the Content Item. WebDAV passes the filename as the display title.

PSXPropertyFieldNameMapping name="getcontenttype" - Required. Specifies the MIME type of the content.

PSXPropertyFieldNameMapping name="getcontentlength" - Required. Specifies the length, in bytes, of the content.

ExcludeFolderProperties - specifies which Rhythmyx custom properties of the parent WebDAV Folder should not be inherited by sub-folders created in WebDAV. Note that the Folder Name and Description are automatically not inherited (you must specify a new Folder Name and Description is always blank).

PropertyName - sys\_pubFileName is listed because it specifies the name of the folder to publish to. This name would most likely differ in the sub-folder.

---

NOTE: Fields specified to store data for a WebDAV configuration must be from a System, Shared, or Local definition. You cannot use child fields to store WebDAV data.

---

## Simple Example Implementation of WebDAV

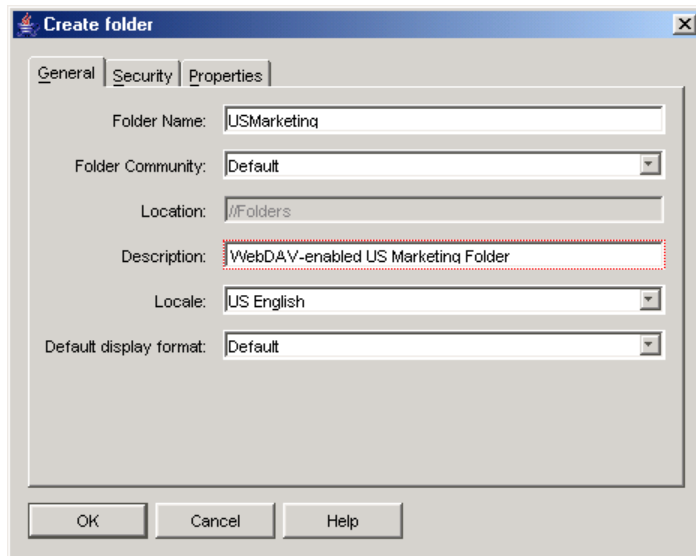
To illustrate a typical implementation using WebDAV, let us configure two WebDAV servlets to support the Rhythmyx Folders for two departments in a company. The two WebDAV servlets will reside in the same Web application and run on the default Rhythmyx Web Application Server (Tomcat).

We will configure:

- a WebDAV servlet to support the USMarketing Folder for the Default Community. The folder is accessible to the Marketing\_Author and Marketing\_Artist Roles as well as the Default Role.
- a WebDAV servlet to support the USEditorial Folder for the Editorial Community. The folder is accessible only to Editor and Administrator Roles.

To set up our sample WebDAV configuration:

- 1 First, as an Administrator, we create the folders in Content Explorer that we want to WebDAV-enable. We create the USMarketing Folder and assign it to the Default Community.



We give the Role Admin Read, Write, and Admin permission, and we give the Roles Marketing\_Author, Marketing\_Artist, and Default Read and Write permissions:

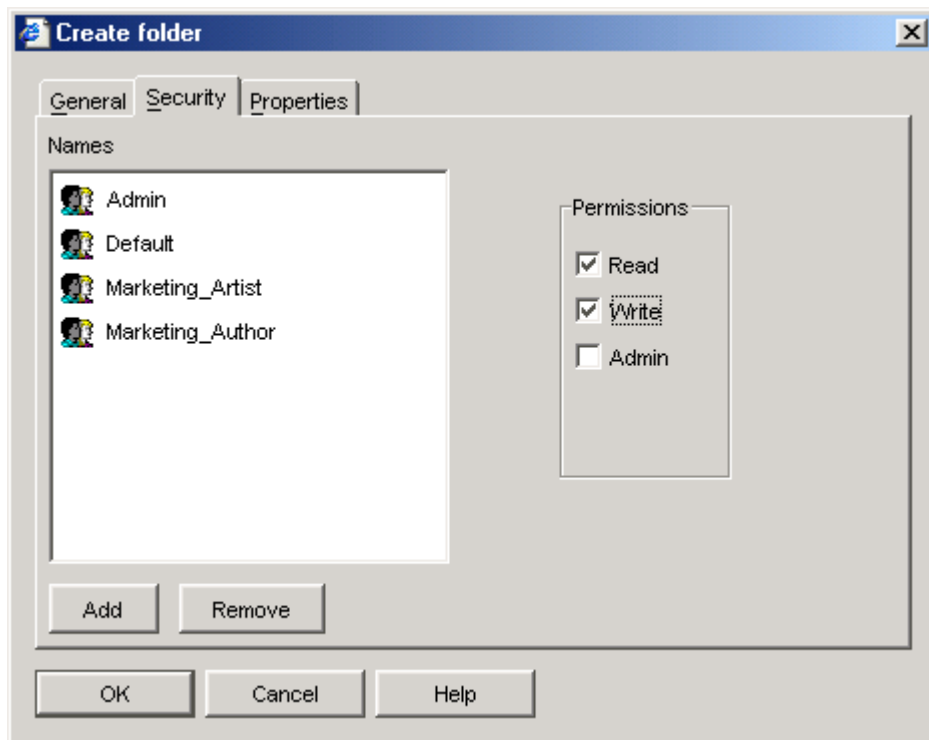


Figure 4: Create Folder Dialog, Security Tab

Then we create the USEditorial folder. We assign it to the Editorial Community and give the Admin Role Read, Write, and Admin permission and the Editor Role Read and Write permission.

Our completed folder structure appears as:

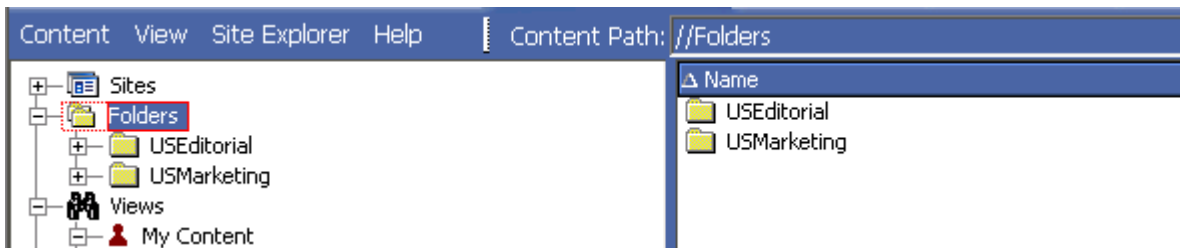


Figure 5: WebDAV-enabled Folders in Rhythmyx

- 2 We create a Web Server Security Provider. (Note: If you have a new installation of Rhythmyx, the Web Server Security Provider is already included, otherwise you must create it.) For details about implementing a Web Server Security Provider, see the topic “Web Server” in the Rhythmyx Server Administrator online Help. NOTE: If you are not using other security providers included in Rhythmyx, remove them to avoid possible errors.
- 3 We add the Roles that will have access to our Folders to the Rhythmyx servlet that is located in our WebDAV Web application. We back up `<Rhythmyxroot>\AppServer\webapps\rxwebdav\WEB-INF\web.xml`. In the current copy, we edit the `<SecurityConstraint>` element of the Rhythmyx servlet:

```
<security-constraint>
  <web-resource-collection>
    <web-resource-name>Entire Application</web-resource-
name>
    <url-pattern>/*</url-pattern>
  </web-resource-collection>
  <auth-constraint>
    <!-- Define a list of roles with rights to access the Servlet -
Roles need to exist in both the App Server and Rhythmyx -->
    <!-- For Tomcat (App Server), you may need to modify tomcat-
root/conf/tomcat-users.xml file, add userid, password and assign
Rhythmyx roles to the userid -->
    <!-- Replace the value of role-name element with a character
'*' if the roles are resolved by Rhythmyx, but not by Tomcat -->
    <role-name>Admin</role-name>
    <role-name>Default</role-name>
    <role-name>rxrole</role-name>
    <role-name>Web_Admin</role-name>
    <role-name>CI_Members</role-name>
    <role-name>Internet_Members</role-name>
    <role-name>CI_Admin_Members</role-name>
    <role-name>Internet_Admin_Members</role-name>
    <role-name>Marketing_Author</role-name>
    <role-name>Marketing_Artist</role-name>
  </auth-constraint>
  <user-data-constraint>
    <transport-guarantee>NONE</transport-guarantee>
  </user-data-constraint>
</security-constraint>
```

- 4 We also add the Roles to `<Rhythmyx root>\AppServer\conf\tomcat-users.xml` and associate them with the user and password we want to use to access WebDAV:

```
<?xml version='1.0' encoding='utf-8'?>
```



```

<tomcat-users>
  <role rolename="CI_Admin_Members"/>
  <role rolename="Intranet_Admin_Members"/>
  <role rolename="CI_Members"/>
  <role rolename="tomcat"/>
  <role rolename="Internet_Members"/>
  <role rolename="Internet_Admin_Members"/>
    <role rolename="Marketing_Artist"/>
  <role rolename="Marketing_Author"/>
  <role rolename="manager"/>
  <role rolename="Default"/>
  <role rolename="Admin"/>
  <role rolename="admin"/>
  <role rolename="Web_Admin"/>
  <user username="admin1" password="demo"
roles="Admin,Default,Web_Admin,Internet_Members,Internet_Admin_Members,M
arketing_Author,Marketing_Artist"/>
  <user username="tomcat" password="tomcat"
roles="admin,manager,tomcat"/>
  <user username="admin2" password="demo"
roles="Admin,Default,Web_Admin,CI_Members,CI_Admin_Members,Marketing_Au
thor,Marketing_Artist"/>
  <user username="rxuser" password="demo"
roles="Admin,Default,Web_Admin,Internet_Members,Internet_Admin_Members,C
I_Members,CI_Admin_Members"/>
</tomcat-users>

```

## 5 We modify the WebDAV deployment descriptor to define our two servlets:

We change the default servlet to have:

- USMarketing as a <servlet-name>;
- RxWebdavConfigMarketing.xml as a config file name in <init-param><param-value>.

