Compoze Portlets

for Microsoft Exchange
version 2.0

Setup Guide

cp20mexsg/0001

Source code and copyright information

This software package contains source code. All source code and related information is subject to the following notice:

Trademarks

Contact information

Compoze Software, Inc.
1100 East Hector Street, Suite 457
Conshohocken, PA 19428

Telephone: Toll-free US Only - 1.866.COMPOZE (1.866.266.7693)
Telephone: 1.610.862.1104
Fax: 1.610.862.1180

Web Site: www.compoze.com
Information: info@compoze.com
Sales: sales@compoze.com
Support: support@compoze.com

Table of Contents

About This Document

This guide is intended for software developers or server administrators who are building Web portal applications. A basic understanding of BEA WebLogic Platform, Microsoft Exchange, Java, and JavaServer Pages (JSP) are assumed.

Prerequisite Software

Setting up the Compoze Portlets assumes the following prerequisite software:

- BEA WebLogic Platform 8.1
- Microsoft Exchange 5.5 with Service Pack 3 or higher

  or
- Microsoft Exchange 2000 with Service Pack 1 or higher

Documentation Syntax Conventions

The Windows convention of "\" as a path separator is used wherever necessary. UNIX users and users of other operating systems should translate these paths and variables accordingly.

Also, since system software and configurations can vary from one system to another, portions of the command syntax displayed in this document may include sample parameters or variables that represent the actual command syntax you would need to enter. These entries

are indicated by parameters in uppercase placed between percent signs (`%PARAMETER%`), and include, but are not limited to the following:

Parameter	Definition
`%COMPOZE_HOME%`	The complete directory specification for Compoze Software products. For example: `c:\Program Files\compoze`
`%COMPOZE_PORTLETS_HOME%`	The complete directory specification for the Compoze Portlets. For example: `c:\Program Files\compoze\compoze_portlets_bea-2.0`
`%JAVA_HOME%`	The complete directory specification for the Java Development Kit. For example: `d:\bea\jdk141_03`
`%WL_PLATFORM_HOME%`	The complete directory specification for the BEA WebLogic Platform. For example: `d:\bea\weblogic81`
`%SAMPLE_PORTAL_HOME%`	The complete directory specification for the Samples Portal. For example: `d:\bea\weblogic81\samples\portal\portalApp`

Compoze Portlets for MS Exchange

The Compoze Portlets installer copies Compoze Portlets files into the appropriate folder under `%SAMPLE_PORTAL_HOME%`. This Setup Guide walks you through the steps you need to perform to complete the Compoze Portlets for use with Microsoft Exchange setup on the "Avitek" Sample Portal.

Chapters in this Guide include:

Note: While you can install the portlets in any portal of your own, this Guide shows you how to set up the Compoze Portlets inside the "Avitek" Sample Portal delivered with the WebLogic Portal. When you have finished all of the setup instructions in each of the above chapters, open the compoze_portlets_users_guide.pdf file located at `%COMPOZE_PORTLETS_HOME%\docs` for information about using each of the Compoze Portlets.

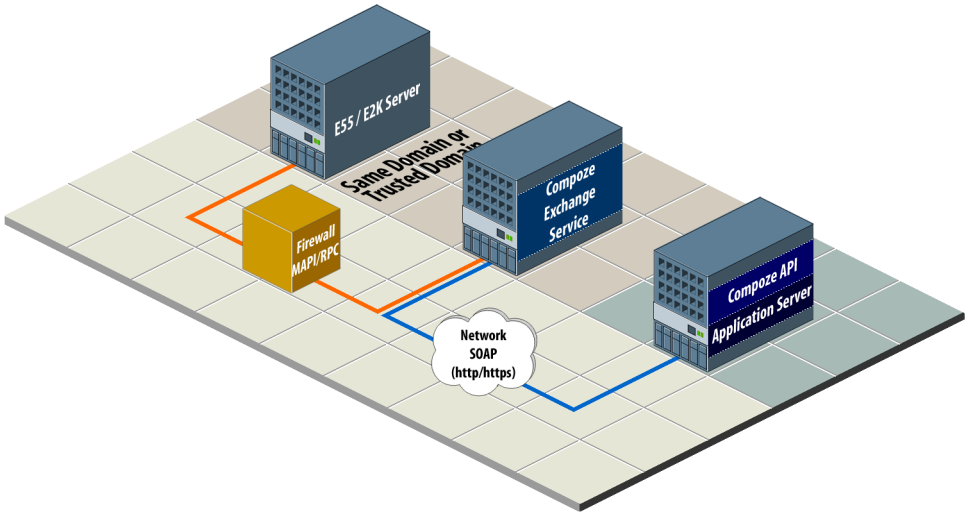
Compoze Exchange Service

This chapter describes the Compoze Exchange Service architecture, installation and configuration and supported platforms.

Architecture

With the Compoze Portlets, Compoze has implemented a new native Windows service which exposes Exchange 5.5 and later functionality to Java programs in a way that is easier to use and offers better performance than Java-to-COM CDO bridging solutions. The following benefits result from the Compoze Portlets architecture:

1. There is no longer a requirement to supply an administrative user on the intermediary server (which used to be the DCOM server, and is now the Compoze Exchange Service). Users can log in and authenticate to their mailbox using their own credentials.
2. Better performance and scalability is achieved by sending coarse grained requests between the Java application and the Compoze Exchange service. Previously, DCOM RPC communications requiring a low latency connection were used. In addition, there is no Java-to-COM bridging required on the client side.
3. The firewall requirements have been reduced from requiring MSRPC to only requiring HTTP over port 80 (or a selected port). HTTPS is also supported.
4. New features are available due to the ability to use low-level MAPI calls on the intermediary server. Rich text is now supported.



The Compoze Exchange Service acts as an intermediary between a Java client application and Microsoft Exchange. The Java application could be a web application, desktop application or web framework such as a portal server. Essentially, any Java client application running on a JVM version 1.3 or greater is supported.

The means the protocol that the Java client uses to connect to the Compoze Exchange Service is hidden in the implementation of the API. Compoze can then replace the APIs connectivity to the service without the need for API users to change their application or understand how the protocol is implemented.

SOAP Interface

Currently, the connectivity between Java and the Compoze Exchange Service is implemented using SOAP (Simple Object Access Protocol) version 1.1. SOAP provides a standard means of XML based messaging (including RPC, or Remote Procedure Call style messaging) that can be used across platforms. Since SOAP is used, this leaves open the possibility that the Compoze Exchange Service can be exposed to languages and programming environments other than Java.

On the Compoze Exchange Service side, the Microsoft SOAP Toolkit v3.0 is used. This runs in IIS on the Windows machine, and handles the SOAP communication with the client. A component written by Compoze handles requests from the client and sends them to another process on the same machine using local RPC. This process isolation insures that complicated Exchange interaction is not occurring in-process with IIS.

On the Java client side, a modified version of the Jakarta Axis 1.0 library is used. Java classes have been written to wrap the SOAP responses and provide the information to the programmer in a usable fashion.

