

MOTOROLA eCARE 5.2.1 SERVER MANUAL



MOTOROLA

Copyright © 2002-2008 Motorola, Inc. All Rights Reserved.
Last Modified June, 2008

Copyright notice

Copyright © 2002-2008 Motorola, Inc. v. 062008

All rights reserved.

This manual and any associated artwork, software, product designs or design concepts are copyrighted with all rights reserved. Under the copyright laws this manual or designs may not be copied, in whole or part, without the written consent of Motorola. Under the law, copying includes translation to another language or format.

Motorola, Inc.

Marketplace Tower

6001 Shellmound Street, 4th Floor

Emeryville, CA 94608

USA

Part Number

This manual is Motorola part number 6161146-PF-07 (released June, 2008).

CONTENTS

Chapter 1: Introduction to eCare	6
What is eCare?	6
eCare Server Components	7
eCare Services	8
eCare Entry Portal URLs	8
Planning an eCare Installation	9
System Requirements	10
Proxy Server Requirements	12
Chapter 2: The eCare Installation Process	13
Making Configuration Decisions	13
eCare Overrides	14
Selecting an Installation Script	14
Installing eCare	16
Adding Additional eCare Services	25
Configuring Portals	27
Configuring the Email Invite Portal	27
Configuring the Reconnect Portal	29
Configuring the TicketConnector Portal	30
Configuring the Agent Passthru Portal	32
Configuring the Admin Passthru Portal	34
Removing an eCare Service	35
Server Maintenance	37
Stopping and Restarting the WTP and eCare Page Servers	37
Stopping and Restarting PostgreSQL	37

Chapter 3: Configuring Your eCare Installation	38
Setting Your Facility Code	39
Localizing eCare	39
Creating and Translating the Properties File.....	39
Turning on Localization	41
Localization Limitations	42
Configuring Security Event Table Validation	42
Security Event Table Validation Syntax	43
Configuring Startup Validation	43
Configuring Daily Validation	45
Configuring eCare Surveys	46
Adding Service Names to Surveys	47
Setting Up IP Blocking.....	48
Specifying Support Agent IP Addresses	48
Specifying Remote User IP Addresses	49
Setting Up a Dual-Homed Server	51
Enabling and Configuring eCare Features	54
Enabling the Control As Admin Service.....	55
Enabling Session Recording.....	55
Enabling Audible Alerts.....	55
Configuring Custom Message Display.....	56
Configuring Report Order and Disabling Reports.....	57
Disabling eCare Features.....	58
Disabling Screen-Sharing Functionality	58
Disabling Examine System	60
Disabling File Transfer	60
Disabling Push URL and Support Agent Tools	61
Disabling Chat	62
Disabling Managed Scripting	63

Chapter 4: Configuring eCare Deployables	64
Capabilities	65
Capabilities in eCare 5.2.1.....	65
Resolving Capabilities.....	66
Deployables.....	66
Deployables in eCare 5.2.1.....	66
Configuring Deployable Policies	67
The login Deployable Plan.....	68
The user-generated-event Deployable Plan	69
The deployable-class Configuration Elements	69
Configuring Specific Deployables to Install at Login.....	70
Configuring Installation of the Remote-Control Component.....	71
Configuring Installation of the Remote Scripting Deployable.....	76
Configuration Summary	77
Configuring Deferred Installation of Deployables	77
Configuring Deferred Installation of the Remote-Control Component	78
Configuring Deferred Installation of the Managed Scripting Deployable.....	79
Understanding the Interaction Between Deployables	80
Enabling Multiple Capabilities with Multiple Deployables.....	80
Disabling Capabilities with Deployables.....	81
Disabling Managed Scripting	83
Appendix A: Installing the eCare Remote-Control Component	84
Installing the JavaRC Applet	85
Installing the eCare Remote-Control Component on Windows Computers	85
Before You Install the eCare ActiveX Control	85
Installing the eCare ActiveX Control with the MSI Installer	86
Installing the eCare ActiveX Control with Component Files.....	90
Installing the eCare ActiveX Control on a Local Computer.....	93
Installing the eCare Remote-Control Component on Macintosh Computers	94
Before You Install the eCare Plugin.....	94
Pre-Installing the eCare Plugin on a Remote Macintosh Computer	94
Pre-Installing the eCare Plugin on the Local Macintosh Computer	95
Appendix B: Motorola Contacts	96

CHAPTER 1: INTRODUCTION TO eCARE

This document describes how to install and configure the server-side components of Motorola's eCare software application. It is intended to provide the server administrator with a good foundation for understanding Motorola's eCare server and how it operates. This document is written for the professional audience and assumes a professional-level understanding of Unix or Unix-like operating systems, Resin or similar servlet engines, and Java.

This document includes

- A description of the Motorola eCare application and server components.
- A detailed discussion of planning an eCare installation, including preplanning considerations such as system requirements and server-software configuration.
- Instructions for installing eCare, including installing multiple services, customizing services, and upgrading existing services.
- A summary of proxy server requirements.
- A list of download locations for required software components.

Successful installation and administration of the eCare application may also require technical documentation from the vendors of the associated required software.

WHAT IS eCARE?

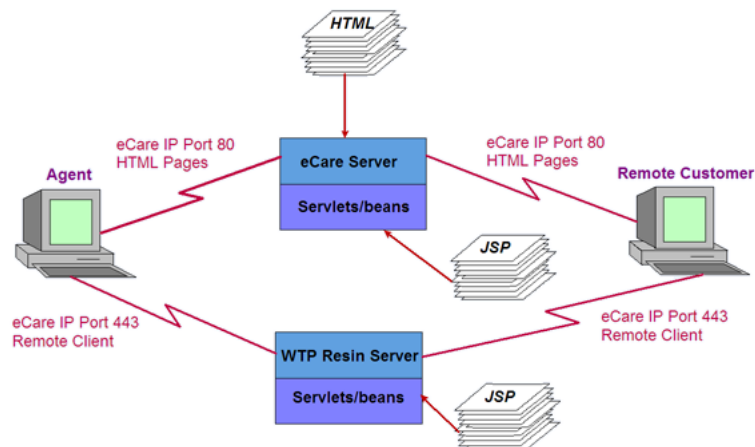
Motorola's eCare application is a cost-effective remote support service that allows Support Agents and their customers to interact in real time. Using a thin-client architecture, eCare enables remote problem solving over the Internet or your IP enterprise network.

Highlights of the eCare feature set include

- Screen sharing services that directly connect your Support Agents and the users they're assisting. Support agents can view the customer's screen and remotely operate their mouse and keyboard.
- Instant text-chat communication between Support Agents and remote users. If the remote user needs additional help at any time, the chat session can be seamlessly escalated to remote-control or any of the other eCare services.
- Real-time Push URL, which allows your Support Agents to sell products and help customers navigate through entire Web sites.
- File Transfer from the Support Agent's desktop directly to the customer's desktop instantaneously and vice versa.
- An online support request queue that allows your Support Agents to connect to customers instantly by clicking on a Web link.

ECARE SERVER COMPONENTS

The eCare server is comprised of two communications servers.



eCare Page Server A Resin server that handles requests associated with eCare HTML and JSP pages. The eCare page server has an Internet-available IP address and runs on port 80.

eCare 5.2 requires Resin 3.1.2.

Web Tunneling Protocol (WTP) Server

A Resin server that handles requests from the remote eCare client via Secure Socket Layer (SSL). The WTP server is available to the Internet and is used only by the eCare client. Customers will never load the WTP URL in their browsers. The WTP server has an Internet-available IP address and runs on port 443 (SSL).

E CARE SERVICES

An *eCare service* is an instance of eCare with its own Support Agents, queue, and maximum number of concurrent Support Agents. You may configure as many services as you wish. A Support Agent may be a member of more than one service and view multiple queues.

You may add services at any time, provided that you have created the databases and JNDI resource references for these services in advance. To remove services, however, you must shut down the WTP and eCare page servers. We suggest that you have a “test” service available at all times for troubleshooting.

You may give your services any name you wish. The service name is attached to the end of the eCare URL. For example, *http://support.ntpa.com/ecare*.

E CARE ENTRY PORTAL URLS

eCare uses four primary URLs to allow individual access points for customers and remote users, Support Agents, and eCare administrators. The primary entry point, located at the top level of the eCare service, is intended for your eCare customers. This top-level URL is in the form

```
http://<ecare-server>/<service-name>
```

or

```
http://<ip-address>/<service-name>
```

For example, *http://support.ntpa.com/ecare* opens the trouble-ticket submission form.

In the following URLs, `<service-root>` is the path to the top-level eCare service. In most cases, this is in the form

```
http://<ecare-server>/<service-name>
```

The four primary eCare entry portal URLs are

http://[service-root] Opens the eCare trouble ticket submission form. Remote users who require assistance will use this URL to access your eCare service center. This is the link you should provide to your customers.

http://[service-root]/agent
Opens the eCare trouble-ticket queue. Support Agents must log in before they can access the queue.

http://[service-root]/admin
Opens the main eCare administration page. eCare administrators must log in before they can access the administration features.

http://[service-root]/manage
Opens the Marae Administration page. eCare administrators can use this entry portal to stop and restart individual eCare services. Only site and super admin-level accounts can log in to stop and restart an eCare service. Other eCare accounts are not authorized to access this entry portal.

PLANNING AN eCARE INSTALLATION

Before installing eCare, make sure your server meets the following system requirements. If your organization uses a proxy server, it must meet the requirements listed under [“Proxy Server Requirements” on page 12](#).

SYSTEM REQUIREMENTS

Your eCare system requires the following hardware, software, and settings.

RECOMMENDED HARDWARE AND OPERATING SYSTEMS

eCare is supported on the following systems.

- Sun Solaris 8 or higher—Sparc platform
- Red Hat Enterprise Linux 4, update 3—Intel platform; CentOS 4.3 (other versions of Linux have not been tested and are not supported by Motorola)

SYSTEM MEMORY

Follow these guidelines for system memory sizing requirements.

- 20MB for Resin
- 12MB for each eCare service
- 4MB per Support Agent that will be logged into the eCare service

MOTOROLA eCARE SERVER SOFTWARE

To run eCare, you will need to obtain the following files from Motorola.

- One or more *appLicense.xml* license files
- The eCare distribution ZIP file
- If SSL is enabled, certain certificate resources (such as a Verisign certificate)

REQUIRED SOFTWARE AND DOWNLOAD LOCATIONS

To install eCare, you must download, install, and properly configure the following software. This software can be downloaded from the URLs provided below. (Note that these URLs may be changed at any time by the respective vendors of the listed software.)

J2SE SDK	Download J2SE SDK Java 1.6.0_01 (Java version 1.6, update 1). (Versions later than 1.6.0_01 have not been fully tested with eCare.) <i>http://java.sun.com/products/archive</i>
PostgreSQL	Download PostgreSQL 8.1.4. (Later versions have not been fully tested with eCare.) Select the distribution that is appropriate for the version of Linux you are running.

<http://wwwmaster.postgresql.org/download/mirrors-ftp>

or

<http://www.postgresql.org/mirrors-ftp.html>

You will not need to download a driver; the JDBC driver *postgresql-8.1-410.jdbc3.jar* is provided with the eCare installation package.

Note: eCare is not compatible with the 8.1-409 JDBC3 driver recommended on the PostgreSQL Web site.

Resin

eCare uses Resin as its servlet container. Download Resin 3.1.2.

<http://www.caucho.com/download/index.xtp>

OPERATING SYSTEM CONSIDERATIONS

Linux

By default, Red Hat Enterprise Linux comes installed with a firewall enabled. During the Red Hat installation procedure, remember to configure the firewall to allow network access to port 80 for the eCare page server, port 443 for the eCare WTP server, and port 5432 for the PostgreSQL server.

During your Linux installation, be sure to keep the following points in mind.

- Ensure that when you install Linux it does not automatically start a Web server that contends for a port you plan to use.
- If your Linux installer installs PostgreSQL, make sure that the installed version is compatible with eCare requirements. If necessary, replace it with a version that is compatible.
- If your Linux installer installs Java, and it is not the version required by eCare, be sure that the Resin container running eCare uses the correct JVM. Set the JAVA_HOME environment variable as needed.

Solaris

There are several patches required for Java to operate correctly. The Java installation instructions include procedures for checking your system and installing the necessary patches.

PROXY SERVER REQUIREMENTS

eCare can make connections using any proxy that correctly implements SSL. You must enable SSL on the eCare server to properly support these proxies.

The following methods of proxy-server determination are supported.

- No proxy
- Explicit or direct proxy for all protocols (with and without exceptions)
- Explicit or direct proxy for specific protocols (with and without exceptions)
- Auto configure script (DAT or PAC script)
- Auto detection (WPAD)

In addition, the proxy server must be WPAD compliant. It may use either Basic or Digest authentication. eCare does not currently support wildcard forms of the DNS name or IP address in the exception list in your browser. Your network administrator must configure one or all these methods.

For organizations using the WPAD protocol, eCare will request the DAT file located at <http://wpad/wpad.dat>. Successful connections require your organization to have a Domain Name resolved computer with a domain name that resolves to WPAD.

CHAPTER 2: THE eCARE INSTALLATION PROCESS

The installation procedures outlined in this document are for installing eCare on a Red Hat Linux platform using the recommended versions of prerequisite software. With the exception of the J2SE SDK, the installation procedure is identical on a Sun Solaris system. Consult your documentation from Sun Microsystems on how to properly install the J2SE SDK.

Unless otherwise noted, all commands in the following installation procedure are run using a regular user account. A `sudo` command is issued (and noted) when root privileges are required. If your regular user account does not have `sudo` privileges then you will need to run these commands under the root account instead.

All commands specified in this document should be entered exactly as described. If a command is too long to fit on one line within this document, it will be broken up into multiple pieces with the Unix continuation character, a backslash (`\`).

Placeholders are noted in red and surrounded by brackets: `[placeholder]`. Always substitute placeholders with values appropriate for your installation.

MAKING CONFIGURATION DECISIONS

Before beginning the installation process, make sure you know the answers to the following questions.

1. What host name, port, and service name do you want to use for the eCare server and the WTP server? Your choices will determine the eCare portal URLs, and they will affect the values used to configure communication between the page server and WTP server.
2. Where will the Postgres server be hosted? You will need the host name and port number.
3. What name will you give the eCare database? There are benefits to using the same name for the database and the database user.

4. What user will eCare be when it accesses the database? Decide now on a user name and password.
5. Select your JNDI name. Single-service installations should use the default JNDI name **ecare-postgres**, which will be supplied automatically by certain installation scripts. Multiple service installations will want to use a different naming scheme.
6. Decide whether you will run WTP as a secure service (SSL) or not. If WTP will run as a secure service, you will need to set up a keystore and have the password available during eCare installation.

If you use SSL, tunnelled screen-sharing communication will be blocked less frequently. However, the WTP server load will be higher because of the encryption process. In most cases, SSL is recommended.
7. Will you need the TicketConnector portal? If you are planning to integrate eCare with an existing CRM system, the answer will be yes.
8. Will you need Support Agent and administrator passthru portals? If you plan to allow Support Agents and administrators to sign in automatically through an existing CRM system, the answer will be yes.
9. Will you need to install the Email Invite and Reconnect portals?
10. Determine the email addresses from which eCare surveys will be mailed, and the delivery address to which they will be sent.

E CARE OVERRIDES

eCare is designed to allow “configuration by exception.” The default eCare configuration resides in the *ecare.xml* file. You will make changes as “overrides” in eCare’s overrides file. This design simplifies the upgrade process and provides you with a clear record of the changes you made during installation and configuration. By default, the overrides file is located in the *ecare4overrides* directory.

SELECTING AN INSTALLATION SCRIPT

eCare contains one interactive and several batch-mode installation scripts.

You should select and use only the script which is appropriate for your situation. These top level scripts are as follows.

If you are upgrading eCare from a previous version, and not performing a new installation, see the *eCare Release Notes* for upgrade instructions.

INSTALL.SH

An interactive script that will configure and install all eCare and WTP components needed for a typical single-service eCare installation. Most self-hosted sites who are installing eCare for the first time, and not upgrading an existing eCare server, should use this script.

This script prepares all scripts you need for an installation in which WTP and eCare are co-located on the same machine, at the same URL, but at different ports. The configuration assumes WTP and eCare will run in separate instances of Resin and that WTP will run on port 8080.

A record of the install values will be written to the directory from which the install runs, as well as to the overrides directory.

INSTALL2.SH

A properties-driven installer. This installer reads the configuration information for the install from the properties files “left behind” by an earlier use of `install.sh`. (It will also read properties files created manually.)

