



Powering connections

Mitel Connect ONSITE Contact Center Chat Toolkit Guide

December 15, 2022

Trademarks

© Copyright 2017, Mitel Networks Corporation. All Rights Reserved. The Mitel word and logo are trademarks of Mitel Networks Corporation. Any reference to third-party trademarks is for reference only and Mitel makes no representation of ownership of these marks.

Patents

Mitel Connect Contact Center is covered by patents as listed at <http://www.shoretel.com/about/patents.html>.

Version Information

Mitel Connect Contact Center

Date: December 15, 2022

Company Information

Mitel Networks Corporation

350 Legget Dr.

Kanata, Ontario K2K 2W7

Canada

+1.844.937.6483

Table of Contents

	Organization	5
	Conventions	6
Chapter 1	Overview	7
Chapter 2	Installing and Upgrading Connect Contact Center Chat	9
	Installation Prerequisites	10
	Browser Configuration	10
	Network Connectivity	11
	Chat Licenses	11
	Access to Web Site	11
	Installing the Connect Contact Center Chat Servlet	11
	Sample Web Page CSS	11
	Optional Installation Process	14
	Configuring the Connect Contact Center Chat Servlet	14
	Configuring Chat on the Connect Contact Center Server	15
	Verifying Server Communication	16
Chapter 3	Installing and Upgrading Connect Contact Center Web Callback	19
	Installation Prerequisites	20
	Tomcat Server	20
	Browser Configuration	20
	Network Connectivity	20
	Access to Web Site	21
	Installing the Web Callback Servlet	21
	Configuring the Web Callback Servlet	21
	Configuring Web Callback on the Connect Contact Center Server	21
	Verifying Server Communication	22
	Installing the Connect Contact Center Web Callback Servlet	23
Chapter 4	Implementing the Chat Toolkit	25
	Customer Chat Page	26

chat.html Input Parameters	26
Create the Customer Chat Page	27
The chat.html Page	28
Creating the Agent Chat Page	29
Customizing the Web Pages	29
Creating the Web Callback Page	30
The webcallback.html Page	30
Customizing the Web Callback Page	33

Before You Start

The following sections provide an overview of the book.

About This Book 5

About This Book

ShoreTel is now part of Mitel. Together, we look forward to helping you power connections that are brilliantly simple.

This guide provides information on using the Chat Toolkit for Connect Contact Center. This guide provides information for system administrators who configure how the Connect Contact Center Chat Toolkit is used.

Organization

The document is divided into the following chapters:

- Chapter 1, [Overview](#)
- Chapter 2, [Installing and Upgrading Connect Contact Center Chat](#)
- Chapter 3, [Installing and Upgrading Connect Contact Center Web Callback](#)
- Chapter 4, [Implementing the Chat Toolkit](#)

Conventions

The following typographical marking conventions are used in this document.

Marking	Meaning
Bold	Names of interface objects, such as buttons and menus.
<code>Courier</code>	Code examples.
<i>Courier Italic</i>	Variables in code examples.
Blue	Cross references with hyperlinks. Click the blue text to go to the indicated section. All chapters have a list of section links on the first page. Note: Table of Contents entries are also links, but they are not shown in blue.

CHAPTER

1

Overview

Connect Contact Center Chat Toolkit provides a chat browser for handling chat contacts. The Toolkit is a set of scripts, applets and HTML templates that work together to provide chat and web callback capabilities.

Chat sessions are initiated through requests from a chat Web browser. The customer clicks a Chat button on their page to open a chat window while waiting for an agent to respond. A chat browser window then opens at the agent end, allowing agents to start the chat session.

Contact Center supports the following chat services:

- Chat conversation - Customers can initiate a chat session with a live agent from your Web site.
- Web callback - Using chat, customers can leave telephony callback information. The system then uses this information to initiate a telephone call with an available agent.
- Supervisor monitoring - Supervisors can silently monitor, barge in, and coach chat sessions. Refer to the *Contact Center Supervisor Guide* for details.