We copy the <servlet> node and paste the copy under the original. We change the copied servlet to have:

- USEditorial as a <servlet-name>;
- RxWebdavConfigEditorial.xml as a config file name.

```

<servlet>
  <servlet-name>USMarketing</servlet-name>
  <display-name>Rhythmyx WebDAV Router</display-name>
  <description>Rhythmyx WebDAV Router</description>
  <servlet-class>com.percussion.webdav.PSWebdavServlet</servlet-class>
  <init-param>
    <param-name>RxWebDAVConfig</param-name>
    <param-value>/RxWebdavConfigMarketing.xml</param-value>
    <description>The webdav configuration file path, which is relative
to the current web application</description>
  </init-param>
</servlet>
<servlet>
  <servlet-name>USEditorial</servlet-name>
  <display-name>Rhythmyx WebDAV Router</display-name>

```

```

<description>Rhythmyx WebDAV Router</description>
<servlet-class>com.percussion.webdav.PSWebdavServlet</servlet-class>
<init-param>
  <param-name>RxWebDAVConfig</param-name>
  <param-value>RxWebdavConfigEditorial.xml</param-value>
  <description>The webdav configuration file path, which is relative
to the current web application</description>
</init-param>
</servlet>

```

We modify the servlet mappings to specify these two servlets:

```

<servlet-mapping>
  <servlet-name>USMarketing</servlet-name>
  <url-pattern>/USMarketing/*</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>USEditorial</servlet-name>
  <url-pattern>/USEditorial/*</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>Rhythmyx</servlet-name>
  <url-pattern>/Rhythmyx/*</url-pattern>
</servlet-mapping>

```

We delete the servlets and servlet mappings that we are not using to prevent WebDAV from failing.

**6** Next, we open each configuration file and specify the Rhythmyx Folder and Community that the WebDAV servlet makes available. In `RxWebdavConfigMarketing`, we make the following changes:

- we change root to `//Folders/USMarketing`;
- to make the Folder `USMarketing` available to members of the Default Community, we change the value of `communityname` to `Default`;
- we leave the default Content Type of File and modify the Image Content Type node to define the PDFfile Content Type;
- in the PDFfile definition:
  - we specify `contentfield` as `pdfupload`, the name of the PDFFile Content Type field that stores WebDAV content;
  - we change `ownerfield` to `lockuser`, the name of the PDFFile Content Type field that stores the WebDAV user who has a lock on the item;
  - we set `default` to `false`, because the File Content Type is already the default.
- we change `<Mimetype>` to the appropriate value for PDFfile, `application/pdf`;
- in `<PropertyMap>`, we map the WebDAV field `getcontenttype` to the Rhythmyx Content Type field `pdfupload_type`. These are the fields in each application that store the MIME type. We also map the WebDAV field `getcontentlength` to the Rhythmyx Content Type field `pdfupload_size`. These are the fields in each application that store the Content Length.

```

<PSXWebdavConfigDef root="//Folders/USMarketing" communityname="Default"
communityid="10" locale="en-us" deleteas="remove">

```

```

<!--
  Defines one or more supported content type for WebDAV

  "id" - the id of the content type in Rhythmyx

  "name" - the name of the content type in Rhythmyx, this is only
used as a
          reference to the id above.

  "contentfield" - the field name for the content

  "ownerfield" - the field name for the lock owner that is
submitted by various WebDAV clients

  "default" - "true" or "false". There must be only one element
with the value "true".
          If it is "false", must define one or more mime
types (hence must define the "Mimetypes" element);
          If it is "true", the "Mimetypes" element is
ignored.
-->

  <PSXWebdavContentType id="318" name="PDFFile"
contentfield="pdfupload" ownerfield="lockuser" default="false">

  <!-- Defines the mime type(s) that be accepted by the current
Rhythmyx content type.-->

  <MimeTypes>
    <MimeType>application/pdf</MimeType>
  </MimeTypes>

  <!--
  Defines the mapper between WebDAV properties and its
corresponding field name of the current Rhythmyx content type

  Couple required mappings:

  "getcontenttype" - maps to a field for the mime-type.

  "getcontentlength" - maps to a field for the content length
-->

  <PropertyMap>
    <PSXPropertyFieldNameMapping name="displayname">
      <FieldName>displaytitle</FieldName>
    </PSXPropertyFieldNameMapping>
    <PSXPropertyFieldNameMapping name="getcontenttype">
      <FieldName>pdfupload_type</FieldName>
    </PSXPropertyFieldNameMapping>
    <PSXPropertyFieldNameMapping name="getcontentlength">
      <FieldName>pdfupload_size</FieldName>
    </PSXPropertyFieldNameMapping>
  </PropertyMap>
</PSXWebdavContentType>

```

```
<PSXWebdavContentType id="309" name="File" contentfield="item"
ownerfield="webdavowner" default="true">

  <PropertyMap>
    <PSXPropertyFieldNameMapping name="displayname">
      <FieldName>displaytitle</FieldName>
    </PSXPropertyFieldNameMapping>
    <PSXPropertyFieldNameMapping name="getcontenttype">
      <FieldName>item_type</FieldName>
    </PSXPropertyFieldNameMapping>
    <PSXPropertyFieldNameMapping name="getcontentlength">
      <FieldName>item_size</FieldName>
    </PSXPropertyFieldNameMapping>
  </PropertyMap>
</PSXWebdavContentType>
<!--
  A list of excluded folder properties when creating a folder in
WebDAV.
  A created folder will inherit all properties of its parent folder
except
  the list of excluded properties and the folder name.
-->
<ExcludeFolderProperties>
  <PropertyName>sys_pubFileName</PropertyName>
</ExcludeFolderProperties>
```

```
</PSXWebdavConfigDef>
```

In RxWebdavConfigEditorial, we change root to //Folders/USEditorial, and we make the Folder Editorial available to members of the Editorial Community. We make the Image Content Type the default, and add a node for the Logo Content Type. The changes we make to the <PSXWebdavContentType> node for Logo are similar to those we made for the PDFFile Content Type, above.

```
<PSXWebdavConfigDef root="//Folders/USEditorial"
communityname="Editorial" communityid="1001" locale="en-us"
deleteas="remove">
```

```
<!--
```

Defines one or more supported content type for WebDAV

"id" - the id of the content type in Rhythmyx

"name" - the name of the content type in Rhythmyx, this is only used as a reference to the id above.

"contentfield" - the field name for the content

"ownerfield" - the field name for the lock owner that is submitted by various WebDAV clients

"default" - "true" or "false". There must be only one element with the value "true".

If it is "false", must define one or more mime types (hence must define the "Mimetypes" element);

ignored.

```
-->
```

```
<PSXWebdavContentType id="307" name="Image" contentfield="img1"
ownerfield="webdavowner" default="true">
```

```
<!--
```

Defines the mapper between WebDAV properties and its corresponding field name of the current Rhythmyx content type

Couple required mappings:

"getcontenttype" - maps to a field for the mime-type.