NORMALSERVICEINSTALL.SH

The command-line version of `install.sh`. Some options (such as the JNDI name for the database connection) which are not configurable in `install.sh` can be set using this script. A hosting service installing the first eCare service on a new server may want to use this script.

ADDSERVICEINSTALL.SH

A command-line script that will add another service to a server that is already hosting other eCare services. Hosting services can use this script to add additional instances of the eCare page server to their existing installation.

INSTALLING eCARE

STEP 1: DOWNLOADING NEEDED FILES

Before you begin installation, download all needed software and configuration files and save them to a convenient location. You will need the distribution and license files from Motorola as well as the software listed under [“Required Software and Download Locations”](#) on page 10.

STEP 2: EXTRACTING THE eCARE DISTRIBUTION FILE

Copy the eCare distribution file to a directory on the server and unzip it. You will need one of its component scripts later in this procedure. However, do not attempt to install eCare at this time.

STEP 3: INSTALLING THE J2SE SDK

LINUX OPERATING SYSTEM

If Java is already installed on your Linux system, you may wish to remove it before proceeding. However, this is not required as long as the eCare installation scripts will run with the correct settings for JAVA_HOME and other variables. (Some of the scripts invoke Java to perform XML transformations and they require the correct version of Java to run without error.) If you wish to keep an earlier version of Java as the default version on your system, you may need to modify these scripts to ensure that they reference the correct version of Java.

TO INSTALL THE J2SE SDK ON LINUX

1. Locate the J2SE SDK file you downloaded from Sun Microsystems and select the appropriate installation package for your system. Follow the installation instructions provided.
2. Set the JAVA_HOME environment variable and add the Java executable to the PATH by adding these two lines to the end of the `/etc/profile` configuration file (replace `[java_home]` with the installation directory for your version of Java).

```
export JAVA_HOME=[ java_home ]/j2sdk[xxx]
export PATH=$PATH:[ java_home ]/j2sdk[xxx]/bin
```

3. Log out and log back in for these environment variables to take effect.

SOLARIS OPERATING SYSTEM

Sun Microsystems uses their own package management system for installation and removal of Java. Please follow Sun's directions for Java installation.

STEP 4: INSTALLING POSTGRESQL

Use the following procedure to install PostgreSQL. Be sure to replace the `xxx` in the code examples with the correct numbers for the version you are installing.

1. Create an operating-system user as which the Postgres process will run.
If you select a user name other than **postgres**, please remember to substitute it for all instances of the **postgres** user name that occur throughout the remainder of this manual. You will also need to provide this alternate user name to the eCare installation script when prompted.

```
sudo /usr/sbin/useradd postgres
```

2. Locate the PostgreSQL archive file you downloaded and install it according to the instructions that come with the distribution.
3. Create the directories where PostgreSQL will store its data and log files.

```
sudo mkdir /usr/local/ecare_pgsql \  
           /usr/local/ecare_pgsql/data \  
           /usr/local/ecare_pgsql/log
```

```
sudo chown DR postgres /usr/local/ecare_pgsql
```

4. Assume the identity of user **postgres** and add the PostgreSQL *bin* directory to the PATH of the **postgres** user by adding the following line to the end of the *.bash_profile* file for the **postgres** account.

```
PATH=$PATH:/usr/local/pgsql[xxx]/bin
```

5. Log out of the **postgres** account and log back in to allow the *.bash_profile* changes to take effect.
6. As user **postgres**, initialize the database.

```
sudo su - postgres  
initdb -W -E UNICODE --locale en_US.UTF-8 -D \  
       /usr/local/ecare_pgsql/data
```

7. Select and confirm a superuser password when prompted. You will be prompted again for this password by the eCare installation script.

8. Start a postmaster to run your database in the background.

```
postmaster -D /usr/local/ecare_pgsql/data >> \
           /usr/local/ecare_pgsql/log/logfile 2>&1 &
```

9. Verify that you are able to access the database.

```
psql template1
```

10. At the `psql` prompt enter the following command.

```
select now();
```

If you are unable to run this command successfully without any error messages, there is a problem with the database.

Enter the following command to exit.

```
\q
```

11. Add a database to store logging information for your eCare service. For easier tracking of your databases, you may wish to name your database with the same name you intend to use for your eCare service.

```
createdb [db_name]
```

12. Locate the `pgsql_schema.sql` file from the eCare distribution file. Run the file to populate the database with tables.

```
psql -d [db_name] -f <path>/pgsql_schema.sql
```

13. In the `/usr/local/ecare_pgsql/data/postgresql.conf` file, configure PostgreSQL to accept incoming TCP/IP connections by uncommenting the line

```
#tcpip_socket = false
```

and setting it to **true**.

14. Set PostgreSQL to listen on port 5432 by uncommenting the line

```
#port = 5432
```

You may also specify a different port number. However, if you change the port number, be sure to take note of it. The eCare installation script will prompt you for this port number. You will also need to adjust the firewall in Red Hat Linux to allow traffic on the port that you specified.

15. Now you will configure PostgreSQL to accept incoming TCP/IP connections from the eCare page server. Edit the `Ipv4-style local connections` section in the `/usr/local/ecare_pgsql/data/pg_hba.conf` file.

- Change `trust` to `password`.

- Add a line for the IP address of the eCare page server that will make a connection to the database.

```
host all all 127.0.0.1 255.255.255.255 password
host all all [ip_address] 255.255.255.255 password
```

16. Stop and restart PostgreSQL for the changes you just made to take effect.

```
pg_ctl -D /usr/local/ecare_pgsql/data stop
postmaster -D /usr/local/ecare_pgsql/data >> \
  /usr/local/ecare_pgsql/log/logfile 2>&1 &
```

17. Log out of the **postgres** account, back into your regular user account.

```
exit
```

STEP 5: INSTALLING RESIN AND THE POSTGRESQL JDBC DRIVER

Use the following procedure to install Resin and the PostgreSQL JDBC driver.

1. Locate the Resin archive file you downloaded from Caucho and extract the source code to the */usr/local* directory.

```
cd /usr/local
sudo tar xzvf [path]/resin-3.1.2.tar.gz
```

2. Create a soft link to Resin.

```
sudo ln -s resin-3.1.2 resin
```

The PostgreSQL JDBC driver will be installed in the Resin *lib* directory when you run the eCare installation script.

Note: If the PostgreSQL JDBC driver is inadvertently deleted after you install eCare, you can copy it from the eCare distribution archive. From the *lib* directory in the eCare distribution ZIP file, expand the *postgresql-8.1-410.jdbc3.jar* file. Or copy the files from the eCare ZIP file to

```
/usr/local/resin/lib/postgresql-8.1-410.jdbc3.jar
```

STEP 6: ENABLING SSL FOR SCREEN SHARING

This step is optional. If you are not using SSL, skip to the next step.

1. Create a keystore using the keytool supplied with the Java SDK.

```
cd /usr/local/resin
sudo mkdir keys
cd keys
sudo keytool -genkey -keyalg RSA -validity 1000
               -keystore server.keystore
```

2. Enter a keystore password of your choosing. You will be prompted for this password again by the eCare installation script.
3. Accept the default answer of [unknown] for all the questions. When asked to verify that your selections are correct enter **yes**.
4. When asked for a key password, press ENTER to use the same password as the keystore password.
5. Extract a certificate from the keystore using the keytool application.

```
sudo keytool -export -keystore server.keystore -rfc
               -file theCert.cer
```

6. When prompted, enter the password you selected in step 2.
7. Add the certificate to the JVM's list of authorized certificates. (Replace [java_home] with the installation directory for your version of Java.)

```
cd [java_home]/j2sdk[xxx]/jre/lib/security
sudo keytool -import -file /usr/local/resin/keys/
               theCert.cer -keystore cacerts -storepass
               changeit
```

8. When asked if you want to trust this certificate, enter **yes**.

STEP 7: RUNNING THE INSTALLATION SCRIPT

Before you install eCare, consult the *Install-HOWTO.txt* file in your eCare distribution ZIP file for additional installation instructions and tips not discussed in this document.

BEFORE RUNNING THE SCRIPT

Remember that if you have multiple versions of Java installed on your system, you must ensure that the installation scripts will invoke the correct version. To determine which version of Java will be invoked, enter

```
java -version
```

If this command does not report the correct version (for eCare 5.2, this should be Java 1.6.0_01), update your environment variables or symlinks as necessary before you run the eCare installation scripts.

Then proceed with the installation. If you are using `install.sh`, continue with [“Step 7a: Running the install.sh Script.”](#) If you are using one of the other scripts, see [“Step 7B: Running an Alternate Installation Script.”](#) (See [“Selecting an Installation Script” on page 14](#) for information about selecting the appropriate script.)

STEP 7A: RUNNING THE INSTALL.SH SCRIPT

1. Change to the directory containing the expanded eCare distribution. Copy the `appLicense.xml` license file provided by Motorola into the directory.
2. Launch the eCare installation script.

```
/bin/bash ./install.sh
```

Note that you **MUST** use `bash` (or a `bash`-like) shell to run this script. It uses pattern-matching features not present in `sh`.

3. Using the information you collected in [“Making Configuration Decisions” on page 13](#), answer the following prompts. Default values, when they exist, are enclosed in brackets.

What is the path to the directory where resin is installed? *example: /usr/local/resin*

What do you want to name the new service? [ecare5]: *The default service name is ecare5 if you do not specify one.*

What is the host name/address for this service (Ex: ecare.host.com or 192.168.0.2)? *Enter the host name or IP address of your server.*

What is the WatchDog port you want to run on?[6701]: *Enter the desired WatchDog port.*

What is the eCare Cluster port you want to run on? [6800]: *Enter the desired cluster port.*

What is the WTP Cluster port you want to run on?[6801]: *Enter the desired WTP cluster port.*

Will you be running the eCare server over SSL (y/n)?
[n]: *Enter y if you will be using SSL.*

What is the password you used for your eCare keystore?[ecare5]: *Enter the password you selected for your keystore. (This prompt will appear only if you are running the eCare server over SSL.)*

What port will the server use (default 443)?[443]: *If you are running the eCare server over SSL, the default port is 443. If you are not using SSL, the default port is 80.*

What SMTP server should the page server use to relay email? *Enter the host name of an SMTP server that eCare can use to send email.*

What email address should be notified in event of a server failure? *Enter the email address of an administrator responsible for the eCare server.*

From what address should transcripts be emailed? *Enter the source email address that you want eCare to place in the From field for ticket transcripts.*

From what address should ticket notifications be emailed? *Enter the source email address that you want eCare to place in the From field for email notifications.*

Will you be running the wtp server over SSL (y/n)? [n]:
Enter y if you enabled SSL for screen sharing.

What is the password you used for your keystore? []:
Enter the password you selected for your keystore. (This prompt will appear only if you are running the WTP server over SSL.)

What port will you be running the wtp server on (default 444)?[444]: *If you are running the WTP server over SSL, the default port is 444. If you are not using SSL, the default port is 8080.*

What is the hostname of the postgres database that this service will use? *Enter the host name or IP address of the PostgreSQL server. This will usually be the same as the address for your eCare server unless you installed PostgreSQL on a different physical machine.*

What is the port number that the postgres server is running on? [5432]: *The default port is 5432 if you do not specify one.*

What is the name of the database that the eCare service will use? *Specify a database name; usually the name is the same as the name of your eCare service.*

What is the username that the service should use to access the postgres server? *Enter the user name, for example, **postgres**.*

What is the password for postgres user? *Enter the password, for example, **pgsql**.*

What default email address should the surveys be sent from? *Enter the source email address that you want eCare to place in the From field for surveys.*

What default email address should the surveys be sent to? *Enter the email address to which you want eCare to send survey results.*

It appears that you do not have a postgres jdbc driver installed in /usr/local/resin/lib. Would you like to install pgxx.2xx.jdbc3.jar (y/n)? [y]: *This prompt will only appear if you incorrectly installed the JDBC driver.*

What facility code do you want to assign to this instance of eCare? [16] *The facility code is used for logging purposes. It must be an integer from 0 to 23 as specified in RFC 3164, "The BSD Syslog Protocol." If you do not specify a different number, the default is 16.*

STEP 7B: RUNNING AN ALTERNATE INSTALLATION SCRIPT

If you wish to use one of the alternate installation scripts, follow this procedure.

1. If you are repeating an existing installation, **make a back up of your existing installation.**

At the least, make a copy (with the `-R` option) of the database contents, as well as the following directories:

- `resin/bin/*`
- `resin/conf/*`
- `resin/webapps/service/*`
- your overrides directory

In particular, if you have made changes to `web.xml`, `ecare.xml`, or `marae.xml`, copy those files to a safe place BEFORE launching the installer. The installation will **always** overwrite those files.

2. Change to the directory containing the expanded eCare distribution.
 - If this is a first-time installation, copy the *appLicense.xml* license file supplied with your distribution into the directory containing the expanded eCare distribution.
 - If this is a properties install (using `install2.sh`), locate a copy of `install.record`.

If you have previously completed an installation of this service, you should find the record of that installation under the `overrides` directory in a subdirectory with a name following the pattern

```
Install-<date>__<time>
```

If you made a mistake during the original installation, you can edit the values in this file, delete the WAR file and service directory from your server's `webapps` directory, and then repeat the installation with

```
/bin/bash ./install2.sh [path to properties files]
```

3. Navigate to the distribution directory and invoke the desired installation script with no arguments. It will display a brief usage summary. The arguments correspond to a subset of the information described for the `install.sh` script. The documentation at the top of each script provides more details about the form expected for certain arguments.
4. Invoke the desired installation script with the arguments appropriate for your installation.

Because some of the installation scripts require numerous arguments, you may find it helpful to copy the description from the script and place it above this command in its own script file.

If you work at the command line, we recommend using the line continuation character to keep the command visible. For example,

```
/bin/bash ./AddServiceInstall.sh /usr/local/resin \
  usr/local/resin/overrides \
  mail.local.server \
  admin@my.com www.companysite.com \
  - helpdesk - false - false
```

Note the use of hyphens (`-`) to request default values. Also notice that you must use `bash`, not `sh`.

5. When the installation is complete, you may need to manually merge site-specific customizations (saved in step 1 above) into your new *web.xml*, *marae.xml*, or *ecare.xml* files.

STEP 8: TESTING YOUR INSTALLATION

1. Start WTP. Navigate to `/usr/local/resin/bin` and execute the command

```
./run_wtp.sh start
```

Once Resin has expanded the WAR file, you should be able to view a diagnostics page at

```
http://<hostname>:8080/wtp
```

If you installed WTP in SSL mode, you will need to load

```
https://<hostname>/wtp
```

2. Wait at least 1 minute after starting the WTP server before starting the eCare page server.
3. Start the page server. In `/usr/local/resin/bin` execute the command

```
./run_eCare.sh start
```

Once Resin has expanded the WAR file, you should be able to access the ticket submission page by visiting

```
http://<hostname:portnumber>/<serviceName>
```

Now you can access your new eCare service using one of the following URLs.

Ticket submission `http://<hostname:portnumber>/<serviceName>`

Support Agent portal

```
http://<hostname:portnumber>/<serviceName>/agent
```

Administrator portal

```
http://<hostname:portnumber>/<serviceName>/admin
```

ADDING ADDITIONAL eCARE SERVICES

Once your first eCare service is installed, you can add additional services with the `AddServiceInstall.sh` script.

TO ADD ADDITIONAL eCARE SERVICES

1. Add another database to store logging information for your new eCare service. For easier tracking of your databases, you may wish to name your database with the same name you intend to assign your eCare service.

```
sudo su - postgres
createdb [db_name]
```

2. Populate your new database with tables.

```
psql -d [db_name] -f [path to extracted eCare
files]/conf/pgsql_schema.sql
exit
```

3. Edit the `/usr/local/resin/conf/resin.conf` file to add a JNDI resource reference for your new database. Copy the original database reference entry for your initial eCare service, paste it below this original entry, and replace `[db_name]` with the name of the database for the service you are adding. It is easiest to assign the JNDI database reference the same name as the eCare service you intend to add.

```
<database>
  <jndi-name>jdbc/ecare-postgres-[db_name]
  </jndi-name>
  <driver type='org.postgresql.Driver'>
    <url>jdbc:postgresql://[hostname]:[port]/
      [db_name]</url>
    <user>[username]</user>
    <password>[password]</password>
  </driver>
</database>
```

4. If you will need to add additional services in the future without bringing down the eCare page server, create several generic JNDI database references and their associated databases in advance. This allows you to add new eCare services without interrupting access to existing ones. Otherwise, you must restart the eCare page server for the new database references to take effect.
5. Run the `AddServiceInstall.sh` script to add your new eCare service. The values requested by the placeholders are identical to those that you were prompted for when you initially installed eCare and created your first service.

```
sudo bash AddServiceInstall.sh /usr/local/resin
ecare4overrides [smtp_server]
[admin_email_address] [hostname]
- [eCare_service_name]
- [true / false] (true if SSL; false if not)
jdbc/ecare-postgres-[db_name] false
[default-survey-from-email-address] [default-
survey-to-email-address]
```

For usage instructions, you may run the `AddServiceInstall.sh` script by itself without any parameters.

```
sudo bash AddServiceInstall.sh
```

Now you can access your new eCare service using one of the following URLs.