Refer to the following sections for more information about the Chat Toolkit:

Contents of the Chat Toolkit 8

Contents of the Chat Toolkit

The Toolkit consists of a set of files containing:

- JavaScript code that should be used as is — without any modification.
- Sample forms that can be embedded in a customer web site.
- Mandatory HTML page templates. Certain mandatory pages that reside on the application server can be customized to provide the look and feel of the your company's web site.
- Collection of servlets that should be deployed on a servlet engine.

CHAPTER

2

Installing and Upgrading Connect Contact Center Chat

The chapter explains the process of installing and upgrading Connect Contact Center Chat.

Refer to the following sections for more information about installing and upgrading Connect Contact Center Chat:

The Installation Process	9
Upgrading Connect Contact Center Chat	17

The Installation Process

The process of installing Connect Contact Center Chat is:

1. Make sure the installation prerequisites have been met.
2. Install the Connect Contact Center Chat servlet.
3. Configure the Connect Contact Center Chat servlet.
4. Configure chat on the Connect Contact Center server.
5. Verify server communication.

Installation Prerequisites

This section describes the prerequisites that must be met before installing and using Connect Contact Center Chat. Tomcat Server

A Tomcat server is used to run the chat Java servlets. You must have a fully functional Tomcat server, version 6.0 to 9.X, with Java Adopt Open JRE version 1.8.0.212 or later, installed and set up. Download Tomcat by visiting <http://tomcat.apache.org> and selecting the Windows Service Installer download. Download Java from <http://www.java.com/en/download>.

Although not required, Mitel recommends that the Tomcat directory be installed on the chat server at the root level of the C or D drive. Installing the directory at the root level makes it easy to locate the directory when updating Connect Contact Center Chat.

When a 64-bit version of Tomcat is installed, it must be installed at the root level to use the Connect Contact Center Chat Server installer. If a 64-bit version of Tomcat is already installed in the Program Files directory, install chat using the `ECCChat.war` file as described in [Optional Installation Process on page 14](#).

Ensure that the Java path to Tomcat is correct. The Tomcat installer prompts for the Java path during installation. This path is in the Program Files directory. The default Tomcat port is **8080**, which can be changed.



Note

The Tomcat server versions 9.0.0.M1 to 9.0.0.30, 8.5.0 to 8.5.50, and 7.0.0 to 7.0.99 are shipped with an AJP Connector enabled by default to listen on all configured IP addresses. This can cause a GhostCat Vulnerability issue. To resolve this issue, you can either directly upgrade the Tomcat server to version 9.0.31, 8.5.51, or 7.0.100 for a vulnerability fix or you can disable the AJP Connector directly by using the following procedure:

1. Edit `<CATALINA_BASE>/conf/server.xml`? find the following line (`<CATALINA_BASE>` is the Tomcat work directory): `<Connector port="8009" protocol="AJP/1.3" redirectPort="8443" />`.
2. Either comment it out or just delete it: `<!--<Connector port="8009" protocol="AJP/1.3" redirectPort="8443" />-->`
3. Save the edit and then restart the Tomcat server.



Note

If you have configured redundancy for your contact center system, port 8080 is reserved for the Mitel Interaction Center and you must configure a different port for your Tomcat server.

The Tomcat server should be in a DMZ or other location available to the public Internet. Refer to the Apache documentation for information on installing and setting up the Tomcat server.



Note

If you are installing the Connect Contact Center Chat on a Linux server, you can install only one instance of the Tomcat server.

Browser Configuration

The following browsers are supported:

- Microsoft Internet Explorer (IE) 10 and 11
- Mozilla Firefox 4.x
- Apple Safari 5.x
- Google Chrome

Network Connectivity

The Connect Contact Center server and the chat servlet communicate with each other over the TCP/IP port **31452**. Therefore, this port should be open on your firewall. (The required IP address and ports can be configured in the WEB.XML deployment descriptor of the Connect Contact Center Chat application.)

Chat Licenses

You must have valid chat licenses for the expected number of logged in agents that are members of a chat group. Information on your current Connect Contact Center licenses can be viewed using the Connect Contact Center Director > **Maintenance** > **License** > **License Keys** > **Current Licenses** tab.