Connectivity to Exchange

Connectivity to Microsoft Exchange is implemented in a Windows service called the Compoze Exchange Service. It runs as a standalone process and exposes its functionality via local RPC to other processes in the system (including IIS). When requests come in, they are given a thread out of a thread pool that grows and shrinks automatically as necessary. Sessions that have been open for longer than the time-out period are automatically closed to avoid holding resources open unnecessarily.

The Compoze Exchange Service is implemented using a combination of MAPI (Messaging Application Programming Interface) and CDO (Collaboration Data Objects). The details of this do not need to be understood by the application programmer, but this is the reason that CDO and MAPI from an Exchange Server CD is a prerequisite of the installation. This is where CDO and the MAPI provider for Exchange are obtained.

Network/Firewall Requirements

The Compoze Exchange Service must be located either on the **same domain** as the Exchange server or a **trusted domain**. Additionally, it must be configured to allow logins from the Exchange users who will be accessing their mailboxes. This requirement exists because the user logging on to Exchange actually performs a login to the Compoze Exchange Service machine in order to obtain a primary access token.

NOTE: A domain controller would not be a suitable machine for the Compoze Exchange Service because by default they do not allow interactive logins by domain users. It is possible to put the Compoze Exchange Service on the Exchange server itself, but be aware of the additional processor and memory burden that will be placed on Exchange.

HTTP or HTTPS traffic must be able to pass between the Java client and the Compoze Exchange Service. Traversing an HTTP proxy is OK as long as it is able to pass the POST requests used by the SOAP protocol. Although a high bandwidth, low-latency connection will improve performance, the protocol has been designed to reduce the number of round trips made on the network. Therefore, packet round trip times of 50-100ms should be tolerable for the application. The amount of bandwidth required will depend on the number of users simultaneously using the application. Each user may consume roughly 1K/sec. on average, with this number increasingly dramatically if users do a lot of work with large file attachments.

MSRPC traffic must be able to pass between the Compoze Exchange Service and Microsoft Exchange. MSRPC requires TCP port 135 to be open plus a range of ports above 1024. The network connection for this MSRPC traffic must have a low latency (less than 10 milliseconds, and preferably a 100 megabit LAN with less then 1 millisecond response times). Round trips are made over the network for each MAPI RPC, therefore the Compoze Exchange Service machine must be located as close as possible to Exchange on the network.

Installation and Configuration

Prerequisites

The Compoze Exchange Service must be installed on a clean Windows 2000 (with Service Pack 3 or higher) or Windows Server 2003 with IIS. Windows 2000 Professional may be used for testing purposes, but it only allows 100 simultaneous socket connections at a time, a limit that you will likely run into in production. If you are using a machine which already has other software installed on it, proceed at your own risk, however, it MUST not have Microsoft Outlook on it or any other product which installs a different version of CDO.

CDO and the MAPI Exchange Service provider are a required prerequisite of the installation. They may only be obtained using one of the two methods below. You may not install CDO using any version of Microsoft Outlook. If you do, it may initially work but Microsoft does not support use the dlls that ship with Outlook in server applications, so you may run into issues under high server load.

The MAPI Exchange Service Provider and CDO (Collaboration Data Objects) may be installed in one of two ways:

Using the Exchange 5.5 Server CD

1. Insert the Microsoft Exchange 5.5 Server CD. If you are running Exchange 2000, this CD also ships with Exchange 2000 distributions.
2. Choose "Setup Server and Components".
3. Choose "Microsoft Exchange Server 5.5".
4. Choose Complete / Custom.
5. Choose "Microsoft Exchange Administrator" only install (uncheck all other boxes).
6. Enter the license key.
7. Install Exchange 5.5 Service Pack 4 ((http://www.microsoft.com/exchange/downloads/55/sp4.asp). This should upgrade `cdo.dll` to version 5.5.2653.12 (may be checked by right clicking the dll in Explorer and going to the Version tab).
8. Install the proper CDO patch (available from http://support.microsoft.com/default.aspx?scid=kb;en-us;818709). This should upgrade `cdo.dll` to version 5.5.2657.55 or later (may be checked by right clicking the dll in Explorer and going to the Version tab). **Warning: If you do not perform this step then CDO and the Compoze Exchange Service will crash when accessing mailboxes that have been used by Outlook 2003**.

Using the CDO installed with an Exchange server

CDO 1.2.1 is installed with all versions of Exchange 5.5, Exchange 2000 and Exchange 2003. However, in some cases the `cdo.dll` that gets installed may not be registered during the installation. To be sure that `cdo.dll` is registered on your Exchange server, perform the following steps after Exchange has been installed:

1. Open a command prompt
2. cd to `exchsrvr\bin` (depends on where you have Exchange installed)
3. run "`regsvr32.exe cdo.dll`" to register CDO

Once CDO is registered, it is a good idea to upgrade your Exchange server to the latest service pack so that the latest CDO is included. With Exchange 5.5 only, follow step 8 from Method #1 to obtain the latest CDO patches. After the installation, you may stop all Microsoft Exchange related services in Administrative Tools > Services so that you do not actually use the server itself.

Running the Installer

After running the main installer, an executable called `setup_service.exe` is placed in `%COMPOZE_PORTLETS_HOME%\8.1\service`. Run this executable on the Compoze Exchange Service machine. During the installation, you will be prompted to specify the location of two important directories:

- The MAPI profile directory: This is a directory where temporary MAPI profiles are stored. The profiles are small and are deleted when they are no longer in use so it is safe to choose a directory on a volume with a couple hundred megs free. This setting will be reflected in the registry key `HKEY_LOCAL_MACHINE\Software\Microsoft\Windows Messaging Subsystem\ProfileDirectory`.
- The attachments directory: This is a directory where attachments are stored temporarily while they are transferred from IIS to the Compoze Exchange Service. It is also critical that the permissions on the attachments directory be set so that all users have read/write access to it. The attachment files are deleted when they are no longer in use, however attachments can be large, and many users can be creating them at any given time. Therefore, it is good practice to point to an attachments directory on a volume with 1 gigabyte or more free (depending on how much load your Compoze Exchange Service will see).

Assuming you have installed the prerequisites, the installation will install the Compoze Exchange Service and register the SOAP component with IIS.

Production License Key

The Compoze Exchange Service only allows up to 20 concurrent connections when running in trial mode. Production keys come with a `service-key` field which unlocks the service for an unlimited number of users. The service key is located inside the license.xml file and must To use the Compoze Exchange Service Configuration dialog, you must run the `Enter Production Key` shortcut installed in the `Compoze Exchange Connector` program group.

Enter the `licensee` and `service-key` fields EXACTLY as they appear in your `license.xml` file. Or, you may browse for your `license.xml` file and the values will be read from it directly. Click OK and restart the service. There should be a message in the Application Log of the Windows Event Viewer every time you start up the service that says “Production license key verified.”

Note: The `license.xml` file must be obtained from Compoze Software.