"getcontentlength" - maps to a field for the content length

```
-->
```

```
<PropertyMap>
```

```
<PSXPropertyFieldNameMapping name="displayname">
```

```
<FieldName>displaytitle</FieldName>
```

```
</PSXPropertyFieldNameMapping>
```

```
<PSXPropertyFieldNameMapping name="getcontenttype">
```

```
<FieldName>img1_type</FieldName>
```

```
</PSXPropertyFieldNameMapping>
```

```
<PSXPropertyFieldNameMapping name="getcontentlength">
```

```
<FieldName>img1_size</FieldName>
```

```
</PSXPropertyFieldNameMapping>
```

```
</PropertyMap>
```

```
</PSXWebdavContentType>
```

```
<PSXWebdavContentType id="12" name="Logo" contentfield="Logobody"
ownerfield="lockuser" default="false">
```

```
<!-- Defines the mime type(s) that be accepted by the current
Rhythmyx content type.-->
```

```
<MimeType>
```

```
<MimeType>image/gif</MimeType>
```

```
</MimeType>
```

```
<PropertyMap>
```

```
<PSXPropertyFieldNameMapping name="displayname">
```

```
<FieldName>displaytitle</FieldName>
```

```
</PSXPropertyFieldNameMapping>
```

```
<PSXPropertyFieldNameMapping name="getcontenttype">
```

```
<FieldName>logobody_type</FieldName>
```

```
</PSXPropertyFieldNameMapping>
```

```
<PSXPropertyFieldNameMapping name="getcontentlength">
```

```
<FieldName>logobody_size</FieldName>
```

```
</PSXPropertyFieldNameMapping>
```

```
</PropertyMap>
```

```
</PSXWebdavContentType>
```

```
<!--
```

A list of excluded folder properties when creating a folder in WebDAV.

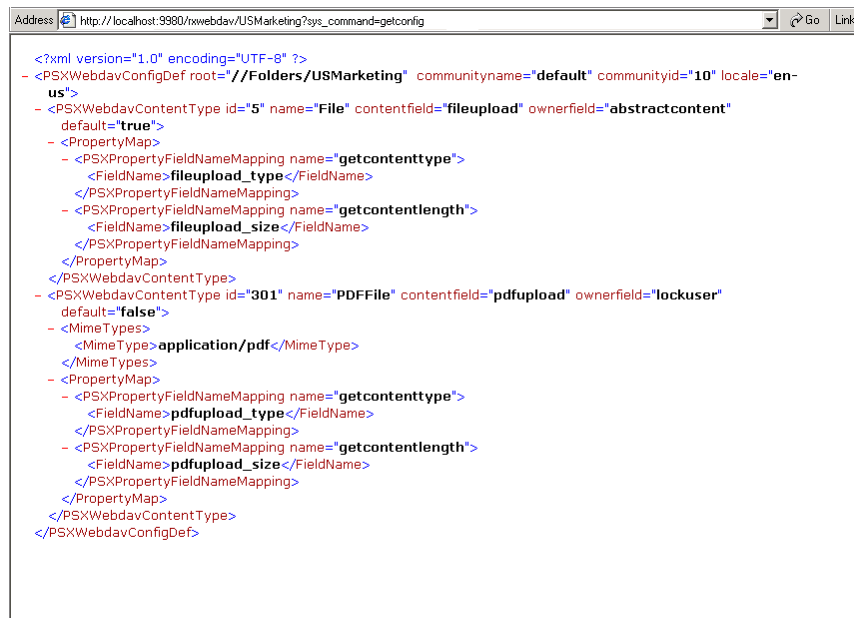
A created folder will inherit all properties of its parent folder except

the list of excluded properties and the folder name.

```
-->
<ExcludeFolderProperties>
  <PropertyName>sys_pubFileName</PropertyName>
</ExcludeFolderProperties>
```

```
</PSXWebdavConfigDef>
```

- After deployment is complete, we restart the Tomcat Server. In our Web browser, we enter: `http://hostname:9980/rxwebdav/USMarketing?sys_command=getConfig`  
If the WebDAV servlet has deployed correctly the browser displays the contents of the WebDAV configuration file.



```
<?xml version="1.0" encoding="UTF-8" ?>
- <PSXWebdavConfigDef root="//Folders/USMarketing" communityname="default" communityid="10" locale="en-us">
- <PSXWebdavContentType id="5" name="File" contentfield="fileupload" ownerfield="abstractcontent" default="true">
- <PropertyMap>
- <PSXPropertyFieldNameMapping name="getcontenttype">
  <FieldName>fileupload_type</FieldName>
</PSXPropertyFieldNameMapping>
- <PSXPropertyFieldNameMapping name="getcontentlength">
  <FieldName>fileupload_size</FieldName>
</PSXPropertyFieldNameMapping>
</PropertyMap>
</PSXWebdavContentType>
- <PSXWebdavContentType id="301" name="PDFFile" contentfield="pdfupload" ownerfield="lockuser" default="false">
- <MimeTypes>
  <MimeType>application/pdf</MimeType>
</MimeTypes>
- <PropertyMap>
- <PSXPropertyFieldNameMapping name="getcontenttype">
  <FieldName>pdfupload_type</FieldName>
</PSXPropertyFieldNameMapping>
- <PSXPropertyFieldNameMapping name="getcontentlength">
  <FieldName>pdfupload_size</FieldName>
</PSXPropertyFieldNameMapping>
</PropertyMap>
</PSXWebdavContentType>
</PSXWebdavConfigDef>
```

We repeat using

`http://hostname:9980/rxwebdav/USEditorial?sys_command=getConfig`

- Then, we enter:  
`http://hostname:9980/rxwebdav/USMarketing?sys_command=validateConfig`  
If the WebDAV servlet is configured correctly, the screen displays a confirmation message.



We repeat using `http://hostname:9980/rxwebdav/USEditorial?sys_command=validateConfig`.

- 8 Now, we can create Network Places in *Windows Explorer* (see "[Opening Web Folders in Windows Explorer](#)" on page 16) that make these Rhythmyx folders available through Windows Explorer. We start Tomcat. In the Add Network Place Wizard, we enter the locations for our WebDAV servlets in the format `http://localhost:<appserver port>/<WebDAV application>/<servlet URL pattern>`. The servlet URL pattern is specified in the `root` parameter of the `<PSXWebDAVConfigDef>` element of the configuration file.

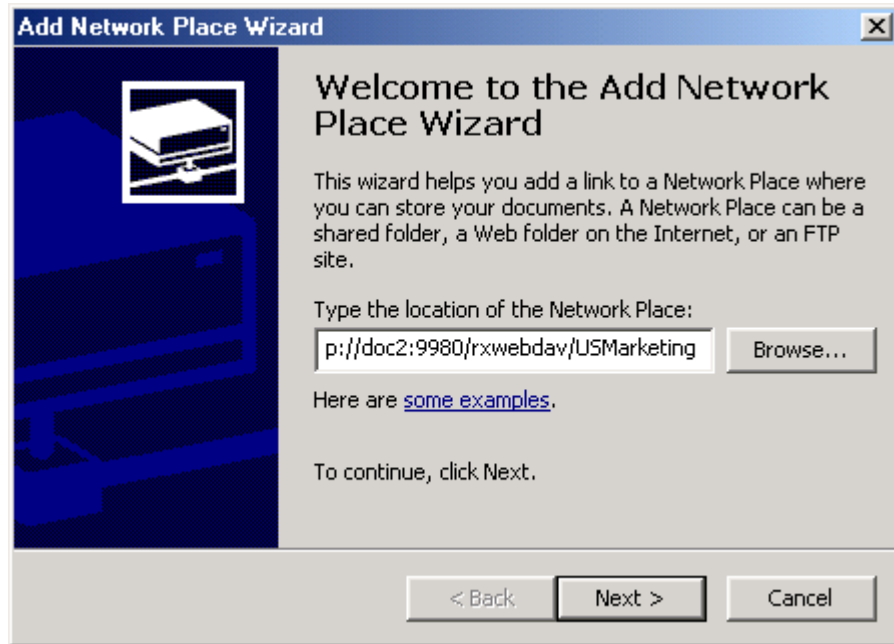


Figure 6: Add Network Place Wizard

Windows Explorer will ask us to log in when it connects to the WebDAV URL. The username and password must be defined in the `<Rhythmyx root>\AppServer\conf\tomcat-users.xml` file. When the login is authenticated, Windows Explorer displays the new Network Place:

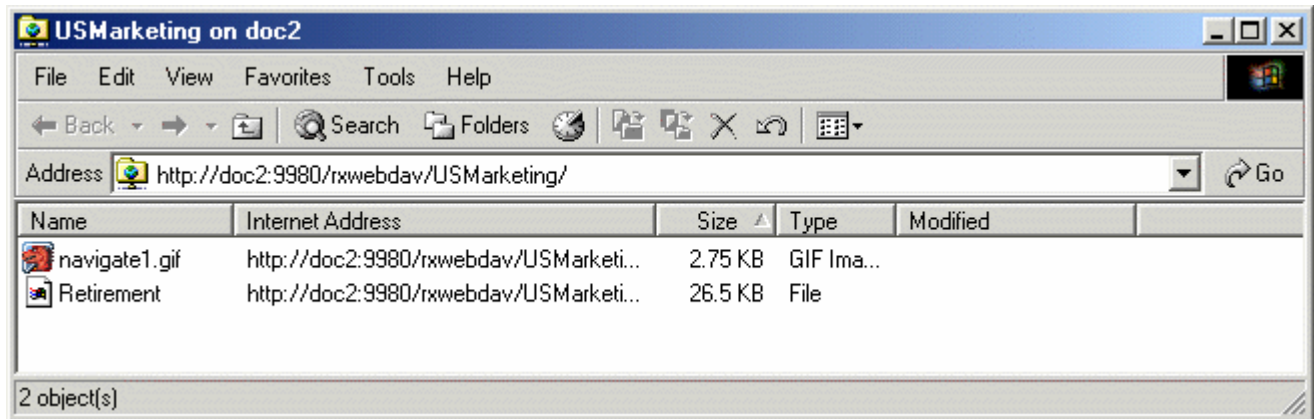


Figure 7: WebDAV opened in Windows Explorer

- 9 We perform any steps that are required to connect a third-party application to the WebDAV enabled folders.