Refer to the *Mitel Connect Contact Center Administrator Guide* for information about requesting and entering feature licenses.

Access to Web Site

Make sure you have access to your company's web site to add forms to start chat.

Installing the Connect Contact Center Chat Servlet

Complete the following steps to install the Connect Contact Center Chat servlet:

1. Navigate to the `Setup\ShoreTel Contact Center Chat Server` directory in the Connect Contact Center installation CD and run the `setup.exe` file.

The file is automatically decompressed by the Tomcat server and creates the ECCChat folder in the webapps directory.

If necessary, click the **Change** button to specify a different installation location for the chat servlet.

2. Specify system information as follows:
 - Tomcat server port: 8080
 - Connect Contact Center Server IP address
 - Tomcat Server IP address

3. Click Next.

Follow the prompts to install the Connect Contact Center Chat servlet.

Sample Web Page CSS

```
body{
    background-color: #dddddd;
    font-size: medium;
    font-family: Trebuchet MS, Tahoma, Verdana, Arial, sans-serif;
}
input, select, textarea, button {
    font-family: Trebuchet MS, Tahoma, Verdana, Arial, sans-serif;
}
#chatcontent{
}
#chatsidebar{
    float: left;
    width: 4em;
}
#chatmain{
    position: absolute;
    left: 5em;
    height: 0px;
    float: left;
}
#chatresponsesmain{
    position: absolute;
    left: 500px;
}
.chatresponsestitle{
    font-weight: bold;
}
#chatresponses{
    float: left;
    font-size: medium;
}
.enablechatresponses{
    overflow: auto;
    height: 300px;
    width: 400px;
    background-color: white;
    border: 1px solid #666666;
}
.chatresponsefolder{
}
.chatresponsefolderopen{
    list-style-image: url(images/openfolder.png);
}
.chatresponsefolderclosed{
    list-style-image: url(images/closedfolder.png);
}
```

```
}
.chatresponsefoldercontents{
}
.chatresponseitem{
    list-style-image:
        url(images/message.png);
}
.chatresponseurl{
    list-style-image: url(images/url.png);
}
.chatresponse{
}
.chatinstance{
}
.debuglog{
    overflow: auto;
    height: 300px;
    width: 400px;
    background-color: white;
    border: 1px solid #666666;
    font-size: medium;
    float: left;
}
.chatcalls{
    overflow: auto;
    height: 300px;
    width: 400px;
    float: left;
}
.chatcontainer{
    overflow: auto;
    height: 300px;
    width: 400px;
    background-color: white;
    border: 1px solid #666666;
    float: left;
}
.chatlog{
    font-size: medium;
}
.chattypingstatus{
    color: #777777;
    font-size: small;
}
p.chatstatus{
    color: red;
}
.chatmsg{
    color: black;
}
.chatuser{
    color: black;
```

```
        font-weight: bold;
    }
    .chatdate{
        font-size: small;
        color: #333333;
    }
    .chatsysmsg{
        color: red;
    }
    .clearboth{
        clear: both;
    }
    .chatarea{
        height: 50px;
        width: 400px;
        background-color: white;
        border-style: solid;
        border-width: 1px;
        font-size: medium;
    }
    .chatareafocus{
        border-color: red;
    }
    .chatareaunfocus{
        border-color: black;
    }
    label {
        float: left;
        clear: left;
        text-align: right;
        white-space: nowrap;
        min-width: 5em;
    }
}
```

Optional Installation Process

You also can install the Connect Contact Center Chat Servlet by copying the `ECCChat.war` file from the Chat Toolkit folder on the Connect Contact Center Installation CD to the Tomcat `ECCChat` directory.

Configuring the Connect Contact Center Chat Servlet



Note

These steps are necessary only if you installed the Connect Contact Center Chat Servlet by copying the `ECCChat.war` file from the Chat Toolkit folder on the Connect Contact Center Installation CD to the Tomcat `ECCChat` directory. If you installed the Connect Contact Center Chat Servlet by running the `setup.exe` file, skip this section.