Ticket submission *http://<service-root>*

Support Agent portal

http://<service-root>/agent

Administrator portal

http://<service-root>/admin

CONFIGURING PORTALS

Once you have completed the normal installation process, you can configure new portals. By default, only the customer, Support Agent, and administrator portals are configured with the standard eCare installation process. If you wish to use additional portals, you will need to configure them separately.

CONFIGURING THE EMAIL INVITE PORTAL

The Email Invite portal makes it possible for your Support Agents to send email invitations, which allow the Support Agent to fill out the trouble-ticket form on the customer's behalf. The Support Agent uses the Send Invitation service to open the trouble ticket and email it to the customer. When the customer clicks the URL included in the email, their Web browser will connect to eCare through the Email Invite portal.

TO CONFIGURE THE EMAIL INVITE PORTAL

1. Run the `AddEmailInvitePortal.sh` script.

```
bash AddEmailInvitePortal.sh [ecare override infile]
                             [hostname] [ecarePort] [serviceName]
                             [clientconnecttimeout] [agentconnecttimeout]
                             [ecare override outfile]
```

For usage instructions, you may run the `AddEmailInvitePortal.sh` script by itself without any parameters.

```
bash AddEmailInvitePortal.sh
```

2. Access the Marae Administration page at the following URL to restart the eCare service.

http://<server>/<service>/manage
3. Click the *Initiate Marae System Shutdown* link and wait for the shutdown to complete.

The eCare service is not completely shut down until the current state is shown as *Shutdown* and the *Start Marae System* link appears. The eCare service will then be unavailable until you restart it.
4. When the shutdown is complete, click the *Start Marae System* link to restart your eCare service.

TO TEST THE EMAIL INVITE PORTAL

1. Sign into the eCare system as a Support Agent.
2. Verify that the *Email Invite* button appears above the trouble-ticket queue.
3. Click the *Email Invite* button and fill out the invitation with an email address you can check on a different computer. Then click *Submit*.

Note: If you receive the email on the same computer, you will not be able to submit the trouble ticket correctly while you are signed in to eCare as a Support Agent. If you sign out from the Support Agent portal, you will not be able to confirm that the ticket appears in the queue.
4. When the email arrives, click the eCare link in the message body.

The trouble-ticket is submitted automatically and you are placed in the eCare queue.
5. On the Support Agent computer, verify that the trouble ticket appears in the queue.

CONFIGURING THE RECONNECT PORTAL

During an eCare session, it may occur that your connection with the customer fails. (For example, the customer's network connection may be interrupted.) When a Support Agent accepts an eCare trouble ticket, and the eCare reconnect component is enabled on your eCare service, the Activate dialog box opens on the customer's screen, asking them to activate the reconnect component. If the customer clicks *Yes*, eCare saves a unique URL as a shortcut on the customer's desktop. If the eCare session is interrupted, the customer can double-click the shortcut to return to the trouble-ticket queue with the same ticket ID as before. When the customer double-clicks the shortcut, their Web browser will connect to eCare through the Reconnect portal.

The Reconnect portal is also used when a customer's computer reconnects to eCare automatically after the Support Agent reboots it with the Reboot Remote System service.

TO CONFIGURE THE RECONNECT PORTAL

1. Run the `AddReconnectPortal.sh` script.

```
bash AddReconnectPortal.sh [ecare override infile]
                           [hostname] [ecarePort] [serviceName]
                           [clientconnecttimeout] [agentconnecttimeout]
                           [ecare override outfile]
```

For usage instructions, you may run the `AddReconnectPortal.sh` script by itself without any parameters.

```
bash AddReconnectPortal.sh
```

2. Access the Marae Administration page at the following URL to restart the eCare service.

http://<server>/<service>/manage

3. Click the *Initiate Marae System Shutdown* link and wait for the shutdown to complete.

The eCare service is not completely shut down until the current state is shown as *Shutdown* and the *Start Marae System* link appears. The eCare service will then be unavailable until you restart it.

4. When the shutdown is complete, click the *Start Marae System* link to restart your eCare service.

TO TEST THE RECONNECT PORTAL

1. Sign into the eCare system as a Support Agent.
2. On a different computer, running Windows 2000 or Windows XP, submit an eCare trouble ticket.
Verify that you are prompted to install the eCare ActiveX control.
3. On the Support Agent computer, make a note of the ticket ID number. Then accept the trouble ticket.
4. On the customer computer, verify that you are prompted to accept the Reconnect control.
This Reconnect control will allow you to reconnect to eCare when your session is interrupted. eCare will place a shortcut on your desktop. Verify that the shortcut is saved there.
5. Once the eCare session has been established, simulate a connection failure by closing the customer's Web browser.
6. On the customer computer, double-click the desktop shortcut to reconnect to eCare.
7. On the Support Agent computer, verify that the trouble ticket reappears in the queue with the same ticket ID number.

CONFIGURING THE TICKETCONNECTOR PORTAL

The TicketConnector portal provides an integration point for eCare into your existing CRM solution. eCare's TicketConnector portal feature lets external systems circumvent the ticket submission page and submit a ticket directly to an eCare queue. This is done by sending a request to the TicketConnector portal URL while passing all of the ticket submission fields as query string parameters.

See the *eCare Integration Guide* for more information about using eCare's integration capabilities.

TO CONFIGURE THE TICKETCONNECTOR PORTAL

1. Run the `AddClientPassthruPortal.sh` script. This script will make the necessary changes to the overrides file for you.

```
sudo bash AddClientPassthruPortal.sh /usr/local/resin/
ecare4overrides/[service]-ecare.xml [hostname]
" " [service] /usr/local/resin/ecare4overrides/
[service]-ecare.xml
```

For usage instructions, run the `AddClientPassthruPortal.sh` script by itself without any parameters.

```
sudo bash AddClientPassthruPortal.sh
```

2. In the overrides file, locate the following lines.

```
<portal name="query-login">
  <role>client</role>
  <login-form auto='true'>GuestLogin.jsp</login-form>
  <param name="passthru">true</param>
  <param name="external-ticket-id">required</param>
  ...
</portal>
```

Edit the `<login-form>` element and add a new `<param>` element as shown below.

```
<portal name="query-login" action="replace">
  <role>client</role>
  <login-form auto='true'>
    ProvisionedTicketLogin.jsp</login-form>
  <param name="passthru">true</param>
  <param name="external-ticket-id">required</param>
  <param name="entry-option">provisioned</param>
  ...
</portal>
```

3. Access the Marae Administration page at the following URL to restart the eCare service.

`http://<server>/<service>/manage`

4. Click the *Initiate Marae System Shutdown* link and wait for the shutdown to complete.

The eCare service is not completely shut down until the current state is shown as *Shutdown* and the *Start Marae System* link appears. The eCare service will then be unavailable until you restart it.

5. When the shutdown is complete, click the *Start Marae System* link to restart your eCare service.

Once your eCare service has been restarted, you can submit a trouble ticket with the TicketConnector portal.

USING THE TICKETCONNECTOR PORTAL

When a ticket is submitted to the TicketConnector portal, eCare will automatically create a ticket and forward the browser directly to the waiting-in-line screen. By default the TicketConnector portal expects the following query string parameters.

firstname	The customer's first name.
lastname	The customer's last name.
email	The customer's email address.
phone	The customer's phone number.
problem	A description of the customer's problem or issue.

These are the same fields that normally appear in the ticket submission form. Depending on what customer information you want to track, you can pass as many or as few of these fields in the TicketConnector portal URL as you want.

To bypass the ticket submission page and submit a trouble ticket automatically through the TicketConnector portal, use a URL like the following.

```
http://<server>/<service>/TicketConnector.jsp?firstname=<name>&lastname=
<name>&email=<address>&phone=<999-999-9999>&problem=<description>
```

When the customer's browser accesses this URL, they are forwarded automatically to the waiting-in-line screen.

CONFIGURING THE AGENT PASSTHRU PORTAL

The Agent Passthru portal enables an existing CRM system to log a Support Agent into eCare, bypassing the Support Agent login screen.

See the *eCare Integration Guide* for more information about using eCare's integration capabilities.

TO CONFIGURE THE AGENT PASSTHRU PORTAL

1. Run the `AddAgentPassthruPortal.sh` script. This script will make the necessary changes to the overrides file for you.

```
sudo bash AddAgentPassthruPortal.sh /usr/local/resin/
ecare4overrides/[service]-ecare.xml [hostname]
[port] [service] [userReconnectTimeout] /usr/
local/resin/ecare4overrides/[service]-ecare.xml
```

The `AddAgentPassthruPortal.sh` script includes two optional parameters: `loginFailurePage` and `failureAction`.

- `loginFailurePage` specifies a URL to load in the event of a login failure at the Agent Passthru portal.
- `failureAction` is required if `loginFailurePage` is supplied; it specifies whether to forward or redirect to the URL provided by `loginFailurePage`. Its value must be `forward` or `redirect`.

For usage instructions, run the `AddAgentPassthruPortal.sh` script by itself without any parameters.

```
sudo bash AddAgentPassthruPortal.sh
```

2. Access the Marae Administration page at the following URL to restart the eCare service.

```
http://<server>/<service>/manage
```

3. Click the *Initiate Marae System Shutdown* link and wait for the shutdown to complete.

The eCare service is not completely shut down until the current state is shown as *Shutdown* and the *Start Marae System* link appears. The eCare service will then be unavailable until you restart it.

4. When the shutdown is complete, click the *Start Marae System* link to restart your eCare service.

Once your eCare service has been restarted, you can use the Agent Passthru portal.

USING THE AGENT PASSTHRU PORTAL

To bypass the Support Agent login screen and log in the Support Agent automatically through the Agent Passthru portal, the CRM system can submit a URL in the following format.

```
http://<server>/<service>/AgentDirect.jsp?agentid=<agentID>&authtoken=
<password>
```

By default, the Agent Passthru portal expects the following query string parameters:

agentid	The Support Agent's user ID.
authtoken	The Support Agent's password.

When the Support Agent's browser loads this URL, the Support Agent is forwarded automatically to the main eCare trouble-ticket queue.

CONFIGURING THE ADMIN PASSTHRU PORTAL

The Admin Passthru portal enables an existing CRM system to log an eCare administrator into eCare and bypass the administrator login screen.

See the *eCare Integration Guide* for more information about using eCare's integration capabilities.

TO CONFIGURE THE ADMIN PASSTHRU PORTAL

1. Run the `AddAdminPassthruPortal.sh` script. This script will make the necessary changes to the overrides file for you.

```
sudo bash AddAdminPassthruPortal.sh /usr/local/resin/
ecare4overrides/[service]-ecare.xml [hostname]
[port] [service] [userReconnectTimeout] /usr/
local/resin/ecare4overrides/[service]-ecare.xml
```

The `AddAdminPassthruPortal.sh` script includes two optional parameters: `loginFailurePage` and `failureAction`.

- `loginFailurePage` specifies a URL to load in the event of a login failure at the Admin Passthru portal.
- `failureAction` is required if `loginFailurePage` is supplied; it specifies whether to forward or redirect to the URL provided by `loginFailurePage`. Its value must be `forward` or `redirect`.

For usage instructions, run the `AddAdminPassthruPortal.sh` script by itself without any parameters.

```
sudo bash AddAdminPassthruPortal.sh
```

2. Access the Marae Administration page at the following URL to restart the eCare service.


```
http://<server>/<service>/manage
```
3. Click *Initiate Marae System Shutdown* and wait for the shutdown to complete. The eCare service is not completely shut down until the current state is shown as *Shutdown* and the *Start Marae System* link appears. The eCare service will then be unavailable until you restart it.

4. When the shutdown is complete, click the *Start Marae System* link to restart your eCare service.

Once your eCare service has been restarted, you can use the Admin Passthru portal.

USING THE ADMIN PASSTHRU PORTAL

To bypass the administrator login screen and log in the administrator automatically through the Admin Passthru portal, the CRM system can submit a URL in the following format.

```
http://<server>/<service>/AdminDirect.jsp?adminid=<adminID>&authtoken=
<password>
```

By default, the Admin Passthru portal expects the following query string parameters:

adminid	The administrator's user ID.
authtoken	The administrator's password.

When the administrator's browser loads this URL, the administrator is forwarded automatically to the main eCare administration page.

REMOVING AN eCARE SERVICE

1. Stop the WTP and eCare page servers.


```
cd /usr/local/resin/bin
sudo ./run_eCare.sh stop
sudo ./run_wtp.sh stop
```
2. Determine which JNDI resource reference is associated with the service to be deleted.


```
grep jndi /usr/local/resin/ecare4overrides/
[service]-ecare.xml
```
3. Use the information returned by the command in the previous step to find the database element associated with the service to be deleted in the */usr/local/resin/conf/resin.conf* file. For example, the command returned


```
<jndi-name> jdbc/ecare-postgres</jndi-name>
```

4. Search for the database element in the *resin.conf* file that contains the element


```
<jndi-name>jdbc/ecare-postgres</jndi-name>
```
5. Once you find the database element associated with the service to be deleted, determine which database this associated with this resource reference by examining the line that contains the `<init-param url>` tag.
6. Once you find the database element associated with the service to be deleted, examine the `<url>` tag to determine which database the service is using. The last piece of text following the port number is the name of the database associated with the JNDI reference.

In the following example, the `msdb` database is associated with the `ecare-postgres` JNDI reference.

```
<database>
  <jndi-name>jdbc/ecare-postgres</jndi-name>
  <driver type='org.postgresql.Driver'>
    <url>jdbc:postgresql://192.168.1.141:5432/
      msdb</url>
    <user>postgres</user>
    <password>pgsql</password>
  </driver>
</database>
```

7. Remove the database element associated with the service to be deleted. This includes everything between the opening and closing `<database>` element tags.
8. Delete the database associated with the service being removed.

```
sudo su - postgres
dropdb msdb
exit
Delete the files for the service.
sudo rm -rf /usr/local/resin/webapps/[service]
/usr/local/resin/webapps/[service].war
rm /usr/local/resin/ecare4overrides/
[service]-ecare.xml
```

9. Restart the WTP and eCare page servers. Wait at least 1 minute after starting the WTP server before starting the eCare page server.

```
cd /usr/local/resin/bin
sudo ./run_wtp.sh start
sudo ./run_eCare.sh start
```

SERVER MAINTENANCE

On occasion, you may need to stop and restart the PostgreSQL, WTP, and eCare page servers. Below are instructions on how to accomplish this.

STOPPING AND RESTARTING THE WTP AND eCARE PAGE SERVERS

In some instances you need to stop and restart the WTP and eCare page servers to allow system changes to take effect, such as when you add a JNDI resource reference to the *resin.conf* file. Run the following commands to stop the eCare servers.

```
cd /usr/local/resin/bin
sudo ./run_eCare.sh stop
sudo ./run_wtp.sh stop
```

To restart the WTP and eCare page servers execute the commands below.

```
cd /usr/local/resin/bin
sudo ./run_wtp.sh start
sudo ./run_eCare.sh start
```

Wait at least one minute after starting the WTP server before starting the eCare page server.

STOPPING AND RESTARTING POSTGRESQL

At times you may need to stop and restart PostgreSQL. The commands to do this must be run under the postgres system user account. Use the following commands to stop the PostgreSQL server.

```
sudo su - postgres
pg_ctl -D /usr/local/ecare_pgsqldata stop
exit
```

The following commands restart the PostgreSQL server.

```
sudo su - postgres
postmaster -D /usr/local/ecare_pgsqldata >> \
  /usr/local/ecare_pgsqldata/log/logfile 2>&1 &
exit
```

CHAPTER 3: CONFIGURING YOUR eCARE INSTALLATION

All of the following eCare customizations require changes to an eCare configuration file such as the overrides, *ecare.xml*, or localization file for the particular eCare service you are changing.

- Changes to these configuration files are noted in [blue](#).
- [\[placeholders\]](#) are noted in red and surrounded by brackets. Always replace placeholders with values appropriate for your eCare installation.
- In some cases, line numbers are included in the configuration examples to help you distinguish between a new line and a one that is wrapped around because it is too long to fit on one line.