---

## Deploying the Rhythmyx WebDAV Servlet in BEA WebLogic

You can deploy Rhythmyx's WebDAV Servlet in a BEA WebLogic Portal. The process involves defining a Group, Role, and User in WebLogic for accessing Rhythmyx WebDAV, and deploying the `rxwebdav.war` file into the Portal application. You must also create the Role in Rhythmyx that you defined in WebLogic for accessing WebDAV.

---

NOTE: The following procedure refers to Rhythmyx's default WebDAV deployment descriptor, WebDAV configuration file, Web Folder, and WebDAV Servlet. Most likely, you will customize these components. See *Implementing WebDAV in Rhythmyx's Installation of Tomcat* (on page 11) in the *Implementing WebDAV in Rhythmyx* document for instructions on customizing them. Replace file locations specified with those in your BEA WebLogic Portal.

This procedure assumes that you have installed FastForward.

---

To implement WebDAV in the BEA WebLogic Portal:

- 1 To function in Rhythmyx, WebDAV requires a Web Server Security Provider. New installations of Rhythmyx include a Web Server Security Provider automatically. If you are upgrading from a previous version of Rhythmyx and have not already implemented a Web Server Security Provider, you must define one. For details about implementing a Web Server Security Provider, see the topic "Web Server" in the *Rhythmyx Server Administrator Online Help*. NOTE: If you are not using other security providers included in Rhythmyx, remove them to avoid possible errors.
- 2 Create the folders that you want to associate with WebDAV under the Sites and Folders nodes in the Content Explorer navigation pane. In this example, we use the default Web Folder `Sites/EnterpriseInvestments/Files` and leave the default Folder properties.
- 3 Open the Administration Console by entering `http://localhost:7001/console` in a browser or, in Windows, by clicking the [Start] button and selecting *Programs > BEA WebLogic Platform 8.1 > Examples > WebLogic Portal > Server Admin Console*, and log in.
- 4 In the Navigation pane, expand *Portal > Security > Realms* to open the Portal > Realms dialog, and under Name click *myrealm*.  
The Administration Console displays the *myrealm* page.
- 5 Add a WebLogic group:
  - a) At the top of the page, click the User Management tab, then *Manage Groups Within This Security Realm*.
  - b) On the Groups page, click *Configure a New Group*.
  - c) Add a new Group for the Portal Connector. In this example, we use the Group `rxgroup`.
  - d) Click the [Apply] button.



- 6 Add a WebLogic Role and associate this Role to the WebLogic Group, `rxgroup`. Note: This Role must exist in Rhythmyx, WebLogic, and the Rhythmyx WebDAV application deployment descriptor. Add this Role to WebLogic in the following steps.
  - a) At the top of the page click *myrealm* then click *Manage Roles Within This Security Realm*.
  - b) The Administration Console displays the Global Roles page.
  - c) Click *Configure a New Global Role*.
  - d) Add a Role to WebLogic that will have access to the WebDAV application. In this example, we add `Web_Admin`.
  - e) Click the [Apply] button.
  - f) Click the Conditions tab.
  - g) Under **Role Condition**, select *Caller is a member of the Group* and click the [Add] button.
  - h) In the **Enter Group Name** field, enter the WebDAV group (`rxgroup`) and click the [Add] button.
  - i) Click the [OK] button.
  - j) Click the [Apply] button.
- 7 Add a WebLogic user and add it to `rxgroup`.
  - a) At the top of the screen click *myrealm* then click *Manage Users within this Security Realm*.
  - b) On the Users page, click *Configure a new user*.
  - c) Add a user for the Portal Connector. In our example, we use the user `rxuser` and password `password`.
  - d) Click [Apply].
  - e) Click the Groups tab.

Move `rxgroups` under Current Groups and click the [Apply] button.
- 8 By default, the Role *Web\_Admin* is included in the Rhythmyx WebDAV application deployment descriptor. If you are using a different Role that is not specified in the deployment descriptor, add it before deploying `rxwebdav.war`. See the topic *Modifying the Rhythmyx Web Application Deployment Descriptor in the Portal Connector for BEA WebLogic* document for instructions.
- 9 In the Navigation pane, expand `portal/Deployments/Web Application Modules`.
- 10 Click *Deploy a New Web Application Module*.
- 11 Follow the instructions on the screens to deploy `rxwebdav.war`. Use the following information:
  - the files are located in: `<Rhythmyx Root>/InstallableApps/bea/wls/rxwebdav.war`;
  - leave the Deployment Target as `portalServer`.

- 12 In the Navigation pane, click *Portal/Deployments/Web Application Modules*.
- 13 If the Role that you added to WebLogic does not already exist in Rhythmyx, add it to Rhythmyx. In this example, we have added `Web_Admin` as the WebLogic Role for WebDAV.
  - a) Add the WebLogic Role to the Rhythmyx Server Administrator.
  - b) Associate the WebLogic Role with a Rhythmyx Community. In our example, we associate `Web_Admin` with the Internet Community.
  - c) Then associate the WebLogic Role with the Workflows you plan to use for Content Items that will be located in WebDAV folders. Also assign the WebLogic Role as an assignee to specific Workflow States within the Workflows. For help, see the topic "Adding a Role" in the online *Rhythmyx Server Administrator Help*, the topic "Associating a Role with a Community" in the online *Rhythmyx Content Management Help* and the section "About Workflows in Rhythmyx" in the online *Rhythmyx Content Management Help*.
- 14 After deployment is complete, restart the BEA Server and the Rhythmyx Server. In your Web browser:
  - Enter:  
`http://hostname:7001/rxwebdav/Sites/EnterpriseInvestments/Files?sys_command=getConfig`. When prompted for your username and password, enter `rxuser` and `password` or the username and password that you have added to WebLogic. If the WebDAV servlet has deployed correctly the browser displays the contents of the WebDAV configuration file:

```
<?xml version="1.0" encoding="UTF-8" ?>
- <!--
  The WebDAV configuration information

  "root" - the virtual root path of the Rhythmyx for all resource
          specified in the request. It must start from "Site" or "Folder"
          For example, root="//Site/mysite", root="//Folder/myfolder".
          It must be an existing folder.

  "communityid" - the name of the community, which is used to communicate
                  with Rhythmyx Server.

  "communityname" - the name of the community, this is only a reference
                    to the communityid, but is not used by the WebDAV.

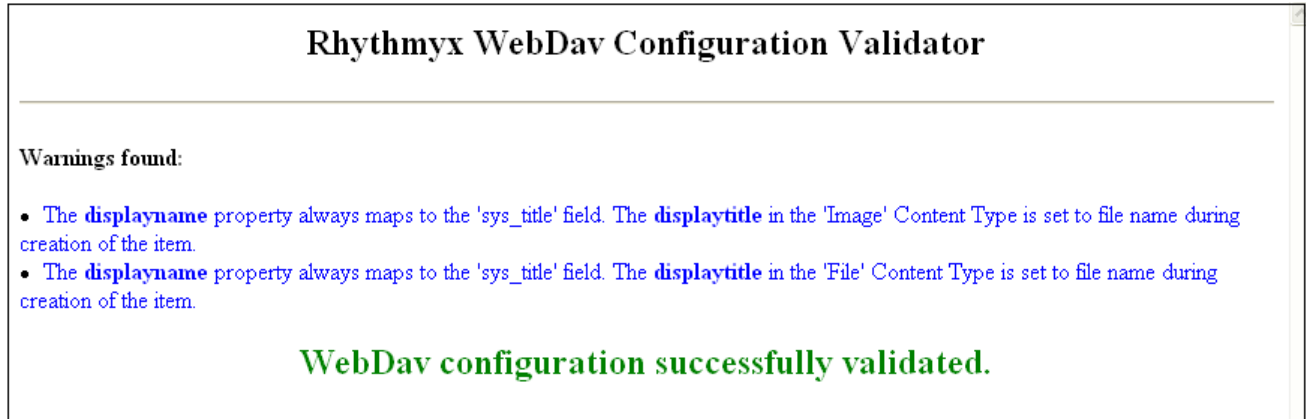
  "locale" - the locale that is used for the created items

  "deleteas" - an optional attribute, indicating the behavior of the
                DELETE operation (in DELETE, COPY and MOVE method).
                The possible values are:

                "purge": indicate the DELETE operation will purge the
                          target items and/or folders. The purged items
                          will not be able to recovered afterwards.

                "remove": indicate the DELETE operation will remove the
                           folder relationships with the target items. The
                           removed items can be recovered afterwards. This
                           is the default behavior if this attribute is not
                           specified.
```

- Enter:  
`http://hostname:7001/rxwebdav/Sites/EnterpriseInvestments/Files?sys_command=validateConfig`. If the WebDAV servlet is configured correctly, the screen displays the following message:



*Figure 8: Validation of WebDAV Configuration File*

- 15** Open the Sites/EnterpriseInvestments/Files Web Folder in *Windows Explorer* (see "Opening Web Folders in Windows Explorer" on page 29) or *Internet Explorer* (see "Opening Web Folders in Internet Explorer" on page 32) and if you plan to access the Folders through a third-party application, configure the application as specified in its documentation.