Complete the following steps to configure the Connect Contact Center Chat servlet:

1. On the chat server, open the `web.xml` file with a text editor. This file controls web application connectivity with Connect Contact Center and displays the Java log window on the server for troubleshooting.

The `web.xml` file can be found in `root:\Apache Software Foundation\Tomcat\webapps\ECCChat\WEB-INF`.

2. In the `web.xml` file, change the parameter value, `CCA14`, to the valid host name or IP address of the Connect Contact Center server. The host name must be a fully qualified domain name (FQDN), such as shown in the example code below:

```
<param-name>ECCServerName</param-name>
<param-value>CCA14</param-value>
```

and

```
<param-name>ECKeystoreServerName</param-name>
<param-value>CCA14</param-value>
```

would be changed to:

```
<param-name>ECCServerName</param-name>
<param-value>ECC Server IP Address</param-value>
```

and

```
<param-name>ECKeystoreServerName</param-name>
<param-value>ECC Server IP Address</param-value>
```

3. Save the `web.xml` file.
4. Open the `parameters.js` file with a text editor. This file can be found in `root:\Apache Software Foundation\Tomcat\webapps\ECCChat`.
5. In the `parameters.js` file, under the Tomcat Server base address section, change the parameter of the value `var baseAddr` to the valid host name or IP address of the chat server. The host name must be a fully qualified domain name (FQDN), such as shown in the example code below:

```
change var baseAddr = "http://tomcat.cs.com:8080"
```

```
to var baseAddr = "http://MSmith-HX380.shoretel.com:8080"
```

6. Save the `parameters.js` file.

Configuring Chat on the Connect Contact Center Server

Complete the following steps to configure chat on the Connect Contact Center server:

1. Using Connect Contact Center Director, create or edit an agent group to handle incoming chat requests. To accept chat contacts, this group must have the **Chat** option selected in the Connect Contact Center Director > **Groupings** > **Groups** > **General** tab.

You may want to change the options in the Statistics area to better represent the slower nature of chat (compared to voice calls) so that reporting is more valid.

2. Create or edit a service to queue the incoming chat requests. In order to accept chat contacts, this service must have the **Chat Enabled** option selected in the Connect Contact Center Director > **Routing** > **Services** > **General** tab.

When creating a service for chat, make sure that the following statements are true:

- Any overflow destinations must also be chat-enabled.
 - Chat requests can only be interflowed to chat-enabled services and IRNs.
 - An announcement plan can be used, but only use call control scripts with the Send Message, Send Page, and Hang Up actions. Using a voice action can have adverse effects and may cause instability within the system.
3. Assign the agent group for chat to the chat service, using the Connect Contact Center Director > **Routing** > **Services** > **Destination** tab.
 4. Create an IRN for routing of chat requests using the Connect Contact Center Director > **Routing** > **IRN** > **General** tab. This will be the default IRN number used for chat.

When creating an IRN for chat, make sure that:

- The **Route Directly to Default Destination** option is selected. This is because ANI domain routing and customer routing do not work with a chat request.
 - You can set the initial value of call profile fields for chat requests that enter the IRN. A call profile field can provide information on the chat request that is passed along to the agent desktop.
 - You can specify a shift, which is a routing destination used at different times of the day, as the IRN's destination, as long as the shift destination is chat-enabled.
5. Specify the desired number of inbound chat contacts the system will handle, using the Connect Contact Center > **System Parameters** > **Chat** > **General** > **Max Chat Calls** option. By default, the number of calls in the Max Chat Calls field is 10.

Verifying Server Communication

Complete the following steps to verify server communication:

1. Launch Connect Contact Center Diagnostics Console on the Connect Contact Center server.

2. View the Chat LED. Green indicates a valid connection between the Connect Contact Center server and the Tomcat server.

**Tips**

If multiple chat servlets are connected to the same Connect Contact Center server, the Chat LED is green as soon as one of the chat servlets properly connects.