Verifying the Installation

After the Compoze Exchange Service installation, you can verify that it was installed properly by verifying the following:

1. You should have a directory tree located in `c:\Program Files\compoze\czex` (or wherever you chose to install the software).
2. There should be a new service in `Administrative Tools > Services called Compoze Exchange Service`. It should be started.
3. There should be a new IIS virtual root in `Administrative Tools > Internet Services Manager` called `czexsoap`. In this virtual root should be two files: `CzExSoap.WSDL` and `CzExSoap.wsml`.

Testing the Service and Connection

After installation, it is highly recommended that you verify that the service has been set up correctly. There are two ways to verify that you can open a connection to the service and that the service can connect to Exchange:

1. Use the Swing based graphical wizard.
2. Use the command line exerciser.

If a graphical environment is available, using the wizard is recommended because it provides online help and resolutions to various common problems that can occur. The command line exerciser is available for text only environments and for performing additional advanced tests.

Running the Wizard from the Shortcut

Simply run the Start Connection Wizard shortcut from `Programs > Compoze Software > Compoze Portlets - BEA WebLogic Edition > Start Exchange Connection Wizard`. For more information, see “Using the Wizard” on page 2-8.

Running the Wizard Manually

1. Add the following jar files to your `CLASSPATH`:
   - `harmony_portlets.jar` found at `%COMPOZE_PORTLETS_HOME%\lib\`
   - `weblogic.jar` found at `%WL_PLATFORM_HOME%\server\lib\`

- `p13n_system.jar` found at `%WL_PLATFORM_HOME%\p13n\lib\`

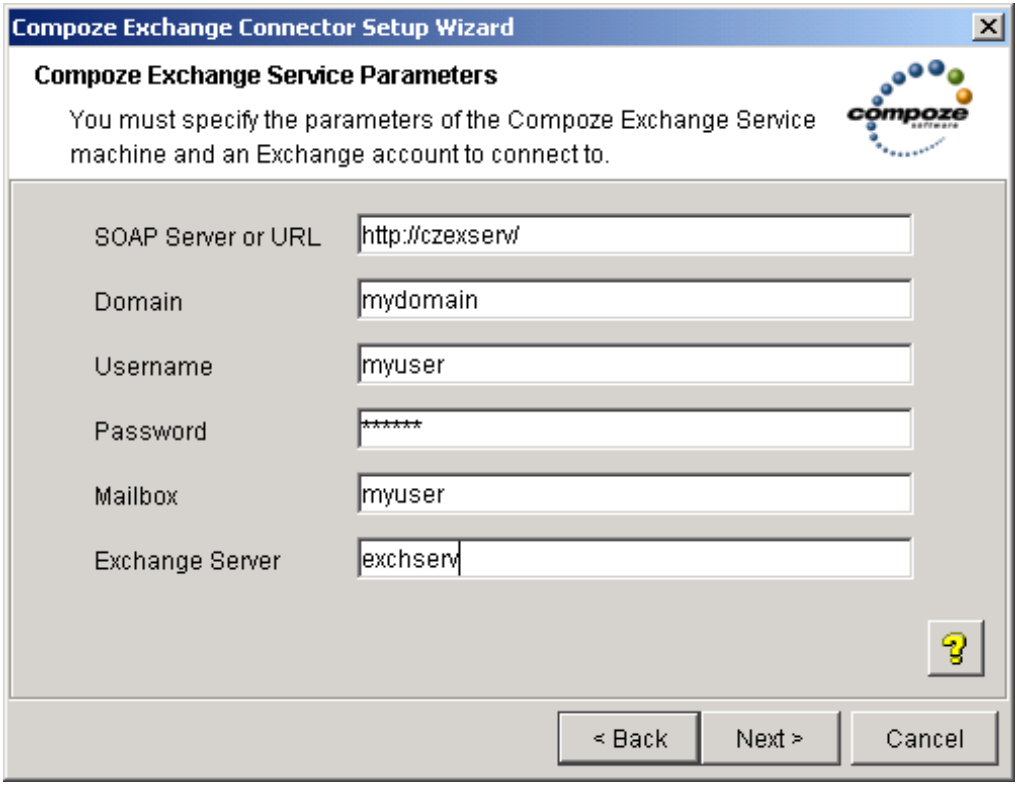
2. Run the exerciser with the following command:

```
java com.compoze.exchange.sessiontest.SessionTest
```

3. See the section on *Using the Wizard* for information on how to use the wizard.

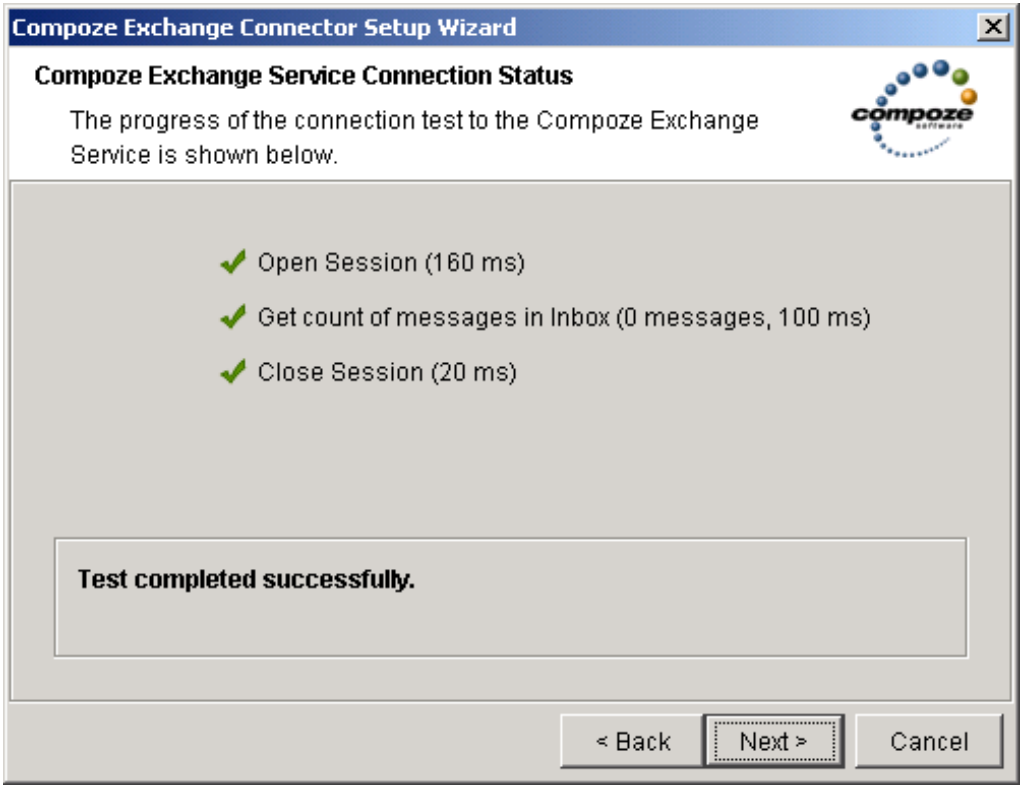
Using the Wizard

1. Fill in the connection parameters. A help dialog is available by clicking the question mark icon. It is highly recommended that you read this fully to understand what each of the connection parameters is.