WebDAV is now deployed in BEA WebLogic and configured in your Rhythmyx system. You may now create Rhythmyx Content Items in Windows Explorer and your other third-party applications.

---

## Deploying the Rhythmyx WebDAV Servlet in IBM WebSphere

You can deploy Rhythmyx's WebDAV Servlet in the IBM Websphere Portal. The process involves configuring security in WebSphere, and installing the `rxwebdav.ear` file in the WebSphere Portal application.

---

Notes:

The following example assumes that WebSphere is configured to take the user and group definitions from the operating system. The `rxwebdav.ear` file maps the user `rxuser` (password `demo`) and the group `rxgroup` to the Role `Web_Admin`. You may use a different user, group, and Roles, but you should define and map them prior to completing the following process. Define new Roles and map them to users and groups during deployment or in WebSphere's Application Assembly Tool.

The following procedure refers to Rhythmyx's default WebDAV deployment descriptor, WebDAV configuration file, Web Folder, and WebDAV Servlet. Most likely, you will customize these components. See *Implementing WebDAV in Rhythmyx's Installation of Tomcat* (on page 11) for instructions on customizing them.

This procedure assumes that you have installed FastForward.

---

To install the WebDAV servlet in the IBM WebSphere Portal:

- 1** To function in Rhythmyx, WebDAV requires a Web Server Security Provider. New installations of Rhythmyx include a Web Server Security Provider automatically. If you are upgrading from a previous version of Rhythmyx and have not already implemented a Web Server Security Provider, you must define one.  
  
For details about implementing a Web Server Security Provider, see the topic "Web Server" in the *Rhythmyx Server Administrator Online Help*.
- 2** In the Rhythmyx Server Administrator, if you have configured a Windows NT Security Provider that you are not using, remove the Windows NT Security Provider. If you do not remove it, your local OS users with access to WebDAV must also have access to the Rhythmyx Server to avoid Security Provider conflicts.
- 3** Create the folders that you want to associate with WebDAV under the Sites and Folders nodes in the Content Explorer navigation pane. In this example, we use the default Web Folder `Sites/EnterpriseInvestments/Files` and leave the default Folder properties.
- 4** Add `rxuser` and `rxgroup` (or the user and group that you have defined) to your local operating system.
- 5** The Roles that you want to have access to your WebDAV servlet must be defined in Rhythmyx, the `rxwebdav` deployment descriptor and WebSphere.
  - a) Define the Roles in Rhythmyx.
  - b) Define the Roles in WebSphere. See the topic *WebSphere Setup Steps in the Rhythmyx Connector for IBM WebSphere* document for instructions.

- c) Define the Roles in the Rhythmyx WebDAV Application Deployment Descriptor. See the topic Customizing the Rhythmyx Web Application in the Rhythmyx Connector for IBM WebSphere document for instructions
- 6** Log in to the WebSphere Administrative Console.
- 7** In the navigation pane, select `Security > User Registries > Local OS`.
- 8** In the Local OS Registry screen, enter your WebSphere **Server User ID** and **Server User Password** (`rxuser` and `demo` or the user and password that you have defined).
- 9** Click [**Apply**] and then click [**OK**].  
The Global Security screen opens.
- 10** Set the following properties:
  - Enabled: check
  - Enforce Java 2 Security: uncheck
  - Active User Registry: Local OSLeave the other properties at the default values.
- 11** Click [**Apply**] and then, in the Menu Bar, click `Save`.  
The Save screen opens.
- 12** Click [**Save**]
- 13** Open a command line window and point to `[WebSphere root]/AppServer/bin`.
- 14** To stop the WebSphere server, enter:  

```
stopServer.bat server1 -user rxuser -password demo
```

  
Note: Change `rxuser` to your username and `demo` to your password.
- 15** To restart the WebSphere server enter:  

```
startServer.bat server1 -user rxuser -password demo
```

  
Note: Change `rxuser` to your WebSphere username and `demo` to your WebSphere password.
- 16** Log in to the WebSphere Administrative Console again.
- 17** In the navigation pane, select `Applications > Install New Application`.  
The Preparing for the application installation screen opens.
- 18** Click `Local path` and browse to `<Rhythmyx Root>/InstallableApps/ibm/wls/rxwebdav.ear`.
- 19** Press [**Next**] to proceed to the following screen. In each screen, press [**Next**] until you reach Step 4: Map Security Roles to Users and Groups. Leave the default Role to User/Group mappings or change the mappings to include user(s) and group(s) that you have added.
- 20** Press [**Next**].
- 21** In the Summary screen, press [**Finish**].  
The Installing Screen opens.

- 22** When you see the message `Application rxwebdav` installed successfully, click [**Save to Master Configuration**].

The Save screen opens.

- 23** Click [**Save**].

- 24** Start the deployed `rxwebdav` application. In the Navigation Panel, select `Applications > Enterprise Applications`.

The Enterprise Applications screen opens.

- 25** Check `rxwebdav` and click [**Start**].

When it is started, the red X next to the application name is replaced with a green arrow.

Deployment is now complete.

- 26** After deployment is complete, run the WebSphere Server and the Rhythmyx Server.

- Enter:  
`http://hostname:9080/rxwebdav/Sites/EnterpriseInvestments/Files?sys_command=getConfig`. When prompted for your username and password, enter `rxuser` and `password` or the username and password that you have added to WebLogic. If the WebDAV servlet has deployed correctly the browser displays the contents of the WebDAV configuration file:

```
<?xml version="1.0" encoding="UTF-8" ?>
- <!--
  The WebDAV configuration information

  "root" - the virtual root path of the Rhythmyx for all resource
           specified in the request. It must start from "Site" or "Folder"
           For example, root="//Site/mysite", root="//Folder/myfolder".
           It must be an existing folder.

  "communityid" - the name of the community, which is used to communicate
                  with Rhythmyx Server.

  "communityname" - the name of the community, this is only a reference
                    to the communityid, but is not used by the WebDAV.

  "locale" - the locale that is used for the created items

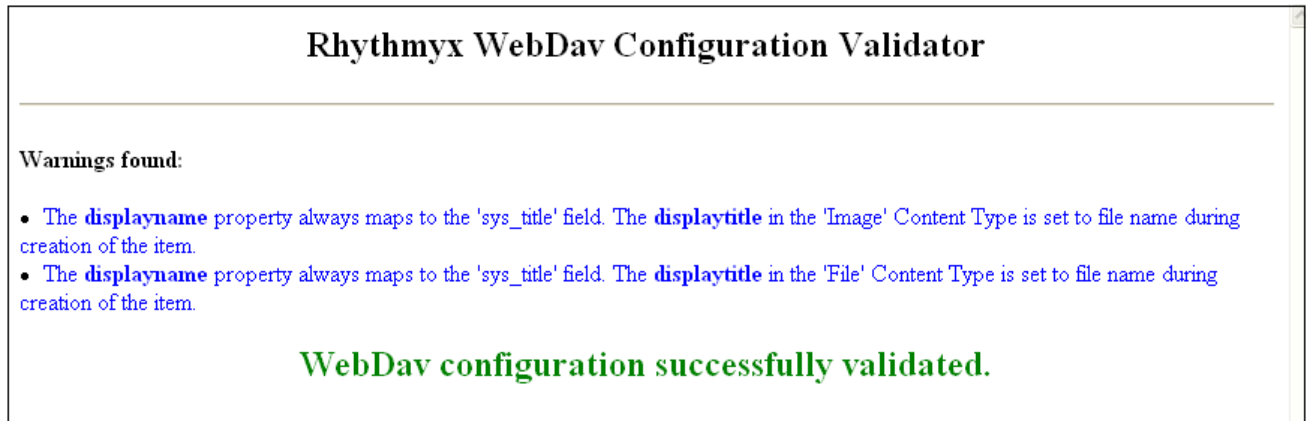
  "deleteas" - an optional attribute, indicating the behavior of the
                DELETE operation (in DELETE, COPY and MOVE method).
                The possible values are:

    "purge":    indicate the DELETE operation will purge the
                 target items and/or folders. The purged items
                 will not be able to recovered afterwards.

    "remove":   indicate the DELETE operation will remove the
                 folder relationships with the target items. The
                 removed items can be recovered afterwards. This
                 is the default behavior if this attribute is not
                 specified.
```

*Figure 9: WebDAV Configuration File*

- Enter:  
`http://hostname:9080/rxwebdav/Sites/EnterpriseInvestments/Files?sys_command=validateConfig`. If the WebDAV servlet is configured correctly, the screen displays the following message:



*Figure 10: Validation of WebDAV Configuration File*

Note: If you cannot access the WebDAV Servlet at port 9080, in the WebSphere Administrative Console look up the port that the rxwebdav application uses:

- a) In the navigation pane, click Servers > Application Servers.  
 The Applications Servers screen opens.
- b) Click server1.
- c) Click Web Container.
- d) Click HTTP Transport.
- e) Use the first Port listed in the table on the HTTP Transport screen.