Upgrading Connect Contact Center Chat

You can upgrade the Connect Contact Center Chat Servlet by following the steps below or by running `setup.exe` the same way you do for installation. Refer to [Installing the Connect Contact Center Chat Servlet on page 11](#).

**WARNING!**

If you use `setup.exe` to upgrade the Connect Contact Center Chat Servlet, customized files are replaced by generic files. The previous configuration is stored in a folder called `ECCChat.bck.yyyy.mm.dd_hh.mm.ss`. This folder is located in the Tomcat folder.

**Note**

If you use the steps in [Installing the Connect Contact Center Chat Servlet on page 11](#) to upgrade Connect Contact Center chat, skip the following steps.

**Tip**

You can use the following procedure to restore default parameter values for the Connect Contact Center Chat Servlet.

**Note**

You must uninstall JRE used by Apache Tomcat server and install Adopt Open JRE 1.8.0.212 or a later version. To do so:

1. In the Tomcat server, go to **Properties** window and click the **Java** tab.
2. Under **Java virtual machine**, select the Java version path that is installed in your system.
3. Click **OK** to change the Java path in the Tomcat server.

Complete the following steps to upgrade Connect Contact Center chat:

1. If the Tomcat server is running, stop it.
2. Move the `ECCChat` folder in the `webapps` directory on the Tomcat server to a backup location.
3. Start the Tomcat server.

4. Copy the `ECCChat.war` file, from the Chat Toolkit folder of the new version of the Connect Contact Center Installation CD, to the Tomcat webapps directory on the chat server.

The file is automatically decompressed by the Tomcat server and creates the Connect Contact Center Chat folder in the webapps directory.

5. Use a text editor to update the `web.xml` file, changing the parameter value for the parameter name `ECCServerName` to the valid host name or IP address of the Connect Contact Center server. The host name must be a fully qualified domain name (FQDN).
6. The `web.xml` file controls web application connectivity with Connect Contact Center and displays the Java log window on the server for troubleshooting. It can be found in `root:\Apache Software Foundation\Tomcat\webapps\ECCChat\WEB-INF`.
7. Open the `parameters.js` file with a text editor. This file is located in `root:\Apache Software Foundation\Tomcat\webapps\ECCChat`.
8. In the `parameters.js` file, under the `//Server Base Address` section, change the parameter of the value `var baseAddr` to the valid host name or IP address of the chat server. The host name must be a fully qualified domain name (FQDN).
9. Copy your current version of the `chat.css` file from your backup location to the Connect Contact Center Chat folder in the webapps directory. This file defines the visual layout of the chat window.
10. Copy the current version of the HTML templates, which includes `chat.html` and any other custom html pages, from your backup location to the ECCChat folder in the webapps directory. These files specify the look and feel of the chat window.
11. Restart the Tomcat server.
12. Verify that the chat server is communicating with the Connect Contact Center server by launching Connect Contact Center Diagnostics Console on the Connect Contact Center server.

If the Chat LED is green, a valid connection between the Connect Contact Center server and the Tomcat server exists.

CHAPTER

3

Installing and Upgrading Connect Contact Center Web Callback

This chapter explains the process of installing and upgrading Connect Contact Center web callback.

Refer to the following sections for more information about installing and upgrading Connect Contact Center web callback:

The Installation Process	19
Upgrading Connect Contact Center Web Callback	22

The Installation Process

The process of installing Connect Contact Center Web Callback is:

1. Make sure the installation prerequisites have been met.
2. Install the Web Callback servlet.
3. Configure the Web Callback servlet.
4. Configure web callback on the Connect Contact Center server.
5. Verify server communication.

Installation Prerequisites

This section describes the prerequisites that must be met before installing and using Connect Contact Center web callback.

Tomcat Server

A Tomcat server is used to run the chat Java servlets. You must have a fully functional Tomcat server, version 6.0 to 9.X, with Java Adopt Open JRE version 1.8.0.212 or later, installed and set up. Download Tomcat from <http://tomcat.apache.org> and select the Windows Service Installer download. Download Java from <http://www.java.com/en/download>.