2. The wizard will open a session to the service, check the number of messages in the Inbox, then close the session. A green checkmark will be placed next to operations that succeed. If an operation fails, a dialog will display more information and attempt to offer

you a resolution for the problem. If you find a problem that you cannot resolve, contact support@compoze.com with the full error report for more help.



NOTE: Opening a session for the first time can take a considerable amount of time as the Java classes are loaded on the Java side and dll files are loaded in the Compoze Exchange Service. Once the test has completed you may wish to click *Back* then *Forward* to open a session again. This will give you a better idea of how long this typically takes in your environment.

Running the Command Line Exerciser Manually

1. Add the following jar files to your `CLASSPATH`:
   - `harmony_portlets.jar` found at `%COMPOZE_PORTLETS_HOME%\lib\`
   - `weblogic.jar` found at `%WL_PLATFORM_HOME%\server\lib\`

- `p13n_system.jar` found at `%WL_PLATFORM_HOME%\p13n\lib\`

2. Run the exerciser with the following command:

```
java com.compoze.exchange.Exerciser
```

3. For more information, see "Using the Command Line Exerciser" on page 2-10.

Using the Command Line Exerciser

Once the exerciser is running, type `open` and hit enter to be prompted for the following parameters:

Field	Description
Domain	the domain of the user that will access the Exchange mailbox
Username	the username of the user that will access the Exchange mailbox
Password	the password of the user that will access the Exchange mailbox
Exchange Server	the name of the Exchange Server as it can be resolved from the Compoze Exchange Service machine (may be omitted to test GAL lookups, but these must first be configured in the service. For more information, see "Global Address List Lookups" on page 2-14.
Exchange Mailbox	the name of a mailbox to connect to (may be omitted to test GAL lookups, but these must first be configured in the service. For more information, see "Global Address List Lookups" on page 2-14.
Service URL (enter=default)	the hostname of the Compoze Exchange Service machine as it can be resolved from the machine you are running this exerciser on

If everything goes correctly, you should see something like the following:

```
Session ID: CDO_support_00000710_00000003
Request/response time: 170 ms
```

If the connection fails you will see an exception with more error information to help you determine what went wrong. Help on further commands in the exerciser is available by typing the command `help`. Help may be obtained for a specific command by typing `help <command>`.

Administration

Service Configuration and Registry Entries

The following registry entries control the behavior of the Compoze Exchange Service. All of the following registry keys are located in `HKEY_LOCAL_MACHINE\Software\Compoze Software, Inc.\czex`. If any of these registry entries are not created, you must first create them.

Key	Description	Change Requires Service Restart?
AttachmentDirectory (REG_SZ)	The full path to the directory where attachments are temporarily stored while sending from IIS to the Compoze Exchange Service. The directory should be on a volume with sufficient disk space to store attachments while in transit. The exact amount of disk requirements will depend on the server load but choose a volume with at least a gigabyte of free space. If the directory does not already exist, then the Compoze Exchange Service will create it.	No
BaseLib (REG_SZ)	This is the base Java `CLASSPATH` information for notifications. Do NOT modify this information. Instead, use ExtraLib	-
ClientVersionCheck (REG_DWORD)	If this entry exists and is set to 1 (the default), verify that the client version is the same as the service, otherwise do not allow sessions to be opened. NOTE: if you do not perform verification that the versions are the same, you may run into compatibility issues between your Java code and the service.	Yes
ExtraLib (REG_SZ)	This is the extra Java `CLASSPATH` information for modifications. Add any additional JAR files or directories to this entry.	Yes
GalDomain (REG_SZ)	the domain of the fixed GAL user (required for lookups in Exchange 2000 and above)	No

GalMailbox	the mailbox of the fixed GAL user (required for lookups to work in any Exchange version)	No
GalPassword	the password of the fixed GAL user (required for lookups in Exchange 2000 and above)	No
GalServer	the fixed Exchange server for the user to connect to (required for lookups in Exchange 2000 and above)	No
GalUsername	the username of the fixed GAL user (required for lookups in Exchange 2000 and above)	No
INotificationListener (REG_SZ)	A list of concrete classes that implement the `com.compoze.exchange.INotificationListener` interface. These classes will receive Notification Events.	Yes
InstallDir (REG_SZ)	The installation directory of the Compoze Exchange Service. For example: `c:\Program Files\compoze\czex`	No
ProfilingEnabled (REG_DWORD)	If set to 1, log timing information for methods called in the Compoze Exchange Service to the event viewer. If set to 0 or if the key does not exist, do not log timing info.	Yes
SessionTimeoutSecs (REG_DWORD)	If a session is idle for this number of seconds, it will be closed. This prevents the Compoze Exchange Service from having sessions open in memory that are no longer in use. The default if the registry entry is not specified is 1800 seconds, or 30 minutes. The value of this setting will be re-read in no longer than one minute, so if it is changed from a large value to a small one for testing purposes, be sure to wait a minute for the change to take affect.	No

SoapEncryptionKey (REG_SZ)	If this parameter is supplied, the user password is encrypted with this key, even if SSL is not being used. A default key is chosen if it is not set. Be sure to supply the corresponding `PROP_SOAP_ENCRYPTION_KEY` property when instantiating the Session object on the client side, otherwise user logins will fail.	No
VMOptions (REG_SZ)	Specify any Java Virtual Machine options.	Yes

The following registry entries control the behavior of the Microsoft SOAP Toolkit (and thus affect the Compoze Exchange Service). All of the following registry keys are located in `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\MSSOAP\30\SOAPISAP`.

Key	Description	Change Requires IIS Restart?
MaxPostSize (REG_DWORD)	The maximum size that a client may post to IIS, in bytes. **NOTE:** this is not the maximum attachment size. The maximum attachment size is actually 4K or so less. This setting affects the size of any post, which could restrict the size of a large message body as well.	Yes
NoNagling (REG_DWORD)	If set to 1, disable the TCP Nagle algorithm. Nagling delays the sending of information until a certain amount of data is obtained. It is safe to leave this set to the default value of 0.	Yes
NumThreads (REG_DWORD)	The number of threads in the IIS pool available to handle requests. Any requests after the maximum pool size has been reached will be queued. The default is 3. In production, Compoze suggests a value that is 4 times the number of CPUs on the machine. For example, a dual CPU machine would have a NumThreads setting of 8.	Yes
ObjCachedPerThread (REG_DWORD)	This is the number of objects (in our case WSDL files) cached per thread. It is safe to leave this at the default setting of 1.	Yes

Global Address List Lookups

The Compoze Exchange Service can automatically look up the mailbox and/or home Exchange server for a user if they are not supplied when the session is opened. This is done by connecting to a fixed Exchange server and querying the Global Address List (GAL). The registry entries that control lookups begin with “Gal” and are in the previous section (Service Configuration and Registry Entries).