Problems with the overrides file or the *ecare.xml* file may prevent your eCare service from starting correctly. Before making any changes to either file be sure to make a backup copy in case you need to restore it later.

For changes to these files to take effect you must shut down and restart the eCare service. Shutting down an eCare service only affects that specific service and is different than restarting the entire eCare page and WTP servers. When you shut down a specific eCare service all other services are unaffected and still accessible. Use the following URL to stop and restart a specific eCare service.

[http://\[server\]/\[service\]/manage](http://[server]/[service]/manage)

You will be prompted to log in. Enter the credentials for a site or super admin-level account for this eCare service; standard administrator accounts do not have privileges to access this page.

Click the *Initiate Marae System Shutdown* link and wait for the shutdown to complete. The eCare service is not completely shut down until the current state is shown as *Shutdown* and the *Start Marae System* link appears. The eCare service will then be unavailable until you restart it after the upgrade is complete.

SETTING YOUR FACILITY CODE

eCare writes an **ops** log file in syslog format (as described in RFC 3164). The syslog format includes the use of a *facility code*—an integer that identifies the facility operating the server. The *ecare.xml* configuration file contains a default facility code as a top-level element. (It is a required configuration element.)

To change the facility code, specify a different code in the overrides file.

```
<log-facility-code action="replace">10</log-facility-code>
```

LOCALIZING eCARE

Depending on the customers that you need to support with eCare, you may wish to translate it to another language besides English or alter the default text strings used in the product for the purposes of branding it. The process of configuring your eCare server to support and display Web pages in a language other than English is known as *localization*. Localizing eCare and altering the default text strings both require you to modify a copy of the *Resource.properties* file for the particular service you want to change.

To create a localized eCare service, you must

- Translate the properties file into the language you wish to use.
- Enable the language-specific settings in the *[service]-marae.xml* file.

The following sections cover these processes in detail.

CREATING AND TRANSLATING THE PROPERTIES FILE

All the text that appears in the eCare user Web interface resides in one file, known as the resources or properties file. The resources file is named in the format

```
Resource_language_country.properties
```

where *language* is a two-letter code for the language used and *country* is a two-letter code for the country supported by the server. For example, the US English resources file is named *Resource_en_US.properties* and the Japanese resources file is named *Resource_ja_JP.properties*.

TO CREATE A LOCALIZED PROPERTIES FILE

1. Create a localization directory.

```
sudo mkdir /usr/local/resin/webapps/[service]/
ecare4/custom/localization
```

2. Copy the default *Resource.properties* file and rename it with the applicable two-letter country and language codes.

```
cd /usr/local/resin/webapps/[service]/ecare4/
custom/localization
sudo cp /usr/local/resin/webapps/[service]/WEB-INF/
classes/com/netopia/app/ecare/localization/
Resource.properties
Resource_[language]_[country].properties
```

The two-letter codes you use must match standard codes you can find at

<http://www.w3.org/International/O-misc-iso3166.html>

Note that some countries also have three-letter codes. You must use the two-letter code for localization to occur correctly.

If you wish to customize some of the default eCare text strings without fully localizing the file, retain the `en_US` naming scheme.

3. Open the new resources file in a text editor.

The resources file includes many text strings, each of which is comprised of a string name and a string value. Each string name starts on a new line and begins with the characters `jsp` or `java`. Each string name is followed by several tabs, an equals sign (=), and a string value.

```
jsp.client.submit.introLine = Welcome To eCare
```

The English phrase following the equals sign is the string value, which you must translate to the desired language.

4. Translate the string values into your desired language. Keep in mind that
 - You must translate only the string values, NOT the string names. The string names (the text that begins with `jsp` or `java` and precedes the equals sign) are used by the eCare server to identify the text strings. They will not be visible to your eCare users.
 - You must use Unicode Escape (UTF-8) formatting for all non-English string values. These Unicode Escape characters for non-English strings take the form `\unnnn` where `nnnn` is a 4-character code for the non-English character.

5. Once translation is complete, save the new resources file.
6. Create a soft link to the new *Resource_[language]_[country].properties* file.

```
cd /usr/local/resin/webapps/[service]/WEB-INF/
  classes/com/netopia/app/ecare/localization
sudo ln -s /usr/local/resin/webapps/[service]/
  ecare4/custom/localization/
  Resource_[language]_[country].properties
  Resource_[language]_[country].properties
```

Make sure to use the actual name of the file, replacing `[language]` and `[country]` with the appropriate two-letter codes.

TURNING ON LOCALIZATION

To enable your eCare service to run in a language other than US English, you must specify the desired language and country codes in the *[service]-marae.xml* file for your service. The language and country codes you need are the same as those you used to create the resources file (see the previous section, “[Creating and Translating the Properties File](#)”).

TO SPECIFY THE LOCALIZED FILE

1. Create a *[service]-marae.xml* file in the */usr/local/resin/ecare4overrides* directory with the following contents. Specify the new language and country codes with the `<locale>` element. Specify the two-letter language code first, followed by a hyphen (–) and the two-letter country code.

```
<marae>
  <configuration>
    <locale action="replace">[language]-[country]
      </locale>
    <!-- en-US should be the default -->
    <!-- de-DE is German -->
    <!-- ja-JP is Japanese -->
  </configuration>
</marae>
```

Note that the language and country codes are separated by a hyphen (–) and not an underscore (–) as in the resources file name.

2. Save the *[service]-marae.xml* file and restart your eCare service.

Note: The eCare pages will continue to be displayed in English instead of your localized language until after your eCare service is restarted.

LOCALIZATION LIMITATIONS

Please note the following limitations in the current eCare localization scheme.

- The eCare client software is not yet localized. Some messages that appear during screen sharing and Examine System sessions, which are generated by the client, will appear in English.
- Some Web browsers and operating systems do not fully support localization. If the remote user's browser is not the language specified for the eCare server, some characters may not appear correctly in some locations.

For example, if the eCare server is running a Japanese localization, but the remote computer is using an English operating system and Web browser, Japanese characters will not display correctly in the title bar and browser alert messages. This occurs even if the browser has Japanese language support.

- eCare report names do not reside in the *Resource.properties* file. To change report names, edit them in the `<reports>` section in the *ecare.xml* file.
- The names and descriptions of eCare roles do not reside in the *Resource.properties* file. To change role names and descriptions, edit them in the `<security_roles>` section in the *ecare.xml* file.

CONFIGURING SECURITY EVENT TABLE VALIDATION

The eCare database includes a table called **securityevent**, which records activities that may be relevant to the security of your eCare server. The only permissible operations on this table are adding records and reading records; records cannot be modified or deleted. In addition, eCare itself is the only entity that is permitted to add records. You can configure when (and if) the eCare server checks the **securityevent** table for tampering.

- The eCare server can perform a full or partial database check every time your eCare service is started. If any evidence of tampering is found, the eCare service will fail to start. An error message is written to both the **ops** log and the **start** log, indicating the nature of the problem.

To enable your eCare service to start after evidence of tampering is found, you must delete the contents of the **securityevent** table (be sure to save a copy of the table to retain evidence of the tampering).

This validation is enabled in the default eCare configuration.

- The eCare server can perform a full or partial database check once a day, at the time of day you specify. If any evidence of tampering is found, the server will write a message to the **ops** log. (The server will *not* shut down.)

This validation is not enabled in the default eCare configuration. Motorola strongly recommends that you enable this validation, specifying a full database check at the most appropriate time of day for your organization.

If you are using this option, be sure to monitor the **ops** log for these messages, which include the prefix `securityevent table:`.

Note that you may delete all records in the **securityevent** table, and eCare will not consider this evidence of tampering. This allows you to archive the table and prevent it from growing too large. Motorola recommends that you archive the **securityevent** table on a regular basis. You may then delete all records in the active table to optimize the server startup process.

You will configure Security Event Table validation in the *ecare.xml* or overrides file.

SECURITY EVENT TABLE VALIDATION SYNTAX

The `<security-event-table-check>` element is now required in the `<common>` section of the *ecare.xml* file. The following sections discuss the full syntax for startup or daily validation.

If you do not wish to validate the **securityevent** table, you may use an empty `<security-event-table-check/>` element to specify that no validation will occur.

CONFIGURING STARTUP VALIDATION

To configure eCare to validate the **securityevent** table when the server starts up, use the following elements.

`validate-on-startup`

Specifies that eCare will attempt to validate the **securityevent** table when the eCare server starts up.

REQUIRED <VALIDATE-ON-STARTUP> CHILD ELEMENTS

table-check-event

Specifies whether eCare performs a full or partial database validation, and, for partial validation, the number of records to validate, in the format

```
<table-check-event type="checkType"
  max-records="records" />
```

<TABLE-CHECK-EVENT> ATTRIBUTES

type	<p>Required attribute of the <table-check-event> element. Specifies whether eCare performs a full or partial securityevent table validation. Valid type values are full and partial.</p> <p>If type is set to full, eCare will validate the entire securityevent table. If type is set to partial, eCare will validate max-records in the table.</p>
max-records	<p>Optional attribute of the <table-check-event> element. Specifies the maximum number of records to be checked in a validation pass. Half of the records will be from the beginning of the securityevent table, and half will be from the end. If the number of records in the table is less than the number specified by max-records, all records will be checked.</p> <p>If type is set to partial, and max-records is not specified, eCare will validate 500 records.</p> <p>If type is set to full, do not specify a max-records value.</p>

EXAMPLE

The following sample configuration sets the eCare server to perform a partial **securityevent** table validation of 1000 records when the server starts up.

```
<security-event-table-check>
  <validate-on-startup>
    <table-check-event type="partial"
      max-records="1000" />
  </validate-on-startup>
</security-event-table-check>
```

CONFIGURING DAILY VALIDATION

To configure eCare to validate the **securityevent** table daily, at the time you specify, use the following elements.

`validate-daily` Specifies that eCare will attempt to validate the **securityevent** table at the time you specify.

<VALIDATE-DAILY> ATTRIBUTES

`time` Required attribute of the `<validate-daily>` element. Specifies the time of day that validation is performed, in *HH:MM* format. *HH* is an integer from 0–23; *MM* is an integer from 0–59. Leading zeros are allowable.

REQUIRED <VALIDATE-DAILY> CHILD ELEMENTS

`table-check-event` Specifies whether eCare performs a full or partial database validation, and, for partial validation, the number of records to validate, in the format

```
<table-check-event type="checkType"
  max-records="records" />
```

<TABLE-CHECK-EVENT> ATTRIBUTES

`type` Required attribute of the `<table-check-event>` element. Specifies whether eCare performs a full or partial **securityevent** table validation. Valid `type` values are `full` and `partial`.

If `type` is set to `full`, eCare will validate the entire **securityevent** table. If `type` is set to `partial`, eCare will validate `max-records` in the table.

`max-records` Optional attribute of the `<table-check-event>` element. Specifies the maximum number of records to be checked in a validation pass. Half of the records will be from the beginning of the **securityevent** table, and half will be from the end. If the number of records in the table is less

than the number specified by `max-records`, all records will be checked.

If `type` is set to `partial`, and `max-records` is not specified, eCare will validate 500 records.

If `type` is set to `full`, do not specify a `max-records` value.

EXAMPLE

The following sample configuration sets the eCare server to perform a full `securityevent` table validation at 1:30 AM daily.

```
<security-event-table-check>
  <validate-daily time="01:30">
    <table-check-event type="full"/>
  </validate-daily>
</security-event-table-check>
```

EXAMPLE

The following example enables both Startup Validation and Daily Validation. Note that both `<validate-on-startup>` and `<validate-daily>` require a child `<table-check-event>` element. `<table-check-event>` can be the same or different for the two validation options.

```
<security-event-table-check>
  <validate-on-startup>
    <table-check-event type="partial" max-records="100"/>
  </validate-on-startup>
  <validate-daily time="2:00">
    <table-check-event type="full"/>
  </validate-daily>
</security-event-table-check>
```

CONFIGURING ECARE SURVEYS

eCare's survey feature lets you monitor customer service levels and find out how your Support Agents are using eCare. As an eCare administrator, you have direct, immediate control over many aspects of surveys:

- Which surveys are offered, and when
- The number of questions

- The content of each question and the response options
- The appearance of the survey, through either the parent eCare style sheet or a special survey-only style sheet
- Where the results are sent

The survey results can even include dynamic content, such as trouble-ticket information or the Support Agent's name, by referencing JavaScript objects that are pre-populated by the eCare server.

Most survey configuration is performed during installation and in the eCare Administrator portal. See the *eCare Administrator's Guide* for information about the survey-management tasks you can perform there.

ADDING SERVICE NAMES TO SURVEYS

If your eCare server hosts multiple eCare services, you may wish to add the eCare service name to your survey results. This allows you to determine which eCare service a particular survey came from.

This configuration requires access to the *ecare.xml* file. Therefore, it cannot be performed with the eCare Administrator interface.

TO ADD THE ECARE SERVICE NAME TO A SURVEYS

1. Add the following configuration text immediately before the closing `</configuration>` tag at the end of the overrides file for your eCare service. The overrides file for a particular service is located at

```
/usr/local/resin/ecare4/overrides/<servicename>-ecare.xml
<ticket-queue>
  <queue-name action="replace">[service]</queue-name>
</ticket-queue>
```

Then save your changes and close the file.

2. Open your Web browser and download the following ZIP file from your eCare service.

```
http://<ecare-server>/<service-name>/ecare4/templates.zip
```

This file includes the default HTML files, JSP file, and CSS file that control eCare surveys.

3. In any text editor, open the HTML include file for the survey to which you wish to add the service name.

4. Add the following line to the top of the `<form>` section in the file.


```
<input type=hidden name="info_service" value="<ecare:
ConfigValue element='ticket-queue.queue-name' />">
```
5. Save and close the file.
6. Repeat steps 3-5 for any other surveys to which you wish to add the service name.
7. Sign in to the eCare Administrator portal. Open the Preferences Manager and use the *Upload* tab to upload the changed HTML files to your eCare server. See [“Upload: Uploading Custom Files”](#) in the *eCare Administrator’s Guide* for details.
8. Restart your eCare service for these changes to become effective.

SETTING UP IP BLOCKING

In some eCare installations, you may wish to prevent certain computers or IP addresses (or all computers except those you specifically allow) from accepting trouble tickets or from submitting eCare trouble tickets to the queue.

For example, you may wish to restrict access to the Support Agent portal to the IP addresses used by your Support Agents, which prevents anyone else from accepting trouble tickets. Or you may wish to prevent your Support Agents from submitting trouble tickets, which could then be accepted by other Support Agents.

SPECIFYING SUPPORT AGENT IP ADDRESSES

The following procedure allows you to specify the IP addresses from which Support Agents are allowed or not allowed to sign in to the eCare service.

TO ALLOW OR RESTRICT THE IP ADDRESSES THAT MAY SIGN IN

1. In a text editor, open the `[service]-ecare.xml` file, located in the `/usr/local/resin/ecare4overrides` directory for the service you want to edit.

- For the **agent-login** entry portal, specify the IP addresses to allow or block using the `<allow>` or `<deny>` property. For example,

```
<portal name="agent-login">
  <role>agent</role>
  <login-form>loginform.jsp</login-form>
  <required-permission>ecare:conversation.
    ecareconversation.agent</required-permission>
  <hours>open</hours>
  <wtp-host-provider>
    <plugin>com.netopia.app.ecare.plugins.
      DefaultWTPHostProvider</plugin>
  </wtp-host-provider>
  <ip-restriction>
    <allow>33.23.20.0/5</allow>
    <allow>33.23.20.76</allow>
    <allow>33.23.48.0/8</allow>
  </ip-restriction>
</portal>
```

This will allow the following addresses to accept trouble tickets.

- 33.23.20.0 netmask 255.255.255.224 (5 zero bits on the end)
- 33.23.20.76 netmask 255.255.255.255
- 33.23.48.0 netmask 255.255.255.0 (8 zero bits on the end)

You must use only the `<allow>` property or only the `<deny>` property, not both. If both the `<allow>` and `<deny>` options are in use, a Java Servlet Exception error will result and the eCare service will fail to restart, making it inaccessible to both customers and Support Agents.

You may create as many instances of the `<allow>` or `<deny>` element as you need. Each IP address or IP address block must be specified in a separate `<allow>` or `<deny>` element.

- Save the `[service]-ecare.xml` file and restart your eCare server.

SPECIFYING REMOTE USER IP ADDRESSES

The following procedure allows you to specify the IP addresses that are allowed or not allowed to submit trouble tickets.