Although not required, it is recommended that the Tomcat directory be installed on the chat server at the root level of your C or D drive. Installing the directory at the root level makes it easy to locate the directory when updating Connect Contact Center web callback.

Make sure that the Java path to Tomcat is correct. The Tomcat installer prompts for the Java path during installation. This path is in the Program Files directory. The default Tomcat port is 8080, which can be changed.



Note

If you have configured redundancy for your contact center system, port 8080 is reserved for the Mitel Interaction Center and you must configure a different port for your Tomcat server.

The Tomcat server should be in a DMZ or other location available to the public Internet. Refer to the Apache documentation for information on installing and setting up the Tomcat server.



Note

If you are installing the Connect Contact Center web callback on a Linux server, you can install only one instance of the Tomcat server.

Browser Configuration

The following browsers are supported:

- Microsoft Internet Explorer (IE) 10 and 11
- Mozilla Firefox 4.x
- Apple Safari 5.x
- Google Chrome

Network Connectivity

The Connect Contact Center server and the web callback servlet communicate with each other over the TCP/IP port **31452**. Therefore, this port should be open on your firewall. The required IP address and ports can be configured in the `web.xml` deployment descriptor of the Connect Contact Center web callback application.

Access to Web Site

Make sure you have access to your company's web site to add forms to start web-based callbacks.

Installing the Web Callback Servlet

Complete the following steps to install the Web Callback servlet on the chat server:

1. If not already running, start Tomcat.
2. Copy the `WebCallback.war` file from the Chat Toolkit folder on the Connect Contact Center Installation CD to the Tomcat webapps directory.

The file is automatically decompressed by the Tomcat server and creates the WebCallback folder in the webapps directory.

Configuring the Web Callback Servlet

Complete the following steps to configure the Web Callback servlet:

1. Open the **web.xml** file with a text editor. This file controls web application connectivity with Connect Contact Center and displays the Java log window on the server for troubleshooting.

The `web.xml` file is in `root:\Apache Software Foundation\Tomcat\webapps\WebCallback\WEB-INF`.

2. In the `web.xml` file, change the parameter value, `CCA13`, to the valid host name or IP address of the Connect Contact Center server. The host name must be a fully qualified domain name (FQDN).

For example

```
<param-name>ECCServerName</param-name>
<param-value>CCA13</param-value>
```

would be changed to:

```
<param-name>ECCServerName</param-name>
<param-value>10.23.55.110</param-value>.
```

Configuring Web Callback on the Connect Contact Center Server

Complete the following steps to configure web callback on the Connect Contact Center server:

1. Using Connect Contact Center Director, create or edit a service to queue incoming web callback requests using the Connect Contact Center Director > **Routing** > **Services** > **General** tab.

When creating a service for web callback, you can specify the following options using the **Scheduled Callbacks** tab:

- How the system validates the customer's callback number (Numbering Plan)
 - How much time can elapse between customer contact and the time the call is actually dialed
 - The script to play before transferring the callback to an agent
 - Whether to require agent confirmation and how long to wait for agent confirmation
 - The default destination for the call
 - Shifts (routing destinations used at different times of the day)
 - Time range for scheduled callbacks
2. Specify the service for web callback in the `service_req` value parameter in the `webcallback.html` file. The default value is `WebCallback`.

You also can specify an agent queue for callbacks by modifying the `webcallback.html` file, replacing `<input type="hidden" name="service_req" value="WebCallback">` with `<input type="hidden" name="agent_req" value="1234">` where 1234 represents the agent ID.

The `webcallback.html` file is located in `root:\Apache Software Foundation\Tomcat\webapps\WebCallback`.

Verifying Server Communication

Complete the following steps to verify that the chat server is communicating with the Connect Contact Center server:

1. Launch Connect Contact Center Diagnostics Console on the Connect Contact Center server.
2. View the Chat LED. A green color indicates a valid connection between the Connect Contact Center server and the Tomcat server.

Upgrading Connect Contact Center Web Callback

When upgrading from a previous version of Connect Contact Center, you also need to upgrade Connect Contact Center web callback.