- 27** Open the Sites/EnterpriseInvestments/Files Web Folder in *Windows Explorer* (see "Opening Web Folders in Windows Explorer" on page 33) or *Internet Explorer* (see "Opening Web Folders in Internet Explorer" on page 36), and if you plan to access the Folder through a third-party application, configure the application as specified in its documentation.

WebDAV is now deployed in IBM WebSphere and configured in your Rhythmyx system. You may now create Rhythmyx Content Items in Windows Explorer and your other third-party applications.

---

# Deploying the Rhythmyx WebDAV Servlet in Rhythmyx Express Portal

You can deploy Rhythmyx's WebDAV Servlet into the Rhythmyx Express Portal by unzipping the rxwebdav.war file into the Portal's webapps folder.

- 1 Associate the user who will access Rhythmyx WebDAV in Rhythmyx Express Portal with a Role defined with the same name in both Rhythmyx and Rhythmyx Express Portal. The Role name is case sensitive; the Roles *admin* in Rhythmyx Express Portal and *Admin* in Rhythmyx do not have the same name.

---

Note: All of the default Rhythmyx Express Portal users except *admin* and *anon* are associated with a Role that is defined in both Rhythmyx and Rhythmyx Express Portal.

---

For more information about defining users and Roles:

- For Rhythmyx Express Portal, see Rhythmyx Express Portal's *Administrator's Guide*;
  - For Rhythmyx, see the online *Rhythmyx Server Administrator Help*.
- 2 In Rhythmyx, associate the Role for accessing WebDAV with a Rhythmyx Community and the Workflows you plan to use for Content Items that will be located in WebDAV folders. Also assign the Role as an assignee to specific Workflow States within the Workflows. For help, see the topic "Adding a Role" in the online *Rhythmyx Server Administrator Help*, the topic "Associating a Role with a Community" in the online *Rhythmyx Content Management Help* and the section "About Workflows in Rhythmyx" in the online *Rhythmyx Content Management Help*.
  - 3 In <Rhythmyx Express Portal root>/webapps, create a subfolder named rxwebdav. Unzip <Rhythmyx root>/InstallableApps/Apache/Tomcat/rxwebdav.war into <Rhythmyx Express Portal root>/webapps/rxwebdav.
  - 4 In <Rhythmyx Express Portal root>\webapps\rxwebdav\WEB-INF\web.xml add the Role to the <SecurityConstraint> element of the Rhythmyx servlet. For help, see ***the topic Simple Example Implementation*** of WebDAV (on page 18) .
  - 5 Create the folders that you want to associate with WebDAV under the Sites and Folders nodes in the Content Explorer navigation pane. In the lowest subfolder's properties, give the Role for accessing WebDAV *Write* or *Admin* privileges.

In this example, we use the default Rhythmyx 5.6 FastForward Web Folder Sites/EnterpriseInvestments/Files and leave the default Folder properties. If you are using a different default or you want to create different WebDAV-enabled Folders in Rhythmyx, you must edit the existing configuration files or create new ones. See the ***topics Editing a WebDAV Configuration File*** (on page 17) and ***Simple Example Implementation of WebDAV*** (on page 18) for help.

- 6 If you have not used the default Rhythmyx server and port (localhost and 9992), edit the WebDAV configuration file <Rhythmyx Express Portal>\webapps\rxwebdav\WEB\_INF\web.xml to specify the server and port that you are using. The elements that contain these values are:



```

<init-param>
  <param-name>Host</param-name>
  <param-value>localhost</param-value>
  <description>The host name where Rhythmyx is installed,
defaults to "localhost".</description>
</init-param>
<init-param>
  <param-name>Port</param-name>
  <param-value>9992</param-value>
  <description>The port number where Rhythmyx servlet is
installed, defaults to "9992".</description>
</init-param>

```

- 7** After deployment is complete, restart Rhythmyx Express Portal and the Rhythmyx Server. In your Web browser, enter:

```
http://hostname:9981/rxwebdav/Sites/EnterpriseInvestments/Files?sys_command=getConfig
```

When prompted for your username and password, enter the Rhythmyx Express Portal username and password that you have configured for accessing WebDAV. If the WebDAV servlet has deployed correctly the browser displays the contents of the WebDAV configuration file:

```

<?xml version="1.0" encoding="UTF-8" ?>
- <!--
  The WebDAV configuration information

  "root" - the virtual root path of the Rhythmyx for all resource
  specified in the request. It must start from "Site" or "Folder"
  For example, root="//Site/mysite", root="//Folder/myfolder".
  It must be an existing folder.

  "communityid" - the name of the community, which is used to communicate
  with Rhythmyx Server.

  "communityname" - the name of the community, this is only a reference
  to the communityid, but is not used by the WebDAV.

  "locale" - the locale that is used for the created items

  "deleteas" - an optional attribute, indicating the behavior of the
  DELETE operation (in DELETE, COPY and MOVE method).
  The possible values are:

  "purge":  indicate the DELETE operation will purge the
  target items and/or folders. The purged items
  will not be able to recovered afterwards.

  "remove": indicate the DELETE operation will remove the
  folder relationships with the target items. The
  removed items can be recovered afterwards. This
  is the default behavior if this attribute is not
  specified.

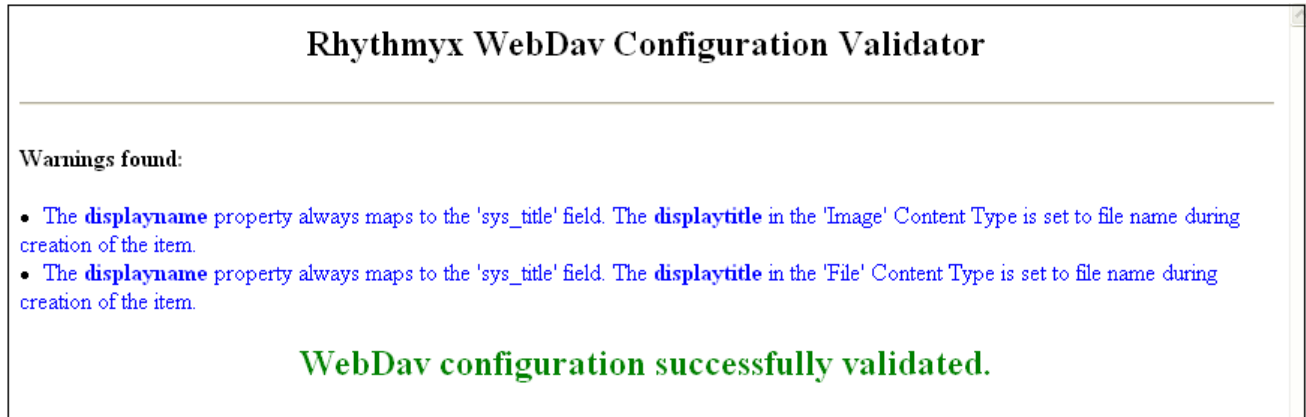
```

*Figure 11: Get Configuration screen*

Enter:

```
http://hostname:9981/rxwebdav/Sites/EnterpriseInvestments/Files?sys_command=validateConfig
```

If the WebDAV servlet is configured correctly, the screen displays the following message:



*Figure 12: Validation screen*

- 8 Open the Sites/EnterpriseInvestments/Files Web Folder in *Windows Explorer* (see "[Opening Web Folders in Windows Explorer](#)" on page 37) or *Internet Explorer* (see "[Opening Web Folders in Internet Explorer](#)" on page 39) . If you plan to access the Folders through a third-party application, configure the application as specified in its documentation.
- 9 WebDAV is now deployed in Rhythmyx Express Portal and configured in your Rhythmyx system. When you run the Window Express Portal server, you may now manage Rhythmyx Content Items in Windows Explorer and your other third-party applications.