To ALLOW OR RESTRICT THE IP ADDRESSES THAT MAY SUBMIT TROUBLE TICKETS

1. In a text editor, open the *[service]-ecare.xml* file, located in the */usr/local/resin/ecare4/overrides* directory for the service you want to edit.
2. For the **client-login** entry portal, specify the IP addresses to allow or block using the `<allow>` or `<deny>` property. For example,

```
<portal name="client-login" default="true">
  <role>client</role>
  <login-form auto="true">GuestLogin.jsp</login-form>
  <required-permission>ecare:conversation.
    ecareconversation.client</required-permission>
  <hours>open</hours>
  <wtp-host-provider>
    <plugin>com.netopia.app.ecare.plugins.
      DefaultWTPHostProvider</plugin>
  </wtp-host-provider>
  <ip-restriction>
    <deny>33.23.20.0/5</deny>
    <deny>33.23.20.76</deny>
    <deny>33.23.48.0/8</deny>
  </ip-restriction>
</portal>
```

This will prevent the following addresses from submitting trouble tickets.

- 33.23.20.0 netmask 255.255.255.224 (5 zero bits on the end)
- 33.23.20.76 netmask 255.255.255.255
- 33.23.48.0 netmask 255.255.255.0 (8 zero bits on the end)

You must use only the `<allow>` property or only the `<deny>` property, not both. If both the `<allow>` and `<deny>` options are in use, a Java Servlet Exception error will result and the eCare service will fail to restart, making it inaccessible to both customers and Support Agents.

You may create as many instances of the `<allow>` or `<deny>` element as you need. Each IP address or IP address block must be specified in a separate `<allow>` or `<deny>` element.

3. Save the *[service]-ecare.xml* file and restart your eCare server.

SETTING UP A DUAL-HOMED SERVER

When an eCare server supports two different networks that cannot be joined, usually for security reasons, the eCare server requires two network interface cards (NICs). Although eCare supports dual homing, the installation script does not. You must manually configure the eCare server to accept dual homing.

Note that in the following example,

- Six unique portals are created: one portal each for customers, Support Agents, and administrators who are connecting from inside your network, and one portal each for customers, Support Agents, and administrators who are connecting from outside your network. If you do not wish to create some of these portals—for example, you do not want to allow entry to Support Agents or administrators outside your network—do not include the `<mapping>` and `<portal>` blocks for those portals.
- Customers, Support Agents, and administrators inside your network are directed to the default WTP comm address (part of the `<swtpp>` configuration element in the `ecare.xml` and overrides files). If you copy this sample configuration, be sure to set the default WTP comm address to that used by *internal* users.

TO CONFIGURE THE ECARE SERVER FOR DUAL HOMING

1. Open the `[service]-ecare.xml` file. This overrides file can be found in the directory
`/usr/local/resin/ecare4overrides`
2. Find the configuration block enclosed by the `<entry-portals>` tags. Remove the entire contents between these tags and replace it with the text below.

When you enter the new configuration, replace the placeholders `[external_ip]` and `[internal_ip]` with appropriate values for your eCare server.

```
<entry-portals action="replace">

<shortcuts>
<mapping match="full-url">
  <url-pattern>http://[external_ip]/[service]/ecare4/
  </url-pattern>
  <portal-name>external-client-login</portal-name>
</mapping>
```

```

<mapping match="full-url">
  <url-pattern>http://[external ip]/[service]/ecare4/
    agent/</url-pattern>
  <portal-name>external-agent-login</portal-name>
</mapping>

<mapping match="full-url">
  <url-pattern>http://[external ip]/[service]/ecare4/
    admin/</url-pattern>
  <portal-name>external-admin-login</portal-name>
</mapping>

<mapping match="full-url">
  <url-pattern>http://[internal ip]/[service]/ecare4/
    admin/</url-pattern>
  <portal-name>internal-admin-login</portal-name>
</mapping>

<mapping match="full-url">
  <url-pattern>http://[internal ip]/[service]/ecare4/
    agent/</url-pattern>
  <portal-name>internal-agent-login</portal-name>
</mapping>

<mapping match="full-url">
  <url-pattern>http://[internal ip]/[service]/ecare4/
    </url-pattern>
  <portal-name>internal-client-login</portal-name>
</mapping>
</shortcuts>

<portal name="external-client-login" default="true">
  <role>client</role>
  <login-form auto="true">GuestLogin.jsp</login-form>
  <required-permission>ecare:conversation.
    ecareconversation.client</required-permission>
  <hours>open</hours>
  <wtp-host-provider>
    <plugin>com.netopia.app.ecare.plugins.
      StaticHostProvider</plugin>
    <server-url>https://[external_ip]/wtp</server-url>
  </wtp-host-provider>
  <download-policy>always</download-policy>
</portal>

```

```

<portal name="external-agent-login">
  <role>agent</role>
  <login-form>loginform.jsp</login-form>
  <required-permission>ecare:conversation.
    ecareconversation.agent</required-permission>
  <hours>open</hours>
  <wtp-host-provider>
    <plugin>com.netopia.app.ecare.plugins.
      StaticHostProvider</plugin>
    <server-url>https://[external_ip]/wtp</server-url>
  </wtp-host-provider>
  <download-policy>always</download-policy>
</portal>

<portal name="external-admin-login">
  <role>admin</role>
  <login-form>loginform.jsp</login-form>
  <required-permission>ecare:superadmin</required-
    permission>
  <hours>open</hours>
  <wtp-host-provider>
    <plugin>com.netopia.app.ecare.plugins.
      StaticHostProvider</plugin>
    <server-url>https://[external_ip]/wtp</server-url>
  </wtp-host-provider>
  <download-policy>always</download-policy>
</portal>

<portal name="internal-client-login">
  <role>client</role>
  <login-form auto="true">GuestLogin.jsp</login-form>
  <required-permission>ecare:conversation.
    ecareconversation.client</required-permission>
  <hours>open</hours>
  <wtp-host-provider>
    <plugin>com.netopia.app.ecare.plugins.
      DefaultWTPHostProvider</plugin>
  </wtp-host-provider>
  <download-policy>always</download-policy>
</portal>

<portal name="internal-agent-login">
  <role>agent</role>
  <login-form>loginform.jsp</login-form>
  <required-permission>ecare:conversation.
    ecareconversation.agent</required-permission>
  <hours>open</hours>

```

```

<download-policy>always</download-policy>
<wtp-host-provider>
  <plugin>com.netopia.app.ecare.plugins.
    DefaultWTPHostProvider</plugin>
</wtp-host-provider>
</portal>

<portal name="internal-admin-login">
  <role>admin</role>
  <login-form>loginform.jsp</login-form>
  <required-permission>ecare:superadmin</required-
    permission>
  <hours>open</hours>
  <download-policy>always</download-policy>
  <wtp-host-provider>
    <plugin>com.netopia.app.ecare.plugins.
      DefaultWTPHostProvider</plugin>
  </wtp-host-provider>
</portal>

</entry-portals>

```

Save your changes to the *[service]-ecare.xml* overrides file and restart your eCare service. Note that your eCare service will respond to both the internal and external addresses only after it has been restarted.

ENABLING AND CONFIGURING eCARE FEATURES

In certain environments you may wish to enable certain eCare features that are not enabled by default, or you may wish to configure features that cannot be modified through the administrator interface. eCare provides a built-in mechanism for you to accomplish this by editing the *ecare.xml* file or the overrides file for the particular service that you want to modify.

In most cases, it is best to modify the overrides file. To do so, open the *[service]-ecare.xml* file in a text editor. This overrides file can be found in the following directory for the service you want to manage.

```
/usr/local/resin/ecare4overrides
```

Note: Problems with the *ecare.xml* or overrides file may prevent your eCare service from starting correctly. Before making any changes to the *ecare.xml* file, be sure to make a backup copy in case you need to restore it later.

ENABLING THE CONTROL AS ADMIN SERVICE

The Control As Admin service allows your Support Agents to launch a Control session *with Administrator privileges* on a remote Windows Vista computer. Running with Administrator privileges allows the eCare remote-control component far greater control over the Windows Vista computer.

To enable the Control As Admin service, make sure the `<show-agent-esc-control-menu>` element is present and set to `true` in the overrides file for your eCare service. (In most cases, the element is present but set to `false`.)

```
<desktop-assist>
  ...
  <show-agent-esc-control-menu>true</show-agent-esc-
    control-menu>
</desktop-assist>
```

Restart your eCare service to complete the configuration.

ENABLING SESSION RECORDING

For security, auditing, and quality assurance purposes, eCare allows you to save and play back recordings of all eCare screen sharing sessions. To enable session recording for an eCare service, make sure the following configuration is present and enabled in the overrides file for that service. (In most cases, the configuration is present but set to `never`.)

```
<desktop-assist>
  <audit action="replace">always</audit>
  <audit-path action="replace">/usr/local/resin/webapps/
    [service]/archive</audit-path>
</desktop-assist>
```

Restart your eCare service to begin recording screen-sharing sessions. Links to the session recording files will not appear in the Ticket Archive window of the eCare Reporting Center until you restart the eCare **page server**.

ENABLING AUDIBLE ALERTS

eCare can be configured to sound an alert (a bell sound) for different eCare events. Each of these events may be individually enabled. The available alerts are

New ticket `ecare:audibleAlert.ticket.new`

Escalated ticket `ecare:audibleAlert.ticket.escalated`

Remote customer chat message

`ecare:audibleAlert.chat.client`

Connection lost `ecare:audibleAlert.lostconnection`

By default, audible alerts for new tickets are active while the remaining audible alerts are disabled. The other alerts are disabled by their presence within a set of `<disabled-features>` tags in the service's overrides file. Therefore, to enable other audible alerts for an eCare service, you will *comment out* the alert features that you want to enable.

In the following example, audible alerts have also been activated for escalated tickets in addition to the default alerts for new tickets.

```
<disabled-features>
  <!--<permission>ecare:audibleAlert.ticket.new
    </permission> -->
  <!--<permission>ecare:audibleAlert.ticket.
    escalated</permission> -->
  <permission>ecare:audibleAlert.chat.client</permission>
  <permission>ecare:audibleAlert.lostconnection
    </permission>
</disabled-features>
```

Restart your eCare service to complete the configuration.

CONFIGURING CUSTOM MESSAGE DISPLAY

The eCare administrator can configure a custom message that will appear at the top of the session transcript panel in the eCare session window. The content and appearance of this message can be changed by editing or replacing a file in the *custom* directory.

The custom message can be displayed to either the Support Agent or the customer, or to both. (The message will be different for Support Agents and customers unless you create identical message files.)

TO CREATE A MESSAGE FOR THE CUSTOMER

1. Copy the *exampleTranscriptPlacard.jsp* file from the *brand/client/* directory to the *custom/client* directory.
2. Rename the file to *clientTranscriptPlacard.jsp*.

3. Open the JSP file and edit the text and formatting of your message as desired. By default, the message will appear as red text on a yellow background. However, you may edit the appearance of your message in any way you wish.

When you are done, save the JSP file. You do not need to restart your eCare service.

The message will now appear in the eCare session window. The standard components of the session window will adjust their position as needed.

TO CREATE A MESSAGE FOR THE SUPPORT AGENT

1. Copy the *exampleTranscriptPlacard.jsp* file from the *brand/helper/* directory to the *custom/helper* directory.
2. Rename the file to *helperTranscriptPlacard.jsp*.
3. Open the JSP file and edit the text and formatting of your message as desired. By default, the message will appear as red text on a yellow background. However, you may edit the appearance of your message in any way you wish.

When you are done, save the JSP file. You do not need to restart your eCare service.

The message will now appear in the eCare session window. The standard components of the session window will adjust their position as needed.

CONFIGURING REPORT ORDER AND DISABLING REPORTS

You can also use the overrides file to control the order in which eCare report names appear in the eCare Reporting Center, rename reports, and remove specific default reports from the reporting interface.

Each eCare report is enabled and configured within a `<report>` tag.

```
<report>
  <report-id>agent-activity-2</report-id>
  <name>Daily Agent Activity Profile</name>
  <query-page>agentActivity2Query.jsp</query-page>
  <presentation-page>agentActivity2.jsp</presentation-
    page>
  <query-plugin>com.netopia.app.ecare.beans.reports.
    AgentActivity2QueryPlugin</query-plugin>
  <bean-plugin>com.netopia.app.ecare.beans.reports.
    AgentActivity2BeanPlugin</bean-plugin>
```

```

    <permissions>
      <permission>ecare:report</permission>
    </permissions>
  </report>

```

The report names appear in the eCare Reporting Center in the same order in which they appear in the *ecare.xml* file.

- To reorder reports, move the entire `<report>` section for each report to the desired location within the `<reports>` parent tag. Do not move any reports out of the `<reports>` section.
- To rename a report, edit the text within its associated `<name>` tag.
- To remove a report from the reporting interface, comment it out.

Restart your eCare service to complete the configuration.

DISABLING eCARE FEATURES

In certain environments you may wish to disable certain eCare functionality, particularly those features that you do not want available to your Support Agents. eCare provides a built-in mechanism for you to accomplish this by editing the *ecare.xml* file or the overrides file for the particular service that you want to modify.

In most cases, it is best to modify the overrides file. To do so, open the *[service]-ecare.xml* file in a text editor. This overrides file can be found in the following directory for the service you want to manage.

```

/usr/local/resin/ecare4overrides

```

Note: Problems with the *ecare.xml* or overrides file may prevent your eCare service from starting correctly. Before making any changes to the *ecare.xml* file, be sure to make a backup copy in case you need to restore it later.

DISABLING SCREEN-SHARING FUNCTIONALITY

By default, all screen-sharing features are enabled. However, a line for each feature permission is located within a set of `<disabled-features>` tags in the *ecare.xml* file. Each permission is within a commented-out section to prevent the feature from being disabled. Therefore, to disable a screen-sharing feature, you will copy or move its permission outside the commented-out section.

The following example disables the Share My View (Invite Observe) and Share My Control (Invite Control) features.

```
<desktop-assist>
  <disabled-features>
    <permission>agentInviteObserve</permission>
    <permission>agentInviteControl</permission>
    <!--
    <permission>agentChangeColorDepth</permission>
    <permission>agentObserve</permission>
    <permission>agentControl</permission>
    <permission>agentInviteObserve</permission>
    <permission>agentInviteControl</permission>
    -->
  </disabled-features>
```

The following screen sharing features can be disabled.

agentChangeColorDepth

Disabling this feature removes the Select Color Depth slider from the Support Agent interface. As a result, all screen sharing sessions will be run at the default color depth of 256 colors (8-bit color). If you wish to select another default color depth, change the following line located elsewhere within the *ecare.xml* file.

```
<default-bit-depth>color8
  </default-bit-depth>
```

Valid color depth values are **bw**, **gray2**, **gray4**, **gray8**, **color8**, and **lossless**.

- | | |
|---------------------------|---|
| agentObserve | Disabling this feature removes the View Remote User button from the Support Agent interface. |
| agentControl | Disabling this feature removes the Control Remote User button from the Support Agent interface. |
| agentInviteObserve | Disabling this feature removes the Share My View button from the Support Agent interface. |
| agentInviteControl | Disabling this feature removes the Share My Controls button from the Support Agent interface. |

Note: These screen sharing features continue to be available to Support Agents in the eCare interface until you restart your eCare service.

DISABLING EXAMINE SYSTEM

By default, the Examine System feature is enabled. However, two lines for the feature permission are located within a set of `<disabled-features>` tags in the *ecare.xml* file. The lines are within a commented-out section to prevent the feature from being disabled. Therefore, to disable the Examine System feature, you will copy or move the permission lines outside the commented-out section.

```
<system-analyzer>
  <disabled-features>
    <permission>agentAnalyze</permission>
    <permission>agentPrivilegedAnalyze</permission>
    <!--
    <permission>agentAnalyze</permission>
    <permission>agentPrivilegedAnalyze</permission>
    -->
  </disabled-features>
  <content-level>generic</content-level>
  <!-- generic, sensitive -->
</system-analyzer>
```

Note: The Examine System service continues to be available to Support Agents in the eCare interface until you restart your eCare service.

DISABLING FILE TRANSFER

By default, the File Transfer feature is enabled. However, two lines for feature permissions are located within a set of `<disabled-features>` tags in the *ecare.xml* file. Each permission is within a commented-out section to prevent the feature from being disabled. Therefore, to disable part or all of the File Transfer feature, you will copy or move the permission lines outside the commented-out section.