If Exchange 5.5 is being used, the connection to the GAL can be anonymous and only the Exchange server must be supplied. Exchange 2000 and above requires that a separate account and mailbox be available for the purpose of GAL queries. Note that for non-anonymous GAL connections the `LogonType` registry entry is used to determine what type of Windows impersonation is done in the service, so the GAL user must have sufficient rights to access the Compoze Exchange Service machine as defined by the rights in LogonType.

Proxy Servers

The Compoze Portlets supports connecting to a Compoze Exchange Service through an http proxy server. The following Java system properties may be set to control the proxy connection (notice that they are the same as the Java system properties that are used except they are prefixed by the string “czex”):

Property	**Description**
`czex.http.proxyHost`	the hostname of the proxy server
`czex.http.proxyPort`	the port of the proxy server
`czex.http.proxyUser`	the username to access the proxy server
`czex.http.proxyPassword`	the password to access the proxy server

Minimal IIS Lockdown Settings

The IIS Lockdown tool is supplied by Microsoft and allows you to protect your IIS server by denying access to certain HTTP methods, file extensions, etc. The following IIS Lockdown wizard walkthrough shows the minimum settings needed for the Compoze Exchange Service to run:

1. Select Server Template Screen: Select the template “Other (Server that does not match any of the listed roles)”
2. Internet Services Screen: Enable only “Web service (HTTP)” and click “remove unselected services” if you wish.
3. Script Maps Screen: Disable support for all selected script maps
4. Additional Security Screen: Remove all selected virtual directories from the server. Set file permissions to prevent anonymous users from performing these actions- include “Running system utilities” and “Writing to Content Directories”. Check “Disable Web Distributed Authoring and Versioning (WebDAV)”
5. URLScan Screen: Check box that says “Install URLScan filter on the server”
6. `urlscan.ini` file: With `AllowVerbs=1` set, the `AllowVerbs` section only needs to contain the “POST” verb. With `AllowExtensions=1` set, the `AllowExtensions` section only needs to contain the extension “.wsdl”.

Minimal Java Security Policy

The following section contains a listing of the minimal Java security policies needed to run Compoze Portlets.

```
// permissions required for exchange.jar
grant codeBase "file:/path/to/harmony_portlets.jar" {

// read access to the czex-client-config.wsdd file (AXIS)
permission java.io.FilePermission "czex-client-config.wsdd", "read,write"; // axis

// read access to the czex-server-config.wsdd file (AXIS)
permission java.io.FilePermission "czex-server-config.wsdd", "read,write"; // axis

// read access to the user's .czexrc file (exerciser only)
permission java.io.FilePermission "${user.home}${/}.czexrc", "read"; // exerciser

// required system properties
permission java.util.PropertyPermission "user.home", "read"; // exerciser
permission java.util.PropertyPermission "com.compoze.*", "read,write"; // Compoze
  properties
permission java.util.PropertyPermission "axis.*", "read"; // axis
permission java.util.PropertyPermission "java.protocol.handler.pkgs",
  "read,write"; // axis
permission java.util.PropertyPermission "czex.*", "read"; // proxy settings

// Compoze and AXIS runtime code permissions
permission java.lang.RuntimePermission "getClassLoader"; // licensing
permission java.lang.RuntimePermission "accessDeclaredMembers"; // axis
```

```
// permission required for making SOAP connection
permission java.net.SocketPermission "*:80", "connect";

};

// required system properties
permission java.util.PropertyPermission "com.compoze.*", "read,write"; // Compoze
  properties

};
```

Logging Information

Logging information for the Compoze Exchange Service and the Microsoft SOAP Toolkit is sent to the Windows Event Viewer. In the Application Log you will find information about the status of the service (during startup and shutdown), and warnings or errors if something goes wrong. The default Application Log size may be too small for your application. A minimum log size of around 5MB is recommended. In addition, the setting "Overwrite as Needed" is recommended to insure that no recent events are lost.

Should the Compoze Exchange Service stop unexpectedly, an error is printed to the Event Viewer. The information in this error contains everything that Compoze needs to investigate the situation further, so the Event Viewer logs should be considered a vital part of the support and monitoring process.

Maintaining Service Uptime

Although Compoze makes every effort to achieve 100% uptime for the Compoze Exchange Service, it is written in native code, uses external Windows libraries and is thus not fully immune to crashes.

Windows offers you the ability to enable crash recovery for the service so that you do not have to monitor and restart it in the event of a crash. To enable this, go to `Control Panel > Administrative Tools > Services > Compoze Exchange Service` properties. Under the Recovery tab, set First Failure, Second Failure and Subsequent Failures to "Restart the Service". You may wish to set the "Restart Service After" time to 0 minutes, so that the service restarts immediately.

Now, if a failure occurs, the service will automatically restart. All existing sessions will be lost, forcing users to log back in again.

Reporting Crash Errors

If you find that the Compoze Exchange Service crashes repeatedly in your environment and you wish to pursue it with Compoze support, there are steps that you can take to help us assist you. Please provide as much of the following information as possible in a single zip file when submitting your request:

1. The `error-#.txt` and `error-#.dmp` files in the `czex/bin` directory off of the Compoze Home directory. These files are created when a service crash occurs, and are critical in order for Compoze to resolve a problem.
2. The contents of your Application Log in the Event Viewer. This can be obtained by running "eventvwr", right clicking on the Application Log and choosing "Save File As". Save the file in `.evt` format.
3. What operation was being performed at the time of the crash, and whose mailbox it occurred in. This may be difficult information to obtain, but if available it does assist us in the debugging process, particularly if you are able to force a crash and give us a series of steps to reproduce it.
4. Version information of each of the service components. In Windows explorer, right-click each of `czex_service.exe`, `czex_soap.dll`, `czex_util.dll, and cdo.dll` and select the `Version tab`. This will aid in matching the supplied debug information with the version installed on your system.

In some situations, Compoze may request that you reproduce the problem send a full crash dump (much larger than the mini dump). These are the steps to obtain a full crash dump:

1. Run `drwtsn32 -i` from a command prompt. You should see the message "Dr. Watson has been installed as the default debugger".
2. Run `drwtsn32` by itself to configure Dr. Watson. Check the box that says "Dump Symbol Table" and select the number of errors that you wish to save (the defaults are acceptable).

In the event that the Compoze Exchange Service crashes, the Compoze support team will ask for as much information as possible in order to resolve the issue. Please provide as much of the following information as possible in a single zip file when submitting your request:

1. The `drwtsn32.log` and `user.dmp` files created by Dr. Watson. The location of these files can be determined by running `drwtsn32` and looking at the settings for "Log File Path"

and “Crash Dump”. These files are critical in order for Compoze to know where a crash occurred.