---

## Troubleshooting WebDAV

If your WebDAV implementation is not working, use the following HTTP commands to check your configuration:

- `getconfig`: retrieves the configuration file;
- `help`: lists help for other commands;
- `validateconfig`: checks the validity of the configuration and returns a message indicating whether the configuration passed or failed the validation. If it fails the validation, the returned message includes a list of errors and warnings.

Issue these commands as values of the `sys_command` parameter:

```
http://localhost:9980/rxwebdav/Sites/EnterpriseInvestments/Files?sys_command=getconfig
http://localhost:9980/rxwebdav/Sites/EnterpriseInvestments/Files?sys_command=help
http://localhost:9980/rxwebdav/Sites/EnterpriseInvestments/Files?sys_command=validateconfig
```



## CHAPTER 3

# Working with WebDAV

This chapter outlines specific techniques for working with WebDAV, including

- How to access WebDAV-enabled Folders in Windows Explorer and Internet Explorer
- How to work with Rhythmyx Content in WebDAV-enabled Folders
- How WebDAV works with Rhythmyx Workflow

---

## Opening Web Folders in Windows Explorer

In Windows, you can access the WebDAV server from Web Folders.

You can create Web Folders that correspond to your WebDAV-enabled Rhythmyx Folders in Windows 2000 and later versions. No special configuration is required to access WebDAV folders through Windows Explorer.

Before you can access a Rhythmyx folder in Windows Explorer through Web Folders, a WebDAV servlet must be deployed in your application server.

To set up a Web Folder through Windows Explorer:

- 1 In the Windows Explorer navigation pane, click on the My Network Places node.
- 2 Double-click the Add Network Place icon.
- 3 Windows Explorer displays the Add Network Place wizard.
- 4 Enter the URL of the WebDAV server you want to access from Windows Explorer. The specific URL depends on the Web application server you use. Use the format:  
`http://<WebDAV host>:<WebDAV port>/<WebDAV application>/<servlet URL pattern>`.  
The URLs for the default Web application server (Tomcat) are:  
`http://hostname:9980/rxwebdav/Sites/EnterpriseInvestments/Files`  
and  
`http://hostname:9980/rxwebdav/Sites/EnterpriseInvestments/Images`.  
  
NOTE: The servlet URL pattern is specified in the *root* parameter of the `<PSXWebDAVConfigDef>` element of the configuration file.
- 5 When you click [**N**ext], Windows Explorer displays the Enter Network Password dialog. Enter your rxwebdav Web application username and password.
- 6 When you click [**N**ext], Windows Explorer displays the Completing the Add Network Place dialog. Click [**F**inish].
- 7 Windows Explorer adds the Folder as a Network Place. Now it appears when you click the My Network Places node. You can add files to this location, and open, modify, and save files in the location. All of your changes will appear in the Folder of the same name in Rhythmyx, and all of the changes that you make to the Folder in Rhythmyx will appear in this Folder.

When you attempt to access a file in this location, Windows Explorer may display the Enter Network Password dialog. Enter your rxwebdav Web application username and password to complete the action.

---

If your version of Windows Explorer does not display an “Add Network Place” icon in the “My Network Places” node, open your WebDAV Folders in *Internet Explorer* (see "[Opening Web Folders in Internet Explorer](#)" on page 39) instead.

---

---

# Opening Web Folders in Internet Explorer

In Windows, you can access the WebDAV server from Internet Explorer.

You can create Web Folders that correspond to your WebDAV-enabled Rhythmyx Folders in Version 5.0 and above of Internet Explorer; however, the Web Folders feature must be installed in Internet Explorer. If the Web Folders feature is not installed in your Internet Explorer installation, go to [support.microsoft.com](http://support.microsoft.com) (<http://support.microsoft.com>) or refer to Internet Explorer's help for installation instructions.

Before you can access a Rhythmyx folder in Internet Explorer through Web Folders, a Rhythmyx WebDAV server must exist.

To set up a Web Folder through Internet Explorer:

- 1 In the menu bar, choose *File > Open*.

Internet Explorer displays the Open dialog.

- 2 Enter the URL of the WebDAV server you want to access from Internet Explorer. The specific URL depends on the Web application server you use. Use the format: `http://<WebDAV host>:<WebDAV port>/<WebDAV application>/<servlet URL pattern>`. The URLs for the default Web application server (Tomcat) are:  
`http://hostname:9980/rxwebdav/Sites/EnterpriseInvestments/Files`  
and  
`http://hostname:9980/rxwebdav/Sites/EnterpriseInvestments/Images`.

NOTE: The servlet URL pattern is specified in the Root parameter of the configuration file.

- 3 Check **Open as Web Folder**.

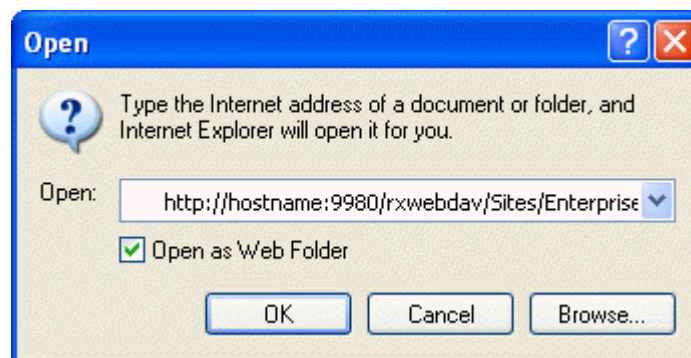


Figure 13: Open dialog

- 4 Click [OK].
- 5 Enter your network username and password if Internet Explorer prompts you for them.

- 6 Internet Explorer adds the Folder. You can add files to this location, and open, modify, and save files in the location. All of your changes will appear in the Folder of the same name in Rhythmyx, and all of the changes that you make to the Folder in Rhythmyx will appear in this Folder.

When you attempt to access a file in this location, Internet Explorer may display the Enter Network Password dialog. Enter your rxwebdav Web application username and password to complete the action.



---

## Opening Web Folders in Mac OS X

In a Mac OS X, you can access the WebDAV server from Web Folders.

You can use the Connect to Server command to access a WebDAV server.

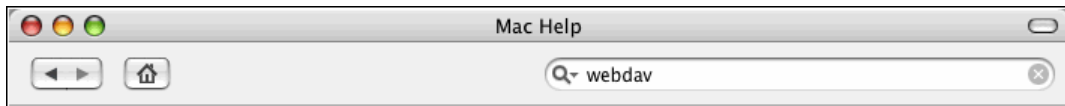
To set up a Web Folder through a Mac OS:

- 1 In the Finder, select *Go > Connect to Server*.

Enter the address of the server in the Server Address field.

The server address should be in a form similar to this: `http://servername.com/path`

- 2 Click the Connect button.



If you change your network settings (by opening Network preferences or by choosing a new location from the Apple menu) while you are connected to the WebDAV server, you should disconnect from the server, then reconnect to it.

To disconnect, select the server and choose Eject from the File menu.

---

## Working with Rhythmyx Content in Web Folders

Some third-party applications require you to set up WebDAV-enabled folders to communicate with a WebDAV server; others require more specialized configuration. Consult your application's documentation to determine whether it supports WebDAV and for instructions about configuring WebDAV support.

Once you have configured WebDAV in Rhythmyx, you can:

- add new Content Items to Rhythmyx directly from your third party applications by saving them to the WebDAV Folder (you will be prompted for your rxwebdav Web application login);
- add new Content Items to Rhythmyx by dragging them into Web Folders in Windows Explorer;
- access existing Content Items directly from your third-party application or Windows Explorer.

In all cases, you will be prompted for your rxwebdav Web application login when you initially access the WebDAV folder. New Files are added to Rhythmyx in the Content Type specified for their MIME Type, or to the default Content Type if you have not specified a MIME Type.

In Rhythmyx, if you copy or create a Content Item in a WebDAV-enabled Folder, the Content Item will appear in the Folder from third-party applications only if the Folder's configuration file handles the MIME type associated with the Content Item.

---

## WebDAV and Rhythmyx Workflow

Content Items in WebDAV folders are subject to standard Rhythmyx Workflow restrictions. Two issues in particular affect WebDAV.