In the example below, only the Request File feature has been disabled.

```
<file-transfer>
  <disabled-features>
    <permission>clientFileTransfer</permission>
    <!--
    <permission>clientFileTransfer</permission>
    <permission>agentFileTransfer</permission>
    -->
  </disabled-features>
  <audit>>false</audit>
```

```

<!-- following path must be relative to the service
directory. -->
<temp-upload-path>WEB-INF/tmp</temp-upload-path>
<max-file-size>2</max-file-size> <!-- in megabytes -->
<file-life-span>1</file-life-span> <!-- number of hours
before file will be deleted -->
<ssl-mode>>false</ssl-mode>
</file-transfer>

```

The following file-transfer features can be disabled. Disabling both features will cause the entire *Files* tab to be removed from the Support Agent interface.

clientFileTransfer Disabling this feature removes the *Request File* button from the Support Agent interface.

agentFileTransfer Disabling this feature removes the *Send File* button from the Support Agent interface.

Note: File transfer services continue to be available to Support Agents in the eCare interface until you restart your eCare service.

By default, eCare's file transfer facility has a file size limit of 2MB. You can increase this limit with the `<max-file-size>` element. In the example below, the file size limit has been set to 30MB.

```

<max-file-size>30</max-file-size> <!-- in megabytes -->

```

DISABLING PUSH URL AND SUPPORT AGENT TOOLS

By default, the Push URL and Support Agent Tools features are enabled. However, several lines for feature permissions are located within a set of `<disabled-features>` tags in the *ecare.xml* file. Each permission is within a commented-out section to prevent the feature from being disabled. Therefore, to disable any of the Push URL or Support Agent Tools features, you will copy or move the permission lines outside the commented-out section.

In the example below, the Push URL and Email Transcript features have been disabled.

```

<ticket>
  <disabled-features>
    <permission>subscribe</permission>
    <permission>pushUrl</permission>
    <!--
    <permission>subscribe</permission>
    <permission>pushUrl</permission>

```

```

    <permission>pushUrlShortcuts</permission>
    <permission>agentTools</permission>
    -->
</disabled-features>

```

The following features can be disabled.

subscribe	Disabling this feature removes the <i>Email Transcript</i> button from both the customer and Support Agent interfaces. Disable this feature if you do not want customers or Support Agents to be able to request emailed session transcripts.
pushUrl	Disabling this feature removes the <i>Select a Shortcut URL</i> drop-down list and the <i>Push URL</i> text field and button from the Support Agent interface.
pushUrlShortcuts	Disabling this feature removes the <i>Select a Shortcut URL</i> drop-down list from the Support Agent interface. However, Support Agents can still push a URL to the customer by manually entering a Web address in the text entry area next to the <i>Push URL</i> button.
agentTools	Disabling this feature removes both the <i>View URL</i> button and the <i>Support Agent Tools</i> drop-down list from the Support Agent interface.

DISABLING CHAT

By default, the Chat features are enabled. However, several lines for feature permissions are located within a set of `<disabled-features>` tags in the *ecare.xml* file. Each permission is within a commented-out section to prevent the feature from being disabled. Therefore, to disable Chat shortcuts or the entire Chat feature, you will copy or move the permission lines outside the commented-out section.

In the following example, the *Select a Shortcut Message* drop-down list, which allows the Support Agent to send preset Chat messages, has been disabled.

```

<chat>
  <disabled-features>
    <permission>chatShortcuts</permission>
    <!--
    <permission>chat</permission>
    <permission>chatShortcuts</permission>

```

```

-->
</disabled-features>

<audit>true</audit>
<transcript-from-address>UNCONFIGURED
</transcript-from-address>
</chat>

```

The following Chat features can be disabled.

- | | |
|----------------------|---|
| chat | Disabling this feature removes the Chat text entry area and <i>Send</i> button from the both the customer and Support Agent interface. Disabling Chat will also automatically disable the <i>Select a Shortcut Message</i> drop-down list.

Do not disable this functionality unless your customers and Support Agents will be able to communicate in some other way, such as over the telephone. |
| chatShortcuts | Disabling this feature removes the <i>Select a Shortcut Message</i> drop-down list from the Support Agent interface. Support Agents can still chat by manually entering messages in the Chat text entry area. |

Note: Chat features continue to be available to Support Agents in the eCare interface until you restart your eCare service.

DISABLING MANAGED SCRIPTING

You may also configure your eCare service to disable Managed Scripting entirely. For information about configuring eCare with deployables, see [“Configuring eCare Deployables” on page 64](#).

In addition, Managed Scripting will not be available without *component trust*, which requires a valid Trust Certificate and Signing Key. See the *eCare Administrator’s Guide* for more information about component trust.

CHAPTER 4: CONFIGURING eCARE DEPLOYABLES

With eCare version 5.2, eCare is beginning a transition to a new model for managing the eCare components and controls that are installed on a computer to allow full use of eCare’s various services. This model uses *capabilities* and *deployables* to manage these components.

A computer’s *capabilities* indicate whether it can perform the tasks that may be required during an eCare session. When the eCare server recognizes that a computer meets certain specified criteria, the computer is said to possess that capability.

Deployables resolve and enable capabilities by managing the eCare components and controls that are downloaded to customer and Support Agent Web browsers.

In general, eCare’s key deployables are used to enable a computer’s key capabilities—its ability to use its installed software and eCare components to work with eCare’s screen-sharing and scripting services.

The capability-deployable model allows for greater flexibility and control over the installation and use of eCare components; you can control when (and if) specific eCare components are installed on the computers that use your eCare service. When certain components are not present on a given computer, the eCare services that use them are not available to that computer.

eCare deployables are primarily configured in the *deployables.xml* file, which is managed by Motorola. You should not need to edit this file. However, you can control several aspects of deployable behavior with the *ecare.xml* file (or the overrides file). The elements that manage this behavior are discussed further in the rest of this chapter.

CAPABILITIES

The operations that eCare can perform on any given computer depend on a variety of factors: the computer's operating system, its Web browser, and the eCare components that may have already been installed.

In the default eCare configuration, each time a computer begins an eCare session, the eCare server *resolves* the capabilities associated with the type of eCare user—customer or Support Agent. Whether or not eCare attempts to enable a missing capability depends on configuration options that you can control with the *ecare.xml* file. (You may also configure eCare to resolve capabilities only on an as-needed basis, or not at all.)

To resolve capabilities, eCare uses objects called *deployables*. When you configure your eCare server, the *deployables.xml* file will provide it with descriptions of all available eCare deployables. eCare's deployables are discussed further in [“Deployables” on page 66](#).

CAPABILITIES IN ECARE 5.2.1

eCare 5.2.1 recognizes two primary capabilities—`SharingReady` and `RemoteScriptingReady`. In the default eCare configuration, the eCare server attempts to resolve both of these capabilities when a computer first connects to the eCare server.

SHARINGREADY

The `SharingReady` capability indicates whether the computer has the proper software and eCare components to use eCare's screen-sharing services. The `SharingReady` capability does not recognize or indicate whether the computer can use the Java-based remote-control component, the native-platform component, or both. However, a computer with the `SharingReady` capability is ready to make and accept connections with eCare's screen-sharing services.

REMOTESCRIPTINGREADY

The `RemoteScriptingReady` capability indicates whether the computer has the proper software and eCare components to use eCare's Managed Scripts. The `RemoteScriptingReady` capability indicates that the computer is running the

required version of Java to support Managed Scripts, and that the correct eCare Java component is installed.

RESOLVING CAPABILITIES

eCare deployables use a recursive algorithm to resolve capabilities. The algorithm begins at the root node of a deployable tree and “walks” the tree until each prerequisite capability of the root deployable has been resolved, or until it encounters a prerequisite capability that cannot be satisfied.

When a deployable’s prerequisites have all been met, eCare runs the deployable to resolve its goal capabilities. If a deployable’s prerequisites cannot be satisfied, the deployable does not run.

DEPLOYABLES

eCare *deployables* are the objects that manage the eCare components and controls that are downloaded to customer and Support Agent Web browsers.

The key deployables in eCare 5.2.1 control the installation and use of the eCare remote-control component and the ScriptRunner applet, which executes Managed Scripts on the remote computer.

DEPLOYABLES IN eCARE 5.2.1

eCare 5.2.1 uses three main deployables—**enable-rc**, **install-java-components**, and **enable-native-rc**. In the default eCare configuration, the eCare server uses the **enable-rc** deployable when it attempts to resolve the `SharingReady` and `RemoteScriptingReady` capabilities.

ENABLE-RC

The **enable-rc** deployable is the default deployable that enables eCare’s screen-sharing services. It supports both Java-based and native-platform remote-control components. Java-based screen-sharing uses a Java applet, while the native-platform component is an ActiveX control for Windows computers and a plugin for Macintosh computers.

The **enable-rc** deployable attempts to enable the `SharingReady` and `RemoteScriptingReady` capabilities by installing the Java-based remote-control component. If this installation fails, the **enable-rc** deployable attempts to install the appropriate native-platform remote-control component. The native-platform component will allow screen-sharing sessions, but it does not support Managed Scripting.

INSTALL-JAVA-COMPONENTS

The **install-java-components** deployable supports screen-sharing services with a Java-based remote-control component.

The **install-java-components** deployable attempts to enable the `SharingReady` and `RemoteScriptingReady` capabilities by installing only the Java-based remote-control component. If this installation fails, no other component installation is attempted.

ENABLE-NATIVE-RC

The **enable-native-rc** deployable supports screen-sharing services with a native-platform remote-control component.

The **enable-native-rc** deployable attempts to enable the `SharingReady` capability by installing only the native-platform remote-control component (an ActiveX control for Windows computers and a plugin for Macintosh computers). If this installation fails, no other component installation is attempted.

If you enable only the **enable-native-rc** deployable, computers that use your eCare service will require no Java. However, the **enable-native-rc** deployable does not support eCare's Managed Scripts feature.

CONFIGURING DEPLOYABLE POLICIES

The use and execution of deployables within your eCare service is controlled by the `deployables.xml` and `ecare.xml` configuration files. The `deployables.xml` file is maintained by Motorola. In general, you should not edit this file directly, as it describes the set of deployables that are available within the current version of eCare release.

You will use the `<deployable-policies>` section in the *ecare.xml* file (or, more commonly, in the equivalent overrides file) to control which deployables are run, and when, during the lifetime of an eCare session.

The `<deployable-policies>` section is a child element of the top-level `<desktop-assist>` element. `<deployable-policies>` contains three subsections that you can use to control eCare's use of deployables:

- A `<deployable-plan>` element named `login`
- A `<deployable-plan>` element named `user-generated-event`
- A list of `<deployable-class>` elements.

THE LOGIN DEPLOYABLE PLAN

The `login <deployable-plan>` element controls which capabilities eCare will resolve when a computer connects to your eCare service—when a customer loads the trouble-ticket submission page in their Web browser, or when the Support Agent signs in to the trouble-ticket queue. It also specifies which deployables the server will use to resolve them.

The `<deployable-plan>` element consists of three optional child elements—`<client>`, `<agent>`, and `<admin>`. These three child elements, or *user role* elements, describe the capabilities that the eCare server will attempt to resolve when a customer, Support Agent, or eCare administrator connects to eCare. Each user role element consists of zero or more `<capability>` elements.

Each `<capability>` element includes a `name` attribute, which names the capability it will resolve. It also includes a required `<deployable>` child element, which specifies the root deployable eCare will use to resolve the capability.

The following XML snippet shows a `<deployable-plan>` element with `<client>` and `<agent>` child elements.

```
<deployable-plan name="login">
  <client>
    <capability name="SharingReady">
      <deployable>enable-native-rc</deployable>
    </capability>
    <capability name="RemoteScriptingReady">
      <deployable>install-java-components</deployable>
    </capability>
  </client>
```

```

    <agent>
      <capability name="SharingReady">
        <deployable>enable-rc</deployable>
      </capability>
    </agent>
  </deployable-plan>

```

In this example, the `<client>` element specifies that when a customer connects to eCare, the server will attempt to resolve the capabilities `SharingReady` and `RemoteScriptingReady`. To resolve `SharingReady`, eCare will use the **enable-native-rc** deployable. To resolve `RemoteScriptingReady`, eCare will use the **install-java-components** deployable.

The `<agent>` element specifies that when a Support Agent signs in, the eCare server will attempt to resolve the `SharingReady` capability. Unlike with the `<client>` element, however, eCare will use the **enable-rc** deployable (instead of **enable-native-rc**) to resolve the `SharingReady` capability.

THE USER-GENERATED-EVENT DEPLOYABLE PLAN

The `<deployable-plan>` element named `user-generated-event` specifies the capabilities that can be resolved on a deferred basis if they were not resolved when the computer first connected to eCare. The eCare server does not resolve the capabilities listed in the `user-generated-event` plan at any predetermined time. Instead, the server attempts to resolve the relevant capability in response to a user-generated event, such as the Support Agent initiating a remote-control session with the customer.

In eCare 5.2.1 there are two capabilities that can be applicable in the `user-generated-event` deployable plan: `SharingReady` and `RemoteScriptingReady`.

The XML syntax for controlling the `user-generated-event` deployable plan is the same as for the `login` deployable plan.

THE DEPLOYABLE-CLASS CONFIGURATION ELEMENTS

Every eCare deployable belongs to a *deployable class*. When eCare runs the deployable, it checks its `<deployable-class>` configuration element in the `ecare.xml` file (or the overrides file) and applies the appropriate policies to the deployable.

`<deployable-class>` policies are specified as name-value parameter pairs. Generally, the eCare server forwards these policies to the deployable as data; in most cases, only the deployables themselves are able to interpret the contents of these parameters.

Many `<deployable-class>` policies specify version information for the deployable. For example, the `JavaProbe` deployable class specifies the minimum version of Java that must be enabled in the computer's Web browser before the deployable will report that Java is present.

```
<deployable-class name="JavaProbe">
  <parameters>
    <param name="minVersion">1.4.2_09</param>
  </parameters>
</deployable-class>
```

In most cases, you will not need to edit version information. However, in certain situations it may be useful to do so. If, for example, your eCare service has experienced problems with a certain version of the eCare remote-control component, it may be possible to exclude that version by changing the version number of the associated deployable class.

CONFIGURING SPECIFIC DEPLOYABLES TO INSTALL AT LOGIN

The `login <deployable-plan>` element controls which capabilities eCare will resolve when a computer signs in to your eCare service, as well as which deployables the server will use to resolve them.

If you remove a `<capability>` child element from within the `login <deployable-plan>` element, the associated capability will not be resolved when the computer signs in to eCare. However, the capability may still be installed later in the eCare session, on an as-needed basis, if you place the capability in the user-generated-event `<deployable-plan>` element. See [“Configuring Deferred Installation of Deployables” on page 77](#) for more information.

CONFIGURING INSTALLATION OF THE REMOTE-CONTROL COMPONENT

Before a computer can use eCare's screen-sharing services, it must possess the top-level `SharingReady` capability. You can configure your eCare server to install this capability when the computer first connects to the eCare service. For customer computers, this occurs when the customer loads the trouble-ticket submission page in their Web browser. For Support Agent computers, it occurs when the Support Agent enters their eCare user name and password and signs in to the trouble-ticket queue.

Your eCare server can

- Preferentially install the Java-based remote-control component. If the Java component cannot be installed, the eCare server will install the native-platform remote-control component—the eCare ActiveX control for Windows computers, and the eCare plugin for Macintosh computers. This is the default eCare configuration.
- Install only the Java-based remote-control component.
- Install only the native-platform remote-control component.

These options are configured per user role—for example, you can configure eCare to install only the native-platform component on customer computers, but install the Java-based component, with native-platform as a fallback, on Support Agent computers. The eCare server fully supports remote-control connections between the Java and native-platform component, as well as between the Windows ActiveX control and the Macintosh plugin.

PREFERENTIALLY INSTALLING THE JAVA REMOTE-CONTROL COMPONENT

In the default eCare installation, the eCare server supports screen-sharing services with both Java-based and native-platform remote-control components.

The **enable-rc** deployable attempts to enable the `SharingReady` capability by installing the Java-based remote-control component. If this installation fails, the **enable-rc** deployable attempts to install the appropriate native-platform remote-control component.

To configure eCare to preferentially install the Java remote-control component when a computer first connects to the eCare server, the `SharingReady` capability in your *ecare.xml* file (or the overrides file) must include an **enable-rc** `<deployable>` element.

To apply this capability to the computers belonging to your eCare customers, the capability must reside within the `<client>` element in the `login` deployable plan. To apply it to Support Agent computers, it must reside within the `<agent>` element.

```
<desktop-assist>
...
<deployable-policies>
  <deployable-plan name="login">
    <client>
      <capability name="SharingReady">
        <deployable>enable-rc</deployable>
      </capability>
    </client>
  </deployable-plan>
  ...
</deployable-policies>
...
```

If `SharingReady` installation fails for both the Java and native-platform remote-control components, the eCare server cannot attempt to install the `SharingReady` capability again during the same eCare session. The computer will not be able to use screen-sharing services during that eCare session. (For more information about deferred installation, see [“Configuring Deferred Installation of Deployables” on page 77.](#))

INSTALLING ONLY THE JAVA REMOTE-CONTROL COMPONENT

In some cases, you may wish to allow computers on your eCare service to use only the Java-based remote-control component for screen-sharing sessions. The `install-java-components` deployable attempts to install only the Java-based remote-control component. If this installation fails, no other component installation is attempted.