2. The contents of your Application Log in the Event Viewer. This can be obtained by running “eventvwr”, right clicking on the Application Log and choosing “Save File As”. Save the file in `.evt` format.
3. What operation was being performed at the time of the crash, and whose mailbox it occurred in. This may be difficult information to obtain, but does assist us in the debugging process, particularly if you are able to force a crash and give us a series of steps to reproduce it.

Reporting Problems With Messages

You may find that a particular message or folder is causing you problems with the API or service. In this case, it is possible to export the original messages for import at Compoze in order to reproduce the problem. To do this:

1. Open Outlook 2000 or above to the account with the problem messages.
2. Choose `File > Import and Export`.
3. In the “Import and Export Wizard” screen, choose “Export to a file.”
4. In the “Export to a File” screen, choose “Personal Folder File (.pst)”.
5. In the “Export Personal Folders” screen, choose the folder that you wish to export. It is easiest to export an entire folder (such as the calendar folder), but you may also click the Filter button and restrict what is exported by date range, subject, attendees, created time, etc. Just make sure that the offending messages get included by the filter you have chosen.

   After exporting the PST file, please place this in a zip file along with a small readme.txt file that explains the problem with the message/s and any filter that was used for the export in step 5. Send this zip file to support@compoze.com.

Using the Java API

The JavaDoc API Documentation contains all of the information you will need to get started using the Compoze Exchange Service API. Browse to `%COMPOZE_PORTLETS_HOME%\docs\index.html` to get started. To get started, look at the documentation for `com.compoze.exchange.Session` to learn how to open a session to the Compoze Exchange Service.

Setting Up Portlets

After running the Compoze Portlet installer application, the Compoze Portlets are available to add to a portal. This chapter shows you how to configure the `CompozeExchangeProfile` user profile and to register the portlets inside the Avitek Portal. This is accomplished with the WebLogic Workshop Platform Edition Portal Designer.

Setting the CompozeExchangeProfile Properties

The Compoze Portlets installer created a BEA User Profile named `CompozeExchangeProfile`. The `CompozeExchangeProfile` holds the Microsoft Exchange server information (such as Exchange Server name) for each user.

User Profile Property Names and Descriptions

The following table list the property names along with a description for each property in the `CompozeExchangeProfile`. All of the values can be set or changed by a Portal Administrator in WebLogic Workshop or in the WebLogic Administration Portal. Additionally, all of the properties, except for the Compoze Exchange Service Server (`compoze.exchange.compoze_exchange_service_url`) can be set or changed by each portal user in the portlets themselves.

Note: The Compoze Exchange Service Server property must be set in WebLogic Workshop or in the WebLogic Administration Portal as it is not a value that can be set through the portlets.

Property Name	Description
`compoze.exchange.compoze_exchange_service_url`	The name of a server, URI, or a full URL to the Compoze Exchange service WSDL file. Examples are as follows: • `server` • `http://server/` • `http://server:port/` • `http://server:port/czexsoap/CzEx-Soap.WSDL`. HTTPS URLs may be used if JSSE is installed in the running virtual machine. **NOTE:** This value needs to be set by the Portal Administrator.
`compoze.exchange.username`	The Exchange user's username.
`compoze.exchange.password`	The Exchange user's password. This value is encrypted and should not be modified by hand.
`compoze.exchange.domain`	The Exchange domain name.
`compoze.exchange.server`	The Exchange server name or IP where the user mailbox exists.

Note: HTTPS URLs may be used if JSSE is installed in the running virtual machine of the BEA WebLogic server. This requires placing `jsse.jar` in the `CLASSPATH`.

Setting CompozeExchangeProfile Properties for a Portal User

You can set the `CompozeExchangeProfile` property values in WebLogic Workshop. The following show you how to set the required Compoze Exchange Service URL property:

1. Launch WebLogic Workshop
2. Open the `CompozeExchangeProfile` (under `portalApp/data/userprofiles/`).
3. Highlight the `compoze.exchange.compoze_exchange_service_url` property.
4. In the `Property Editor`, click on the little button next to the text box named `value(s)` to bring up `Enter the Property Value` window.
5. Enter the default value to be your Compoze Exchange Service URL.
6. Click `OK` and save the file.

Placing the Compoze Portlets in the Portal

You can drag and drop Compoze Portlets onto pages in the WebLogic Workshop. To do this perform the following steps:

1. Launch WebLogic Workshop
2. Open the portal file (sample.portal) and navigate to the page on which you want to place the portal on.
3. In the Data Palette window, drag the portlet you want (i.e., `Compoze Exchange Addressbook`) into the placeholder on the page.
4. Select the portlet and use the `Property Editor` window to edit the portlet properties if desired.
5. Save the portal file.
6. Close WebLogic Workshop and log into the portal. The portlets will now be available to the portal users. The first time a user accesses one of the Compoze Exchange Portlets, a wizard will be presented to them to guide them through the remaining necessary configuration.

Modifying the Compoze Portlet Look and Feel

The Compoze Portlets are built with custom Cascading Style Sheet (CSS) classes. This gives you the greatest flexibility in modifying the look and feel of the portlets. The CSS classes need to be copied into the existing `portlet.css` for each skin that is available to the Portal users. To do this, perform the following steps:

1. Open the `%SAMPLE_PORTAL_DOMAIN_HOME%/sampleportal/framework/skins/avitek/portlet.css` file.
2. Append the contents of the `%COMPOZE_PORTLETS_HOME%/portlets/compoze_portlet.css` file into the `portlet.css` file.
3. Save the portlet.css file.
4. The Compoze Portlets should now have the default look and feel while in the Avitek Sample Portal. You can modify Compoze Portlet CSS classes (`.compoze_portlet_*`) in the `portlet.css` file as you see fit.
5. **Note**: This must be done for **every** portal domain that contains the Compoze Portlets

Portlet Sign-on Configuration

This chapter describes to configure the portlet user profile for customized sign-on, Auto GAL lookup and Basic IIS Authentication.

Setting the CompozeConfigProfile Properties

The Compoze Portlets installer created a user profile named `CompozeConfigProfile`. The `CompozeConfigProfile` holds the configuration information for each user. To specify configurations globally for all users, the properties default values should be set before accessing the profile.

User Profile Property Names and Descriptions

The section lists the property names and descriptions found in the `CompozeConfigProfile`. All of the values can be set or changed by a Portal Administrator in BEA WebLogic Workshop or in the BEA WebLogic Administration Portal. Depending on configuration, the portal user will have the ability to choose whether login parameters are drawn from the user profile or session parameter values.

Note: These properties should initially all be set in WebLogic Workshop, assuming that the initial default configuration will apply to all users. Additionally, you cannot set `use_user_profile` and `use_session_credentials` to `false`.

