# FilePanel WebApp

Setup

The FilePanel WebApp is delivered as a standalone WAR archive that can be deployed in any Java servlet container. Separate versions are published for JDK 1.5 and JDK 1.6, and each of these have been tested under WebLogic Server and Apache Tomcat.

The detailed functionality of FilePanel is described in later sections of this document, however at a high-level it enables users to view application logs and config files through their web browsers. Large files are accessed in chunks via Ajax callbacks as a user scrolls through the file. Optionally the capability to edit remote files can also be enabled.

Setup involves exploding the WAR file and editing the following sections of WEB-INF/faces-config.xml to specify the comma-separated locations of your log and config files:

  <managed-bean>

    <managed-bean-name>filepanelDirView</managed-bean-name>

    <managed-bean-class>com.qwest.mdw.web.filepanel.DirView</managed-bean-class>

    <managed-bean-scope>session</managed-bean-scope>

    <managed-property>

      <property-name>rootDirectories</property-name>

      <value>./logs,/tmp/argos/log,~/.argos/log</value>

    </managed-property>

    <managed-property>

      <property-name>excludePatterns</property-name>

      <value>\*.zip,\*.gz</value>

    </managed-property>

  </managed-bean>

  <managed-bean>

    <managed-bean-name>propertyDirView</managed-bean-name>

    <managed-bean-class>com.qwest.mdw.web.filepanel.DirViewNoAuth</managed-bean-class>

    <managed-bean-scope>session</managed-bean-scope>

    <managed-property>

      <property-name>rootDirectories</property-name>

      <value>./Qwest/config</value>

    </managed-property>

    <managed-property>

      <property-name>editable</property-name>

      <value>false</value>

    </managed-property>

</managed-bean>

It's best if the locations are referenced in a relative manner so that a single version of the WAR can be deployed in multiple environments (e.g. Dev, Test, Prod). The specified directories will be traversed recursively and displayed in a tree view in the webapp.

Editing of the remote files can be enabled by setting the "editable" managed-property. This is a very coarse-grained enablement which allows or disallows editing at a global level for all users. You can provide more specific enablement by setting "editable" to true and then implementing a custom DirView extension class instead of the DirViewNoAuth managed-bean specified in the default faces-config.xml file. Your custom implementation might look something like the following and can be located in the WEB-INF/classes folder of the deployed WAR archive.

import java.io.File;

import com.qwest.mdw.web.filepanel.DirView;

public class DirViewNoAuth extends DirView

{

@Override

protected boolean userCanEdit(File directory, File file)

{

boolean myCustomEnablement = true;

// isEditable() reflects the value in faces-config

return isEditable() && myCustomEnablement;