First, when a user attempts to open a file in a WebDAV-enabled directory, Rhythmyx attempts to check out the associated Content Item to that user. If the Content Item is not in a Public State, the checkout occurs normally. If the Content Item is in a Public State, the WebDAV client issues a second request to Transition the Content Item to Quick Edit, then checkout the Content Item. The following processing occurs:

1. The WebDAV client requests the Content Item.
2. Rhythmyx returns a message that the Content Item is not available. The message includes the Content Item metadata, part of which is the value of the CONTENTVALID Flag.
3. The WebDAV client compares the value of CONTENTVALID flag to the values in the publicValidTokens parameter in the WebDAV servlet configuration.
4. If any of these values match, WebDAV assumes the Content Item is in a Public state and issues another request to Transition the Content Item to the first State in the Workflow where CONTENTVALID matches the value of the QEValidTokens in the WebDAV servlet configuration.
5. Once this Transition is completed successfully, Rhythmyx checks the Content Item out to the user. Rhythmyx also records a Transition comment indicating that the Transition was performed by WebDAV.

Note that the default values of these attributes:

publicValidToken=y

QEValidToken=i

These values match the standard CONTENTVALID values used for Public (y) and Quick Edit (i) States in Rhythmyx Workflows. If you use different values for these States, you will need to change the values of these attributes in the WebDAV configuration to match.

Next, the State of a Content Item affects the processing of Purge requests from WebDAV. When you attempt to delete a Content Item from a WebDAV folder WebDAV issues a PURGE request. The value of the `deleteas` attribute of the WebDAV configuration determines the processing of this request. If the value of the `deleteas` attribute is *purge*, then Rhythmyx purges the Content Item from the Repository. The Content Item cannot be recovered. If the value of the `deleteas` attribute is *remove*, Rhythmyx only removes the Content Item from the Folder; it still exists in the Repository and can be recovered. In either case, however, if the Content Item is in a Public State, no processing occurs.



# Appendices



## APPENDIX I

---

## Default Rhythmyx WebDAV Configuration File

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- The WebDAV configuration information

    "root" - the virtual root path of the Rhythmyx for all resource
             specified in the request. It must start from "Site" or
"Folder"
             For example, root="//Site/mysite",
root="//Folder/myfolder".
             It must be an existing folder.

    "communityid" - the name of the community, which is used to
communicate
                 with Rhythmyx Server.

    "communityname" - the name of the community, this is only a
reference
                 to the communityid, but is not used by the
WebDAV.

    "locale" - the locale that is used for the created items

    "deleteas" - an optional attribute, indicating the behavior of the
DELETE operation (in DELETE, COPY and MOVE method).
                 The possible values are:

                 "purge":  indicate the DELETE operation will purge
the
                 target items and/or folders. The purged
items
                 will not be able to recovered afterwards.
                 "remove": indicate the DELETE operation will remove
the
                 folder relationships with the target items.
The
                 removed items can be recovered afterwards.
This
                 is the default behavior if this attribute
is not
                 specified.

    "publicValidTokens"
                 - comma separated list of chars for public state
                 default "y"

    "QEValidTokens"
                 - comma separated list of char for quick-edit state
                 default "i"

```

```
-->
<PSXWebdavConfigDef root="//Sites/CorporateInvestments/Files"
communityname="Corporate Investments" communityid="1002" locale="en-us"
deleteas="remove"
    publicValidTokens="y" QEValidTokens="i">

<!--
    Defines one or more supported content type for WebDAV

    "id" - the id of the content type in Rhythmyx

    "name" - the name of the content type in Rhythmyx, this is only
used as a
        reference to the id above.

    "contentfield" - the field name for the content

    "ownerfield" - the field name for the lock owner that is
submitted by WebDAV clients

    "default" - "true" or "false". There must be only one element
with the value "true".
        If it is "false", must define one or more mime
types (hence must define the "Mimetypes" element);
        If it is "true", the "Mimetypes" element is
ignored.
-->
    <PSXWebdavContentType id="307" name="Image" contentfield="img1"
ownerfield="webdavowner" default="false">

    <!-- Defines the mime type(s) that be accepted by the current
Rhythmyx content type.-->

    <MimeTypes>
        <MimeType>image/gif</MimeType>
        <MimeType>image/bmp</MimeType>
        <MimeType>image/jpeg</MimeType>
        <MimeType>image/png</MimeType>
    </MimeTypes>

    <!--
    Defines the mapper between WebDAV properties and its
corresponding field name of the current Rhythmyx content type

    Required mappings:

        "getcontenttype" - maps to a Rhythmyx Content Editor field
for the mime-type.

        "getcontentlength" - maps to a Rhythmyx Content Editor field
for the content length.

    Rhythmyx WebDAV automatically maps the following:
```



```

        "displayname" - always maps to Rhythmyx sys_title field

        "creationdate" - maps to Rhythmyx sys_contentcreatedate
field

        "getlastmodified" - maps to Rhythmyx
sys_contentlastmodifieddate field

        "getcontentlanguage" - always maps to Rhythmyx sys_lang
field

-->
<PropertyMap>
  <PSXPropertyFieldNameMapping name="displayname">
    <FieldName>displaytitle</FieldName>
  </PSXPropertyFieldNameMapping>
  <PSXPropertyFieldNameMapping name="getcontenttype">
    <FieldName>img1_type</FieldName>
  </PSXPropertyFieldNameMapping>
  <PSXPropertyFieldNameMapping name="getcontentlength">
    <FieldName>img1_size</FieldName>
  </PSXPropertyFieldNameMapping>
</PropertyMap>
</PSXWebdavContentType>

  <PSXWebdavContentType id="309" name="File" contentfield="item"
ownerfield="webdavowner" default="true">

    <PropertyMap>
      <PSXPropertyFieldNameMapping name="displayname">
        <FieldName>displaytitle</FieldName>
      </PSXPropertyFieldNameMapping>
      <PSXPropertyFieldNameMapping name="getcontenttype">
        <FieldName>item_type</FieldName>
      </PSXPropertyFieldNameMapping>
      <PSXPropertyFieldNameMapping name="getcontentlength">
        <FieldName>item_size</FieldName>
      </PSXPropertyFieldNameMapping>
    </PropertyMap>
  </PSXWebdavContentType>

  <!--
  A list of excluded folder properties when creating a folder in
WebDAV.
  A created folder will inherit all properties of its parent folder
except
  the list of excluded properties and the folder name.
-->
  <ExcludeFolderProperties>
    <PropertyName>sys_pubFileName</PropertyName>
  </ExcludeFolderProperties>
</PSXWebdavConfigDef>

```



## APPENDIX II

---

## WebDAV Methods

Supported WebDAV Methods:

<b>WebDAV Method</b>	<b>Rhythmyx Processing</b>
OPTIONS	Requests the allowed WebDAV methods for the resource identified by the Request-URI.
GET	Retrieves the content of the resource (or item) identified by the Request-URI. If the resource is a folder, then retrieves all properties of its immediate children.
HEAD	Retrieves the header information as in the GET method, but not the content.
PROPFIND	Retrieves properties defined on the resource identified by the Request-URI.
PROPPATCH	Processes instructions specified in the request body to set properties defined on the resource identified by the Request-URI. Removing properties is not supported.
MKCOL	Creates a new collection or folder.
PUT	Creates a new item or updates the content of an existing item. A PUT cannot be performed on a collection (or folder) according to the WebDAV specification.
DELETE	Deletes an item or a folder identified by the Request-URI. Deletes a folder will also recursively delete all its child items and sub-folders if exists.
COPY	Copies items and folders from one location to another. Supporting “0” and “infinity” depth for copying folders.
MOVE	Moves items and folders from. Move for properties is not supported.
LOCK	Checks out the resource identified by the Request-URI.
UNLOCK	Checks in the resource identified by the Request-URI.



---

# Index

## A

- Adding Additional WebDAV Servlets in Rhythmyx's Installation of Tomcat • 12, 16
- Appendices • 51
- Architecture of WebDAV Components in Rhythmyx • 4

## D

- Default Rhythmyx WebDAV Configuration File • 53
- Deploying the Rhythmyx WebDAV Servlet in BEA WebLogic • 6, 29
- Deploying the Rhythmyx WebDAV Servlet in IBM WebSphere • 6, 33
- Deploying the Rhythmyx WebDAV Servlet in Rhythmyx Express Portal • 6, 37

## E

- Editing a WebDAV Configuration File • 5, 13, 17, 37

## I

- Implementing WebDAV • 9
- Implementing WebDAV in Rhythmyx's Installation of Tomcat • 6, 11, 29, 33
- Introduction • 3

## O

- Opening Web Folders in Internet Explorer • 15, 32, 36, 40, 44, 45
- Opening Web Folders in Mac OS X • 47
- Opening Web Folders in Windows Explorer • 15, 28, 32, 36, 40, 44
- Options for Implementing WebDAV Servlets • 10

## P

- Prerequisites and Recommendations • 5

## S

- Simple Example Implementation of WebDAV • 11, 19, 37

- Supported WebDAV Clients and Application Servers • 7

## T

- Troubleshooting WebDAV • 41

## W

- WebDAV and Rhythmyx Workflow • 49
- WebDAV Methods • 57
- WebDAV Web Applications • 6
- Working with Rhythmyx Content in Web Folders • 48
- Working with WebDAV • 43