Complete the following steps to upgrade to Connect Contact Center web callback:

1. If the Tomcat server is running, stop it.
2. Move the `WebCallback` folder in the `webapps` directory on the Tomcat server to a backup location.
3. Start the Tomcat server.
4. Copy the `WebCallback.war` file, from the Chat Toolkit folder on the Connect Contact Center Installation CD, to the Tomcat `webapps` directory on the chat server.

The file is automatically decompressed by the Tomcat server and creates the WebCallback folder in the webapps directory.

5. Open `web.xml` in a text editor and change the `ECCServerName` value to the valid host name or IP address of the Connect Contact Center server. The host name must be a fully qualified domain name (FQDN).

The `web.xml` file controls web application connectivity with Connect Contact Center and displays the Java log window on the server for troubleshooting. It can be found in `root:\Apache Software Foundation\Tomcat\webapps\WebCallbck\WEB-INF`.

6. Copy the current version of the `chat.css` file from your backup location to the WebCallback folder in the webapps directory.

This file defines the visual layout of the chat window.

7. Copy the current version of the HTML templates (`webcallback.html` and any other custom html pages) from your backup location to the WebCallback folder in the webapps directory.

These files specify the look and feel of the chat window.

8. Restart the Tomcat server.
9. Verify that the chat server is communicating with the Connect Contact Center server by launching Connect Contact Center Diagnostics Console on the Connect Contact Center server.

If the Chat LED has a green color, a valid connection between the Connect Contact Center server and the Tomcat server exists.

Make sure you have access to your company's web site to add forms to start a web callback.

Installing the Connect Contact Center Web Callback Servlet

Complete the following steps to install the Connect Contact Center Chat servlet:

1. Navigate to the `Setup\ShoreTel Contact Center Chat Server` directory in the Connect Contact Center installation CD and run the `setup.exe` file.

The file is automatically decompressed by the Tomcat server and creates the ECCChat folder in the webapps directory.

If necessary, click the **Change** button to specify a different installation location for the chat servlet.

2. Specify system information as follows:
 - Tomcat server port: 8080
 - Connect Contact Center Server IP address
 - Tomcat Server IP address
3. Click **Next**.

Follow the prompts to install the Connect Contact Center Chat servlet.

CHAPTER

4

Implementing the Chat Toolkit

To begin a chat conversation, the customer navigates to a specific page that is usually part of your company's web site. From this page, the chat session is initiated.

Refer to the following sections for more information about implementing the Chat Toolkit:

Enabling a Site with Chat	26
Enabling Web Callback	29

Enabling a Site with Chat

The process of enabling a site with chat includes the following steps:

1. Create the web page used by the customer.

The agent uses the Mitel Interaction Center to reply to chats.

2. Customize the look and feel of the customer's web pages.

Customer Chat Page

The customer chat page solicits the necessary customer information and starts the chat process. The Connect Contact Center Chat Toolkit provides a chat page that you use as a starting place for developing your customer chat page. The page, `chat.html`, includes code for text entry fields for customer name and email address, for a Login button, and to connect with the Connect Contact Center system.

Information can be entered in the page in any way that your web developer considers appropriate for the requirements of your company. The responsibility for creating the page and validating the entered data resides with the web developer. The Connect Contact Center Chat Toolkit performs no tests on the data received.

chat.html Input Parameters

The input parameters of the `chat.html` page are:

- **chatloginname (mandatory)** — The customer name used during the chat session.
- **chatemail (optional)** — The customer email address to which the chat transcript will be sent.
- **routing_device (mandatory)** — The chat IRN defined in Connect Contact Center Director.
- **service_req (optional)** — The service that Connect Contact Center will use to handle chat contacts. This service must be defined in Connect Contact Center Director and chat enabled.



Note

If you define the `service_req` parameter, be aware that it will override the `routing_device` parameter.

- **Call Profile Fields (optional)** — Specify the initial values of the call profile fields defined in Connect Contact Center Director. These values must be strings and are passed as is to the Connect Contact Center system in the following format:

```
<input type="hidden" name="Priority" value="0">
<input type="hidden" name="LanguageText" value="ENGLISH">
```

In the example above, `Priority` and `LanguageText` are sample user-defined call profile fields.