To configure eCare to install only the Java remote-control component when a customer’s computer first connects to the eCare server, the `SharingReady` capability in your `ecare.xml` file (or the overrides file) must include an `install-java-components` `<deployable>` element.

To apply this capability to the computers belonging to your eCare customers, the capability must reside within the `<client>` element in the `login` deployable plan. To apply it to Support Agent computers, it must reside within the `<agent>` element.

```

<desktop-assist>
  ...
  <deployable-policies>
    <deployable-plan name="login">
      <client>
        <capability name="SharingReady">
          <deployable>install-java-
            components</deployable>
        </capability>
      </client>
    </deployable-plan>
    ...
  </deployable-policies>
  ...

```

If `SharingReady` installation fails for the Java remote-control component, it may still be possible for the eCare server to install the `SharingReady` capability later in the eCare session. (For more information about deferred installation, see [“Configuring Deferred Installation of Deployables” on page 77.](#))

INSTALLING ONLY THE NATIVE-PLATFORM REMOTE-CONTROL COMPONENT

In some cases, you may wish to allow computers on your eCare service to use only the native-platform remote-control component for screen-sharing sessions. The **enable-native-rc** deployable attempts to install only the native remote-control component—an ActiveX control for Windows computers, and a plugin for Macintosh computers. If this installation fails, no other component installation is attempted.

To configure eCare to install only the native-platform remote-control component when a customer’s computer first connects to the eCare server, the `SharingReady` capability in your *ecare.xml* file (or the overrides file) must include an **enable-native-rc** `<deployable>` element.

To apply this capability to the computers belonging to your eCare customers, the capability must reside within the `<client>` element in the `login` deployable plan. To apply it to Support Agent computers, it must reside within the `<agent>` element.

```

<desktop-assist>
  ...
  <deployable-policies>
    <deployable-plan name="login">
      <client>
        <capability name="SharingReady">
          <deployable>enable-native-rc</deployable>
        </capability>
      </client>
    </deployable-plan>
    ...
  </deployable-policies>
  ...

```

If `SharingReady` installation fails for the native-platform remote-control component, it may still be possible for the eCare server to install the `SharingReady` capability later in the eCare session. (For more information about deferred installation, see [“Configuring Deferred Installation of Deployables” on page 77.](#))

MIXING REMOTE-CONTROL COMPONENT TYPES

The eCare server fully supports remote-control connections between the Java and native-platform component, as well as between the Windows ActiveX control and the Macintosh plugin. Therefore, you may configure your eCare server to enable the `SharingReady` capability using different components for customer and Support Agent computers.

The following example will configure your eCare server to install only the native-platform remote-control component when a customer’s computer first connects to the eCare server. However, when a Support Agent signs in to eCare, the server will attempt to enable the `SharingReady` capability by installing the Java remote-control component, and falling back to the native-platform component if the Java-based installation fails.

```

<desktop-assist>
  ...
  <deployable-policies>
    <deployable-plan name="login">
      <client>
        <capability name="SharingReady">
          <deployable>enable-native-rc</deployable>
        </capability>
      </client>
      <agent>
        <capability name="SharingReady">
          <deployable>enable-rc</deployable>
        </capability>
      </agent>
    </deployable-plan>
    ...
  </deployable-policies>
  ...

```

Note that the `<client>` and `<agent>` configurations reside within the same `<deployable-plan>` element.

NO REMOTE-CONTROL COMPONENT DEPLOYMENT

Finally, you may configure your eCare server to make no attempt to enable the `SharingReady` capability when a computer connects to the eCare service. Instead, installation of the capability can be deferred (or even prevented). Deferred installation allows the eCare server to enable the `SharingReady` capability only if and when it is needed. (See [“Configuring Deferred Installation of Deployables” on page 77](#) for more information about this option.)

For example, if eCare does not install the remote-control component when a customer loads the trouble-ticket submission page in their Web browser, the customer can enter the queue immediately. No time is spent on the installation of components that may not be required for the customer’s eCare session. If the Support Agent determines that a screen-sharing service is required, the eCare server can enable the `SharingReady` capability at that time.

CONFIGURING INSTALLATION OF THE REMOTE SCRIPTING DEPLOYABLE

Before a Support Agent can execute a Managed Script on a customer's computer, the customer's computer must possess the `RemoteScriptingReady` capability.

To configure eCare to enable this capability when a computer first connects to the eCare server, the `RemoteScriptingReady` capability in your `ecare.xml` file (or the overrides file) must include an **install-java-components** `<deployable>` element. The capability must reside within the `<client>` element in the `login` deployable plan.

```
<desktop-assist>
  ...
  <deployable-policies>
    <deployable-plan name="login">
      <client>
        <capability name="SharingReady">
          <deployable>enable-native-rc</deployable>
        </capability>
        <capability name="RemoteScriptingReady">
          <deployable>install-java-
            components</deployable>
        </capability>
      </client>
      ...
    </deployable-plan>
    ...
  </deployable-policies>
  ...
```

If `RemoteScriptingReady` installation fails—for example, because the computer's user denies permission to install the Java applet—the eCare server cannot attempt to install the `RemoteScriptingReady` capability again during the same eCare session. The computer will not be able to use Managed Scripts during that eCare session. (For more information about deferred installation, see [“Configuring Deferred Installation of Deployables” on page 77.](#))

CONFIGURATION SUMMARY

The configuration options for `login` installation are summarized in the following table below.

Login Installation Options				
Capability	Applicable Roles	Enables	Deployable	Function
SharingReady	<client> , <agent>	using screen-sharing services with the remote-control component	enable-rc	installs the Java component, then the native component if Java installation fails
			install-java-components	installs only the Java component
			enable-native-rc	installs only the native-platform component
RemoteScriptingReady	<client>	running Managed Scripts	install-java-components	installs only the Java component

CONFIGURING DEFERRED INSTALLATION OF DEPLOYABLES

In certain cases, if the eCare server does not install a particular deployable when a customer's computer first connects to the eCare service, it can install it on an as-needed basis during an active eCare session. For example, if a Support Agent determines that a screen-sharing service is required, and the customer's computer does not have the `SharingReady` capability, the eCare server can enable it at that time.

The capabilities that can be enabled on a deferred basis are specified in the `user-generated-event <deployable-plan>` element. The syntax of this element is the same as that of the `login` deployable plan. (See [“Configuring Specific Deployables to Install at Login” on page 70](#) for more information about using the `login` deployable plan.)

There are two main differences between the `login` and `user-generated-event` deployable plans:

- The `login` deployable plan is invoked automatically when a computer connects to the eCare service; the `user-generated-event` deployable plan is invoked only in response to a user-generated event, such as when a Support Agent launches one of eCare's screen-sharing services.
- The `login` deployable plan can be used for both the `<client>` and `<agent>` user roles; the `user-generated-event` deployable plan is currently used only for the `<client>` user role. (Because Support Agents use eCare on a regular basis, they are assumed to have all required components at the beginning of any eCare session.)

In eCare 5.2.1, there are two occasions in which the eCare server will invoke the `user-generated-event` deployable plan:

- When the Support Agent attempts to launch one of eCare's screen-sharing services, and the `SharingReady` capability has not yet been enabled on the customer's computer.
- When the Support Agent attempts to execute a Managed Script on the customer's computer, and the `RemoteScriptingReady` capability has not yet been installed on the customer's computer.

Note that the eCare server will *not* invoke the `user-generated-event` deployable plan if the computer is already known to lack a prerequisite capability, such as a Java-enabled browser, that cannot be enabled on a deferred basis.

CONFIGURING DEFERRED INSTALLATION OF THE REMOTE-CONTROL COMPONENT

Before a computer can use eCare's screen-sharing services, it must possess the top-level `SharingReady` capability. Use the `user-generated-event` deployable plan to configure eCare to enable this capability on an as-needed basis.

In your `ecare.xml` file (or the overrides file), locate the `<client>` element in the `user-generated-event` deployable plan. Set the `<deployable>` element for the `SharingReady` capability to **enable-rc**, **install-java-components**, or **enable-native-rc**.

The following example demonstrates deferred installation of the `SharingReady` capability using the Java remote-control component preferentially.

```

<desktop-assist>
  ...
  <deployable-policies>
    <deployable-plan name="user-generated-event">
      <client>
        <capability name="SharingReady">
          <deployable>enable-rc</deployable>
        </capability>
      </client>
    </deployable-plan>
    ...
  </deployable-policies>
  ...

```

CONFIGURING DEFERRED INSTALLATION OF THE MANAGED SCRIPTING DEPLOYABLE

Before a computer can use eCare's Managed Scripts feature, it must possess the top-level `RemoteScriptingReady` capability. Use the `user-generated-event` deployable plan to configure eCare to enable this capability on an as-needed basis.

In your *ecare.xml* file (or the overrides file), locate the `<client>` element in the `user-generated-event` deployable plan. Set the `<deployable>` element for the `RemoteScriptingReady` capability to **install-java-components**.

```

<desktop-assist>
  ...
  <deployable-policies>
    <deployable-plan name="user-generated-event">
      <client>
        <capability name="RemoteScriptingReady">
          <deployable>install-java-
            components</deployable>
        </capability>
      </client>
    </deployable-plan>
    ...
  </deployable-policies>
  ...

```

The options for configuring deferred installation are summarized in the table below. These options are applicable to the <client> user role only.

Deferred Installation Options			
Capability	Enables	Deployable	Function
SharingReady	using screen-sharing services with the remote-control component	enable-rc	installs the Java component, then the native component if Java installation fails
		install-java-components	installs only the Java component
		enable-native-rc	installs only the native-platform component
RemoteScriptingReady	running Managed Scripts	install-java-components	installs only the Java component

UNDERSTANDING THE INTERACTION BETWEEN DEPLOYABLES

As you configure your eCare deployables, keep in mind that the capabilities and deployables you specify with the `login` and `user-generated-event` deployable plans can potentially interact in unexpected ways.

- Some deployables can enable more than one capability. Similarly, some capabilities may be enabled by more than one deployable.
- Some deployables can disable a capability for the rest of an eCare session, making deferred installation impossible.

For examples of each potential interaction, see the following sections.

ENABLING MULTIPLE CAPABILITIES WITH MULTIPLE DEPLOYABLES

Some deployables can enable more than one capability. For example, the `install-java-components` deployable can enable both the `SharingReady` and `RemoteScriptingReady` capabilities. If you configure your eCare server to enable the `RemoteScriptingReady` capability, the `install-java-components` deployable will also enable the `SharingReady` capability.

Similarly, some capabilities may be enabled by more than one deployable. For example, the Java remote-control component can be installed by both the **enable-rc** and **install-java-components** deployables. If you configure your eCare server to use the **enable-native-rc** deployable to enable the `SharingReady` capability, but you also enable the `RemoteScriptingReady` capability, the **install-java-components** deployable that installs it will also install the Java-based remote-control component. When a Support Agent initiates a screen-sharing session with the customer, the customer's computer will use the Java component instead of the expected native-platform component. (When both components are available, eCare uses the Java component.)

The following table summarizes the capabilities that are enabled by each eCare deployable.

eCare Deployables and Associated Capabilities	
Deployable	Capabilities Installed
enable-rc	<code>SharingReady</code> , <code>RemoteScriptingReady</code> (if Java component installation is successful), and one of the following: <code>JavaRC</code> , <code>ActiveXRC</code> (Windows), or <code>PluginRC</code> (Macintosh)
install-java-components	<code>RemoteScriptingReady</code> , <code>SharingReady</code> , <code>JavaRC</code>
enable-native-rc	<code>SharingReady</code> and either <code>ActiveXRC</code> (Windows) or <code>PluginRC</code> (Macintosh)

DISABLING CAPABILITIES WITH DEPLOYABLES

If any `<capability>` element within a deployable's configuration has the attribute `setnotcapable` set to `true` in the `deployables.xml` file, the deployable may potentially disable that capability for the remainder of the computer's eCare session.

This will occur if the deployable fails to satisfy its prerequisites. Even if it would be possible to install that capability on a deferred basis, eCare will not attempt to do so.

For example, the following configuration specifies that eCare will run the **enable-rc** deployable when the customer connects to the eCare service and the **enable-native-rc** deployable on a deferred basis.

```

<desktop-assist>
  ...
  <deployable-policies>
    <deployable-plan name="login">
      <client>
        <capability name="SharingReady">
          <deployable>enable-rc</deployable>
        </capability>
      </client>
      <agent>
        ...
      </agent>
    </deployable-plan>
    <deployable-plan name="user-generated-event">
      <client>
        <capability name="SharingReady">
          <deployable>enable-native-rc</deployable>
        </capability>
      </client>
    </deployable-plan>
    ...
  </deployable-policies>
  ...

```

Further, in the *deployables.xml* file, the **enable-rc** deployable contains the following `<capability>` element:

```

<capabilities>
  <capability setnotcapable="true">SharingReady
  </capability>
</capabilities>

```

In this example, the `setnotcapable` attribute is `true` for the `SharingReady` capability. Therefore, if the **enable-rc** deployable fails to satisfy its prerequisites—for example, if the computer’s user refused to allow the installation of any remote-control component—the eCare server will not attempt to install a remote-control component on a deferred basis. When the Support Agent selects the customer’s trouble ticket, all screen-sharing services will be disabled in the *Share* menu.

DISABLING MANAGED SCRIPTING

In the default eCare configuration, the eCare server attempts to enable the `RemoteScriptingReady` capability on the customer's computer when it first connects to the eCare server. For simplicity, the default `ecare.xml` file also configures the eCare server to enable the `RemoteScriptingReady` capability with the `user-generated-event` deployable plan.

If you wish to prevent all use of the Managed Scripting feature, remove the `RemoteScriptingReady` capability from both the `login` and `user-generated-event` deployable plans.

```
<desktop-assist>
...
<deployable-policies>
  <deployable-plan name="login">
    <client>
      <!-- <capability name="RemoteScriptingReady">
        <deployable>install-java-
          components</deployable>
      </capability> -->
    </client>
  </deployable-plan>
  <deployable-plan name="user-generated-event">
    <client>
      <!-- <capability name="RemoteScriptingReady">
        <deployable>install-java-
          components</deployable>
      </capability> -->
    </client>
  </deployable-plan>
  ...
</deployable-policies>
...
```

With this configuration, Managed Scripts are disabled even if eCare installs the Java-based remote-control component with the **enable-rc** deployable.

APPENDIX A: INSTALLING THE eCARE REMOTE-CONTROL COMPONENT

Before your Support Agents can use eCare's screen-sharing services, their computers will require the *eCare remote-control component*, which is a control that enables the computer to use these services. Your customers will also require the remote-control component. Windows computers use the eCare ActiveX control, while Macintosh computers use the eCare plugin.

By default, the Support Agent's Web browser automatically downloads the eCare remote-control component the first time the agent accesses your eCare Service Center. (The component is also updated automatically anytime a new version is installed on the eCare server.) However, you may also choose to pre-install the eCare ActiveX control on a Support Agent's Windows computer, particularly in either of the following situations:

- Your organization may have implemented policies or security measures that prevent users from downloading or installing ActiveX controls or plugins on Windows computers.
- Your Support Agents do not have Windows Administrator privileges for their computers.

To enable Internet Explorer to automatically install the eCare ActiveX control on Windows computers, the computer's user must have Windows Administrator privileges. Therefore, you may also need to deploy the eCare ActiveX control to those computers whose regular users do not have permission to download or install it.

This document discusses your options for pre-installing the eCare ActiveX control—using the MSI installer, downloading and registering the component file, or using your Windows Administrator privileges to pre-install the ActiveX control on a Support Agent's computer. You may also make the MSI installer available to your customers, so that they can install the eCare ActiveX control before entering the eCare system.

Note that Macintosh computers do not generally require special privileges to install the eCare plugin. However, you may pre-install the eCare plugin on Macintosh computers if you wish to do so. See [“Installing the eCare Remote-Control Component on Macintosh Computers”](#) on page 94.

INSTALLING THE JAVARC APPLET

You can pre-install the eCare JavaRC applet by logging in to the Support Agent’s computer and signing in to the eCare Support Agent portal. (You may also load the eCare customer portal.) The eCare system will automatically detect the missing Java applet and prompt you to install it.