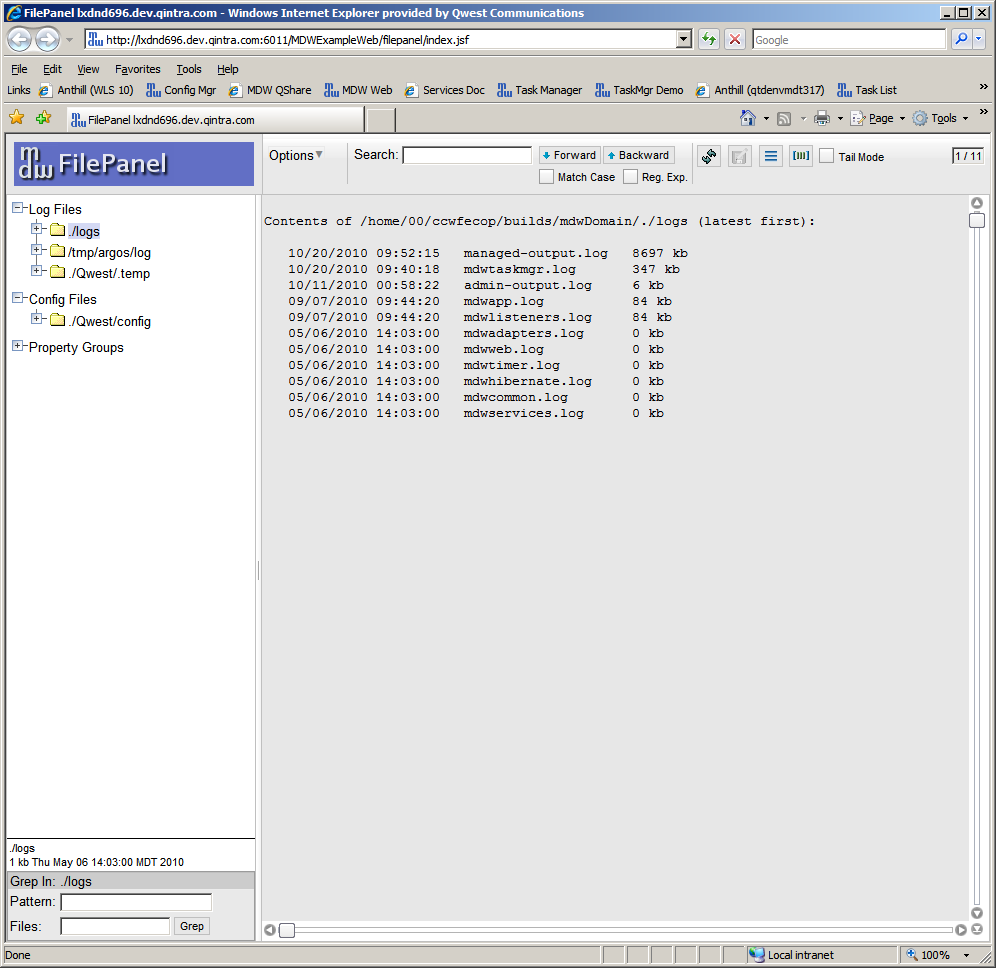
}

}

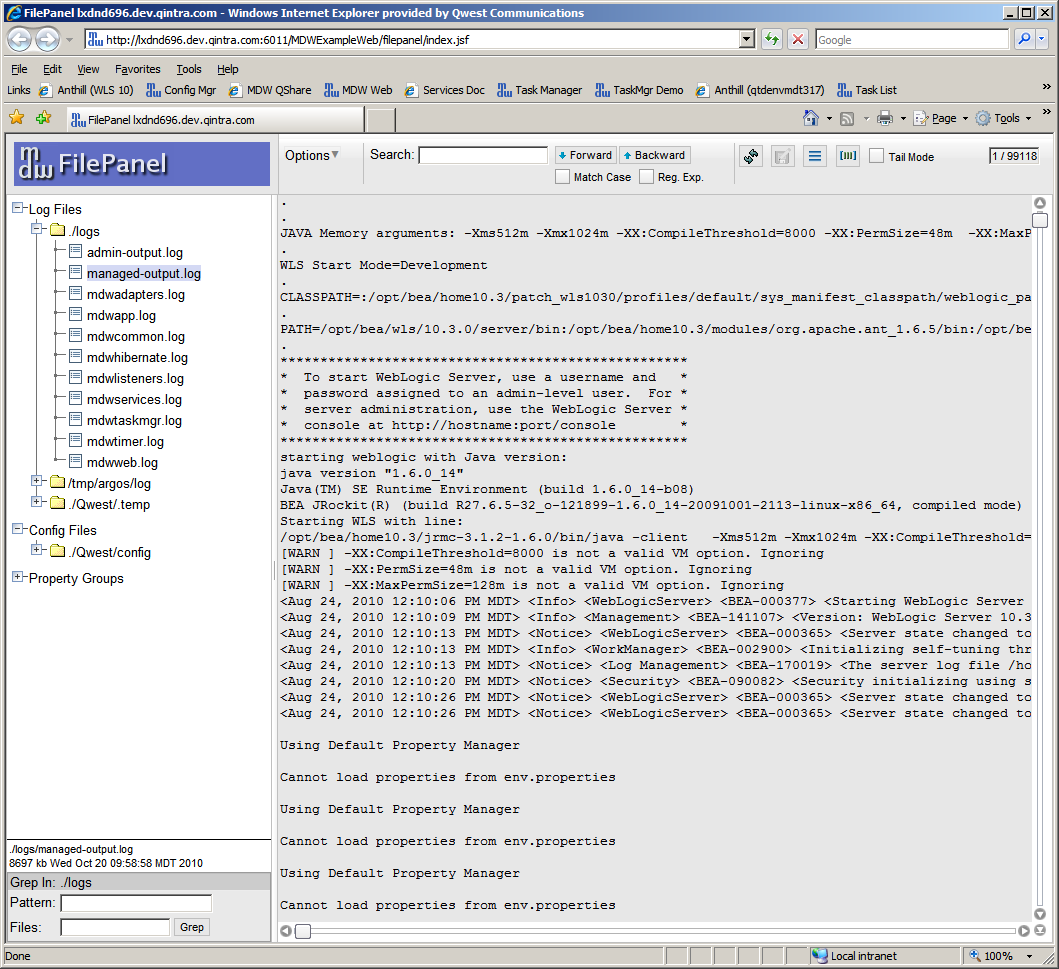
The remaining sections of this document describe FilePanel usage in greater detail with accompanying screenshots.

Log Viewer

Authenticated users can view the server logs located in specified directories on the file system. Clicking on the directory name displays of files, sorted according to the latest write timestamp.



Expand the tree view and select a file to view its contents. Note, only a portion of the file is loaded at any given time. Ajax requests are utilized to refresh this buffer when scrolling. The buffer size can be adjusted through the menu Options > Buffer Lines.



Files can be searched individually, or a grep operation can be performed by selecting a directory and entering a search pattern and a filename pattern (\* for all files in the directory, or \*.log for all log files, etc.).

Config Files

Configuration files can be viewed in the same manner as described above for log files. Users with appropriate permissions can also make changes to configuration settings by clicking on the Edit icon in the toolbar.