Property Name	Description
`use_session_credentials`	Set to `true` if the portlets should use the session variables stored in `http_session_var_username`, `http_session_var_password`, and `http_session_var_domain` as the logon parameters. Setting this to `true` also assumes Automatic GAL Lookup.
`use_user_profile`	Set to `true` if the portlets should use the `CompozeExchangeProfile` parameter values as logon values.
`http_session_var_username`	This property represents the name of the session parameter containing the username value.
`http_session_var_password`	This property represents the name of the session parameter containing the password value. The password value itself must be Base64 encoded.
`http_session_var_domain`	This property represents the name of the session parameter containing the domain value.
`enable_exchange_password_override`	This property represents whether the password field should be overridden. Set to `false` to display the field.
`enable_exchange_server_override`	This property represents whether the exchange server field should be overridden. Set `false` to display the field. **Note:** If this property is set to `true`, the Compoze Exchange Service settings must also be set. The user's username, password, and domain must still be specified for Auto GAL lookup.
`enable_exchange_alias_override`	This property represents the exchange alias field override. Set `false` to display the field.
`enable_domain_override`	This property represents the domain field override. Set `false` to display the field.

`enable_exchange_username_override`	This property represents the username field override. Set `false` to display the field.
`enable_account_settup_link`	This property represents whether the Account Setup link is displayed. Set to `true` to display the link. This setting will also have the same effect for the display of the portlet configuration wizard.
`use_bea_encoding`	This property represents whether the default Base64 (`com.compoze.util.StringUtility`) or the BEA Base64 (`weblogic.xerces.impl.dv.util.Base64`) class is used to encode/decode the user password. **Note:** if upgrading from a previous version, this should be set to `false`.
`http_authentication_username`	This property represents the IIS HTTP authentication username. (domain\username).
`http_authentication_password`	This property represents the IIS HTTP authentication password.

Use Cases

The following section describes possible Use Cases for sign-on and how to set the profile properties.

1. **"I want the user to be confronted with the full wizard the first time they sign-in, and also have full access to their Account Setup link."**

   The Administrator makes no changes after the default installation. Users see all wizard properties and account setup properties/fields. Users are able to change their profile settings.

`CompozeConfigProfile` settings:

Property Name	Setting
`use_session_credentials`	`false`
`use_user_profile`	`true`
`enable_exchange_password_override`	`false`
`enable_exchange_username_override`	`false`
`enable_exchange_domain_override`	`false`
`enable_exchange_alias_override`	`false`
`enable_exchange_server_override`	`false`
`enable_account_setup_link`	`true`

2. **"I want the user to be confronted with the full wizard and have full access to their Account Setup link, except for exchange alias (or another specified option)."**

   The user will not be able to set the Exchange Alias field, since it is not displayed in the wizard or the Account Setup in the portlets.

   `CompozeConfigProfile` settings are the same as Use Case #1 except the following:

Property Name	Setting
`enable_exchange_alias_override`	`true`

   Note: Each override property, overrides the display of that field to the user, this Use Case demonstrates this concept using the "Exchange Alias" field.

3. **"I want the user to be confronted with the wizard and allow their Account Setup link to be displayed, and enable Auto GAL lookup."**

   The "Exchange Server" field is not displayed to the user, which automatically enables Auto GAL Lookup (assuming the Compoze Exchange Service registry entries are configured correctly, refer to the Auto GAL Lookup section below for further details).

`CompozeConfigProfile` settings are the same as Use Case #1 except the following:

Property Name	Setting
`enable_exchange_server_override`	`true`

Using the Http Session to Retrieve Sign-On Credentials

The following Use Cases demonstrate the single sign on configuration, using session credentials to log onto Exchange via the portlets. To enable the use of Http session credentials the following parameters have to be set in the `CompozeConfigProfile`:

Property Name	Setting
`use_session_credentials`	`true`
`http_session_var_username`	`<the session variable containing the username>`
`http_session_var_password`	`<the session variable containing the password>`
`http_session_var_domain`	`<the session variable containing the domain>`

Note: this value is base64 encoded. Depending on the setting of the `use_bea_encoding` property, the Default or BEA Base64 encoding class is used. If you are upgrading from a previous version of the Compoze Portlets, do NOT use the BEA Base64 class (i.e. set `use_bea_encoding` to `false`).

Note: When using the session credentials, the portlets assume Auto GAL lookup has been configured. If Auto GAL lookup has not been configured, this Use Case will not function correctly.

4. **"I want the users to log in via the username, password, and Domain that are stored in the http session.**"

This will retrieve the username, password, and domain using the session parameter name set in the `http_session_var_username`, `http_session_var_password`, `http_session_var_domain`. The password value is assumed to be Base64 encoded via default (`com.compoze.util.StringUtility`) or BEA (`weblogic.xerces.impl.dv.util.Base64`) encoding. With these settings, the user will never be confronted with a wizard when initially logging into the portal (assuming all the provided logon information is correct).

`CompozeConfigProfile` settings:

Property Name	Setting
`http_session_var_domain`	`<the session variable containing the domain>`
`http_session_var_username`	`<the session variable containing the username>`
`http_session_var_password`	`<the session variable containing the password>`
`use_session_credentials`	`true`
`use_user_profile`	`false`

5. **"I want the users to log in via the username, password, and domain that are stored in session, and not have access to their Account Setup link."**

   This will act exactly as above however the user will be unable to use a custom profile to get to their account. They will not see an Account Setup link.

   - Set `enable_account_settup_link` to `false`. This will disable user's access to the wizard and Account Setup link.

     Note: This configuration is not recommended since Time Zone will not be set for the user. The Time Zone will default to the Exchange Server's Time Zone.

   - Set `enable_account_setup_link` to `true`. This allows access to the wizard and Account Setup link. Disable all configuration fields to display only the Time Zone.

     Note: this is the recommended configuration.

Set the following parameters:

Property Name	Setting
enable_exchange_username_override	true
enable_exchange_password_override	true
enable_domain_override	true
enable_exchange_server_override	true
enable_exchange_alias_override	true

6. **"I want the users to log in via the username, password, and domain that are stored in session, and be able to change their Exchange Alias only."**

   When the portlets have been configured to use session credentials, the portlets allow the exchange alias to be saved regardless of whether the use user profile checkbox has been checked. This allows users to log into the exchange portlets using either their session credentials, or using an exchange alias that this exchange user has permissions to.

`CompozeConfigProfile` settings:

Property Name	Setting
`use_session_credentials`	`true`
`use_user_profile`	`false`
`enable_account_setup_link`	`true`
`enable_exchange_username_override`	`true`
`enable_exchange_password_override`	`true`
`enable_domain_override`	`true`
`enable_exchange_server_override`	`true`
`enable_exchange_alias_override`	`false`

7. **"I want to use the SSO Login, by storing the values in the session but also still allow the user to modify his/her account settings if they choose to do so."**

   `CompozeConfigProfile` settings:

Property Name	Setting
`use_session_credentials`	`true`
`enable_account_setup_link`	`true`

For the user to successfully to modify his/her account settings and use the user profile instead of the session variables as log on the following parameters must also be set:

Property Name	Setting
`enable_exchange_username_override`	`false`
`enable_exchange_password_override`	`false`
`enable_exchange_domain_override`	`false (Note: optional if compoze.exchange.domain in CompozeExchangeProfile is set)`

8. **"I want to use BEA Base64 class for encoding/decoding of user passwords."**

   This Use Case provides the simple encoding solution. The `weblogic.xerces.impl.dv.util.Base64` should only be used if no previous versions of the Compoze Portlets have been used.