TO INSTALL THE JAVARC APPLET

1. Sign in to the eCare Support Agent portal.
Your Web browser will display a window asking you to run the JavaRC applet.
2. To install the Java applet, click the *Run* button. The JavaRC applet will download and install automatically.

When the installation is complete, the main eCare session window appears. The Support Agent’s computer is now ready for the Support Agent to sign in.

INSTALLING THE eCARE REMOTE-CONTROL COMPONENT ON WINDOWS COMPUTERS

You may pre-install the eCare ActiveX control on both local and remote Windows computers.

BEFORE YOU INSTALL THE eCARE ACTIVEX CONTROL

On the Support Agent’s computer, the following options must be set in Internet Explorer.

- The eCare server must be set as a Trusted Site.
- JavaScript and ActiveX must be enabled.

- Java must be enabled, if the Support Agent will have eCare administrator privileges (Java is required to play session recordings, which are accessible only to administrators).
- The browser security level must be set to Medium or lower.
- Cookies must be allowed.

In addition, *all* pop-up blocking software *must be turned off* before the Support Agent can use eCare. This includes the pop-up blockers that are built in to Internet Explorer, as well as third-party blocking software for all platforms.

INSTALLING THE eCARE ACTIVE X CONTROL WITH THE MSI INSTALLER

If your organization restricts Windows Administrator privileges on Support Agent computers, your Support Agents cannot install the eCare ActiveX control when they sign in to your eCare Service Center. In this situation, the eCare ActiveX control must be installed before the Support Agent can begin working. With the MSI installer, you can quickly install the eCare ActiveX control on both local and remote computers. On local computers, you can run the installer directly. For remote computers, use a software deployment application. Or provide your Support Agents with the MSI file and the information they need to install it themselves.

The MSI installer may also enable your eCare customers to install the eCare ActiveX control before they submit an eCare trouble ticket. See [“Installing the eCare Remote-Control Component for Customers” on page 87](#).

If you require the MSI installer to pre-install the eCare ActiveX control, you may download it (along with other eCare documentation and FAQs) at the eCare Resource Page:

<http://www.netopia.com/support/software/ecare/>

Note that the MSI installer will not install the correct version of the eCare ActiveX control on Windows 98 and Windows ME computers. To pre-install the eCare ActiveX on these computers, see [“Installing the eCare ActiveX Control by Downloading and Registering Component Files” on page 90](#).

INSTALLING THE eCARE REMOTE-CONTROL COMPONENT FOR SUPPORT AGENTS

To use the MSI installer to pre-install the eCare ActiveX control on your Support Agent computers, use the following procedure.

TO INSTALL THE eCARE ACTIVE X CONTROL WITH THE MSI INSTALLER ON WINDOWS VISTA, WINDOWS XP, OR WINDOWS 2000

1. Copy the MSI file to your computer.
If you wish to install the eCare ActiveX control on other computers on your network, use a software deployment tool to distribute the MSI file to the target computers.
2. Run the MSI installer.
 - On the local computer, double-click the MSI file. The Setup Wizard will guide you through the installation process.
 - On the remote computer, use the software deployment tool to execute the following command.

```
msiexec /package eCareClient.msi /quiet
```

Be sure to use the `/quiet` switch to ensure that no user interaction is required.

The MSI installer will copy a DLL file to the `c:\eCare Windows Client` directory, and then register the DLL with Windows.

If you later wish to remove the eCare ActiveX control from your computer, use the Add or Remove Programs list in the Windows Control Panel. Windows will unregister the DLL and delete the DLL file and the `eCare Windows Client` directory.

INSTALLING THE eCARE REMOTE-CONTROL COMPONENT FOR CUSTOMERS

With the MSI installer, your eCare customers can install the eCare ActiveX control before they submit an eCare trouble ticket.

Upload the MSI file to your Web server and create a download link on your main support page or eCare entry page. Customers can download and run the MSI installer before they enter the eCare system.

In addition, for your customers who are using **Internet Explorer version 7 on Windows 2000 or Windows XP**, Motorola strongly recommends that you provide a script that will automatically set your eCare server as an Internet Explorer Trusted Site. See [“The Trusted Site Script” on page 88](#).

On the MSI download page, provide the following instructions.

TO INSTALL THE eCARE ACTIVE X CONTROL ON WINDOWS VISTA

1. Copy the MSI file to your computer.
2. Run the MSI installer by double-clicking the file. The Setup Wizard will guide you through the installation process.
3. Connect to eCare.

When you connect to eCare, the eCare server will prompt you to automatically download and run a script that will set the eCare server as a Trusted Site in your Web browser. For the best eCare experience, please allow this installation.

TO INSTALL THE eCARE ACTIVE X CONTROL FOR INTERNET EXPLORER VERSION 6 ON WINDOWS XP AND WINDOWS 2000

1. Copy the MSI file to your computer.
2. Run the MSI installer by double-clicking the file. The Setup Wizard will guide you through the installation process.
3. Connect to eCare.

TO INSTALL THE eCARE ACTIVE X CONTROL FOR INTERNET EXPLORER VERSION 7 ON WINDOWS XP AND WINDOWS 2000

1. Copy the MSI file and the *setTrustedSite.js* file to your computer.
2. Run the MSI installer by double-clicking the file. The Setup Wizard will guide you through the installation process.
3. Run the *setTrustedSite.js* file by double-clicking the file.
4. Connect to eCare.

The Trusted Site Script

For your customers who are using **Internet Explorer version 7 on Windows 2000 or Windows XP**, Motorola strongly recommends that you provide a script that will automatically set your eCare server as an Internet Explorer Trusted Site.

Enter the following script in a text file and save it as *setTrustedSite.js*.

```
//In HKCU
function setTrustedSite (domainName)
{
    var HKEY_CURRENT_USER = 0x80000001;
    var strValueName = "http";
    var dwValue = 2;
    var strComputer = ".";
    var objReg= GetObject("winmgmts:\\\\" + strComputer
    + "\\root\\default:StdRegProv");
    var strKeyPath = "Software\\Microsoft\\Windows\\
    Current Version\\Internet Settings\\ZoneMap\\
    Domains\\";

    strKeyPath += domainName;
    if (objReg.CreateKey(HKEY_CURRENT_USER, strKeyPath)
    != 0)
    {
        return "Failed: failed to create domain key";
    }
    if (objReg.SetDWORDValue(HKEY_CURRENT_USER,
    strKeyPath, strValueName, dwValue) != 0)
    {
        return "Failed: failed to create http key";
    }
    return "Success";
}

setTrustedSite ("eCaredomain.com");
```

Be sure to replace *eCaredomain.com* with your eCare domain in the last line of the script. For example, if your eCare server is accessed from

http://ecareserver.ecare.com/ecareservice

The domain is *ecare.com* and the last line of the script should be

```
setTrustedSite ("ecare.com");
```

Note that your customers who are using Internet Explorer version 6, or Internet Explorer version 7 with Windows Vista, do *not* need to use this script. Internet Explorer version 6 does not require the eCare server to be set as a Trusted Site. When customers with Windows Vista access eCare for the first time, the eCare server will prompt them to automatically download and run a script that creates the Trusted Site setting.

INSTALLING THE eCARE ACTIVEX CONTROL BY DOWNLOADING AND REGISTERING COMPONENT FILES

If you are not able or do not wish to install the eCare ActiveX control with the MSI installer, you may also download it and register its component DLL.

With this procedure, you will download the eCare ActiveX control source file, extract its components, and register the components with the Windows operating system. Once it has been registered, the ActiveX control is available for use.

With a software deployment application, you can also use this procedure to install the eCare ActiveX control on remote Support Agent computers. Or you can provide them with the information they need to do it themselves.

DOWNLOADING THE eCARE ACTIVEX CONTROL

Begin by downloading a local copy of the eCare ActiveX control.

TO DOWNLOAD THE eCARE ACTIVEX CONTROL

1. In a text file, enter the following lines.

```
<html>
<a href="http://<server>/<service>/ecare4/components/
  CobAgent_4.0_w98.cab">eCare ActiveX for Windows
98</a>
<a href="http://<server>/<service>/ecare4/components/
  CobAgent_4.2.1.318.cab">eCare ActiveX</a>
</html>
```

In place of `<server>` and `<service>`, enter the location of your eCare service. For example,

```
<a href="http://ecare.motorola.com/247service/ecare4/
  components/CobAgent_4.2.1.318.cab">eCare
ActiveX</a>
```

2. Save this file as an HTML file (for example, *activex.html*).
3. Open the file in Internet Explorer.
4. Right-click each link and choose *Save Target As* to save the ActiveX file to your computer.

Note: Internet Explorer may attempt to save the ActiveX file as an HTM file. Be sure to specify the correct CAB file extension and the *All Files* file type before you save the file.

INSTALLING AND REGISTERING THE eCARE ACTIVE X CONTROL

Now install and register the eCare ActiveX control. You may register it on the local computer, or deploy it to other computers on your network.

TO INSTALL AND REGISTER THE eCARE ACTIVE X CONTROL ON THE LOCAL COMPUTER

1. Download the CAB file for the eCare ActiveX control by following the above procedure.
2. Using a file-extraction utility, open the CAB file and extract the files that comprise the eCare remote-control component: *CobAgent4.dll* and *CobAgent4.inf* for Windows 98, and *CobAgent4_2_1_318.dll* and *CobAgent4_2_1_318.inf* for Windows XP.
3. Place the files in the `\WINNT\Downloaded Program Files` or `\WINDOWS\Downloaded Programs Files` folder.

Note that you cannot copy files to the Downloaded Program Files folder using Windows Explorer. You must copy them at the command line with the `copy` command.

4. On the command line, enter the following command to register the eCare ActiveX control with Windows Vista, Windows XP, and Windows 2000 computers.

```
regsvr32 /s C:\WINDOWS\Downloaded Program Files\  
CobAgent4_2_1_318.dll
```

For Windows 98 and Windows ME, use

```
regsvr32 /s C:\WINDOWS\Downloaded Program Files\  
CobAgent4.dll
```

The ActiveX control is now installed and registered for use with Windows. (Note that it will *not* appear in the Downloaded Program Files folder in Windows Explorer.)

TO DEPLOY THE eCARE ACTIVE X CONTROL TO REMOTE COMPUTERS

1. Download the CAB file for the eCare ActiveX control by following the above procedure.
2. Using a file-extraction utility, open the CAB file and extract the files that comprise the eCare remote-control component: *CobAgent4.dll* and *CobAgent4.inf* for Windows 98, and *CobAgent4_2_1_318.dll* and *CobAgent4_2_1_318.inf* for Windows XP.

3. Using the software deployment tool of your choice, distribute the appropriate files (**not** the directories) to the target computers.
 - Use the *CobAgent4_2_1_318* files for Windows Vista, Windows XP, and Windows 2000 computers.
 - Use the *CobAgent4* files for Windows 98 and Windows ME computers.

Place the files in the *\WINNT\Downloaded Program Files* or *\WINDOWS\Downloaded Programs Files* folder on the target computers.

Note that you cannot copy files to the Downloaded Program Files folder using Windows Explorer. You must copy them at the command line with the copy command.

4. On the command line, enter the following command to register the eCare ActiveX control with Windows Vista, Windows XP, and Windows 2000 computers.

```
regsvr32 /s C:\WINDOWS\Downloaded Program Files\  
CobAgent4_2_1_318.dll
```

For Windows 98 and Windows ME, use

```
regsvr32 /s C:\WINDOWS\Downloaded Program Files\  
CobAgent4.dll
```

The ActiveX control is now installed and registered for use with Windows. (Note that it will *not* appear in the Downloaded Program Files folder in Windows Explorer.)

TO REMOVE THE ACTIVEX CONTROL

If you need to unregister the ActiveX control, run the following command on Windows Vista, Windows XP, and Windows 2000 computers.

```
regsvr32 /s /u C:\WINDOWS\Downloaded Program Files\  
CobAgent4_2_1_318.dll
```

For Windows 98 and Windows ME, use

```
regsvr32 /s /u C:\WINDOWS\Downloaded Program Files\  
CobAgent4.dll
```

INSTALLING THE eCARE ACTIVE X CONTROL ON A LOCAL COMPUTER

If you have Windows Administrator privileges on Support Agent computers, you can also pre-install the eCare ActiveX control by logging in to the Support Agent's computer and signing in to the eCare Support Agent portal. (You may also load the eCare customer portal.) The eCare system will automatically detect the missing (or outdated) eCare ActiveX control and prompt you to install it.

TO INSTALL THE eCARE ACTIVE X CONTROL ON WINDOWS XP AND WINDOWS 2000

1. Sign in to the eCare Support Agent portal.
Internet Explorer will display a window asking you to download and install the eCare ActiveX control. In the download window, click the *Accept* button.
2. In the next dialog box, which asks you to confirm your intention to install the remote-control component, click *Yes*.
3. On Windows XP computers, an additional window appears: Installing Browser Add-On.
Click the yellow ActiveX warning bar and select *Install ActiveX Control* to begin installing the eCare remote-control component. When a window appears asking if you want to install the software, click *Install*.

The ActiveX control will download and install automatically. The ActiveX control is named *CobAgent4 Class*. It will be installed in the *WINNT/Downloaded Program Files* or *WINDOWS/Downloaded Program Files* folder.

TO INSTALL THE eCARE ACTIVE X CONTROL ON WINDOWS VISTA

1. Sign in to the eCare Support Agent portal.
Internet Explorer will display a window asking you to download and install the eCare ActiveX control. In the download window, click the *Accept* button.
2. In the Security Warning dialog box, click *Yes*.
3. In the Installing Browser Add-On window, right-click the yellow ActiveX warning bar and select *Install ActiveX Control*.
4. In the User Account Control dialog box, click *Continue*.
5. In the Security Warning dialog box, click the *Install* button.
Internet Explorer will download and install the eCare ActiveX control.

INSTALLING THE eCARE REMOTE-CONTROL COMPONENT ON MACINTOSH COMPUTERS

You may pre-install the eCare plugin control on both local and remote Macintosh computers.

BEFORE YOU INSTALL THE eCARE PLUGIN

On the Support Agent's computer, the following options must be set in Safari.

- Plug-ins and JavaScript must be enabled.
- Cookies must be allowed.

In addition, *all* pop-up blocking software *must be turned off* before the Support Agent can use eCare. This includes the pop-up blockers that are built in to Safari, as well as third-party blocking software for all platforms.

PRE-INSTALLING THE eCARE PLUGIN ON A REMOTE MACINTOSH COMPUTER

Before you can install the eCare plugin control on a remote computer, you must download a local copy. To do so, use the following procedure.

TO DOWNLOAD THE eCARE PLUGIN

- Open Safari and enter the following URL.
<http://<server>/<service>/ecare4/components/Netopia RC Installer.dmg>
The plugin disk image will download.

TO DEPLOY THE eCARE PLUGIN TO REMOTE COMPUTERS

1. Download the DMG file for the Netopia RC Installer by following the above procedure.
2. Using the software deployment tool of your choice, distribute the file to the target computers.
3. Execute the installer.

PRE-INSTALLING THE eCARE PLUGIN ON THE LOCAL MACINTOSH COMPUTER

You can also pre-install the eCare plugin control logging in to the Support Agent's computer and signing in to the eCare Support Agent portal. (You may also load the eCare customer portal.) The eCare system will automatically detect the missing (or outdated) eCare plugin and prompt you to install it.

TO DOWNLOAD AND INSTALL THE eCARE PLUGIN

1. Sign in to the eCare Support Agent portal.
Safari will display a window asking you to install the eCare plugin.
2. Click the *Accept* button.
The eCare plugin is downloaded as a disk image and opened automatically.
3. To install the plugin, double-click the *Netopia RC Installer* file.
4. A dialog box appears, indicating that the browser plug-in will be installed. Click *Yes*.

Installation then proceeds automatically. When installation is complete, you will be notified. Quit and restart Safari to make the eCare plugin available for use.

The eCare plugin is named *Netopia RC Plugin*. It is located in the *Library/Internet Plug-ins* folder in the current user's home folder.

APPENDIX B: MOTOROLA CONTACTS

MOTOROLA CUSTOMER SERVICE

<http://www.netopia.com/support/software/ecare/>

Customer Support is available Monday–Friday from 6AM–5:30PM Pacific time.

6001 Shellmound Street, 4th Floor
Emeryville, CA 94608
USA

NORTH AMERICA SOFTWARE SALES

1507 LBJ Freeway
Suite 700
Farmers Branch, TX 75234
USA

(800) 485-5741

EUROPE SOFTWARE SALES

Becanusstraat 13-15, bus5
6216 BX
Maastricht Netherlands

+31 (0) 43 354 5020