   `CompozeConfigProfile` settings:

Property Name	Setting
`use_bea_encoding`	`true`

9. **"I have anonymous log in turned off in IIS, what configuration do I need to change?"**

   The Compoze Exchange Service URL is protected with basic authentication. In this situation, one user would perform basic authentication (Base64 encoded text in header) to IIS for authentication, then the actual user would authenticate again.

   `CompozeConfigProfile` settings:

Property Name	Setting

http_authentication_username	<IIS authenticated username in the format "domain\username">
http_authentication_password	<IIS authenticated user's password>

Configuration

If the CompozeConfigProfile settings are to apply to all users, the default values for each parameter MUST be set before activating the profile. This is done by opening the CompozeConfigProfile.usr file in WebLogic Workshop and setting the default value for each parameter.

If you are using the session credentials for sign-on, we recommend that you set default values in the following parameters inside the CompozeConfigProfile http_session_var_username, http_session_var_password and http_session_var_domain. The CompozeExchangeProfile compoze.exchange.compoze_exchange_service_url property should also be set with a default as value.

The use_bea_encoding property allows selection between which Base64 class is used. If this property is set to true, BEA weblogic.xerces.impl.dv.util.Base64 is used. Otherwise, the default com.compoze.util.StringUtility class is used.

Note: If you are upgrading from a previous version of the Compoze Portlets set the use_bea_encoding to false. If this property is set to true and a previous version of the Compoze Portlets exist, no currently existing users will be able to log into the system without the administrator having to reconfigure their account.

Time Zone

We highly recommend that the Account Setup link is always left active as the user will be able to set his/her Time Zone here. If the user does not set his/her Time Zone via the wizard or Account Setup link, the Exchange Server's Time Zone will be assumed as the default Time Zone for this user.

Enabling the user profile when using session credentials (Wizard/Account Setup)

If the portlet configuration is set to use session credentials for log on, the administrator can still give users permission to customize their exchange account logon information once they are logged into the portal. The "Use this Account information during login" checkbox has to be checked for users to use their profile `(CompozeExchangeProfile)` and will appear only during certain configurations.

1. The checkbox will never be visible to the user when `use_session_credentials` is `false`, it is assumed that the user profile `(CompozeExchangeProfile)` parameters will be used for the account log-on information.
2. If `use_session_credentials` is `true`, the checkbox is displayed under the following conditions:

Property Name	Setting
`use_session_credentials`	`true`
`nable_exchange_username_override`	`false`
`enable_exchange_password_override`	`false`
`enable_exchange_domain_override`	`false (Note: if this is set to false, then http_session_var_domain must contain a value)`

Note: The checkbox appears when the user is able to modify his/her username, password and domain. If the "Exchange Domain" field is overridden (not visible) the portlets will attempt to pull the value from the session variable stored for domain. If this is also empty, the user will not be able to log on.

Auto GAL Lookup

For Automatic GAL look up to function, the following registry entries have to be set on the Compoze Exchange Service machine:

- GalServer

- GalDomain
- GalUsername
- GalPassword
- GalMailbox

To enable Auto GAL lookup in the portlets, set the `CompozeConfigProfile` property `enable_exchange_server_override` is `true`.

Error Messages

The section describes possible portlet error message displayed to the portal user.

1. **"Please contact your System administrator, the compoze CompozeConfigProfile settings need to be re configured by the System Administrator."**

   This indicates that `use_session_credentials` and `use_user_profile` have both been set to `false`.

2. **"Please contact your System administrator, your CompozeConfigProfile session values are currently not set or set incorrectly."**

   This means that `http_session_var_username`, `http_session_var_password` and `http_session_var_domain` have not been set or are set incorrectly when `use_session_credentials` is `true`.

Manual Setup

The Compoze Portlets installer automatically copies the portlet JSP files, portlet definition files, User Profile files, `contact_ejb.jar`, and the `harmony_portlets.jar` in place. These file are copied to directories under the `%WEBLOGIC81_HOME%\samples\portal\portal-App\sampleportal` directory. The installer also modifies the `web.xml` to add necessary servlet mappings for the Compoze Portlets.

The following procedure shows you how to do these steps manually. Adjust the domain name accordingly if different than the `sampleportal`.

Copying the Compoze Portlets into place

1. Navigate to `%COMPOZE_PORTLETS_HOME%\portlets` and copy the `compoze` folder and all its contents.
2. Paste the `compoze` folder into the following directory:
   `%WEBLOGIC81_HOME%\samples\portal\portalApp\sampleportal\portlets\`

Copying the .portlet files into place

1. Navigate to `%COMPOZE_PORTLETS_HOME%\portlets` and copy the `includes folder` and all its contents.
2. Paste the includes folder into the following directory:
   `%WEBLOGIC81_HOME%\samples\portal\portalApp\sampleportal\portlets\`

Copying the harmony_portlets.jar into place

1. Navigate to `%COMPOZE_PORTLETS_HOME%\lib` and copy the `harmony_portlets.jar` file.
2. Paste the `harmony_portlets.jar` file into the following directory:
   `%WEBLOGIC81_HOME%\samples\portal\portalApp\APP-INF\lib`

Copying the contact_ejb.jar into place

1. Navigate to `%COMPOZE_PORTLETS_HOME%\lib` and copy the `contact_ejb.jar` file.
2. Paste the `contact_ejb.jar` file into the following folder:
   `%WEBLOGIC81_HOME%\samples\portal\portalApp`

Copying the User profiles into place

1. Navigate to `%COMPOZE_PORTLETS_HOME%\userprofiles` and copy the `CompozeExchangeProfile.usr` file.
2. Paste the `CompozeExchangeProfile.usr` files into the following folder:
   `%WEBLOGIC81_HOME%\samples\portal\portalApp\META-INF\data\userprofiles`

Modifying the web.xml file

1. Navigate to `%WEBLOGIC81_HOME%\samples\portal\portalApp\sampleportal\WEB-INF\`
2. Open the `web.xml` file with a text editor and find the first section that starts with the following:

```
<!-- Compoze Servlet -->
```

3. After this section, add the following:

```
<servlet>
<servlet-name>CompozeExchangeAttachmentServlet</servlet-name>
<servlet-class>com.compoze.exchange.AttachmentServlet</servlet-class>
</servlet>
```

4. In the same `web.xml` file, navigate to the section that begins with the following:

```
<!-- Compoze Servlet Mapping -->
```

5. After this section, add the following:

```
<servlet-mapping>
<servlet-name>CompozeExchangeAttachmentServlet</servlet-name>
<url-pattern>*.compozeexchangeattachmentservlet</url-pattern>
</servlet-mapping>
```