- **Skills (optional)** — Identifies the values of the skills defined in Connect Contact Center Director. The values must be numeric, and are passed as is to the Connect Contact Center system to route the call. This information is passed in the following format:

```
<input type="hidden" name="English" value="100">
<input type="hidden" name="Spanish" value="0">
<input type="hidden" name="Russian" value="0">
```

In the example above, English, Spanish, and Russian are the names of skills defined in Connect Contact Center Director. The value listed indicates the importance of the skill. In the example above, the configuration of the chat window places a high priority on English skills.

Create the Customer Chat Page

Whether a field in a page is hidden, filled in by the customer, or hard coded is irrelevant to Connect Contact Center. However, it is important for page field names and their values to match the field names and values as defined in Connect Contact Center Director.

After the page is submitted by the customer, Connect Contact Center starts the chat session.

Complete the following steps to create the customer chat page:

1. Create a container HTML page.
2. In the container HTML page, embed the code found in the `chat.html` page.

The `chat.html` page is located in `\Apache Software Foundation\Tomcat\webapps\ECCChat`. Open the page using a text editor. Refer to [The chat.html Page on page 28](#) for a detailed explanation of the `chat.html` page.

3. Specify the routing device as the chat IRN created in Connect Contact Center Director by changing the value of the `routing_device` parameter in the `chat.html` page.

For example,

```
change <input type="hidden" name="routing_device" value="5555">
to <input type="hidden" name="routing_device" value="30125">
```

4. Identify the chat service created in Connect Contact Center Director by changing the value of the `service_req` parameter in the `chat.html` page. This must be the service pointed to by the chat IRN.

For example,

```
change <input type="hidden" name="service_req" value="SalesService">
to <input type="hidden" name="service_req" value="Chat Service">
```

5. If you want to use call profile fields and/or skills to gather information and further specify routing, include the necessary parameters. For example, to set the initial value of the `Priority` and `LanguageText` call profile fields, set the values as follows:

```
<input type="hidden" name="Priority" value="0">
```

```
<input type="hidden" name="LanguageText" value="ENGLISH">
```

To set the value of the Skill fields, set the values as follows:

```
<input type="hidden" name="English" value="100">
<input type="hidden" name="Spanish" value="0">
<input type="hidden" name="Russian" value="0">
```

6. Save the HTML page.

The chat.html Page

The `chat.html` page is a default web page, provided by the Connect Contact Center Chat Toolkit, that you can use as a starting place for developing your customer web page. This page is in `\Apache Software Foundation\Tomcat\webapps\ECCChat`.

The contents of the `chat.html` file are:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Chat</title>
    <link rel="stylesheet" type="text/css" href="chat.css">
    <script language="javascript" type="text/javascript" src="date.js"></script>
    <script language="javascript" type="text/javascript" src="parameters.js"></script>
    <script language="javascript" type="text/javascript" src="chatcommon.js"></script>
    <script language="javascript" type="text/javascript" src="chat.js"></script>
  </head>
  <body onload="startChat();" onunload="stopChat();">
    <h2>Chat</h2>
    <div id="chatlogin">
      <form id="chatform" method="post" target="chatframe" action=""
onsubmit="return login();">
        <div> <label for="chatloginname"> Name:</label>
          <input id="chatloginname" name="name" type="text" maxlength="16">
        </div>
        <div> <label for="chatemail"> Email:</label>
          <input id="chatemail" name="Cust_email" type="text"><i>(optional)</i>
        </div>
        <div> <label for="chatloginbutton">&nbsp;</label>
          <input id="chatloginbutton" type="submit" value="Login">
        </div>
        <input type="hidden" name="usertype" value="client">
        <input type="hidden" name="routing_device" value="5555">
        <input type="hidden" name="frameset_page" value="frame_index.html">
        <input type="hidden" name="service_req" value="SalesService">
        <input type='hidden' name='system_target' value='browser_win'>
      </form>
    </div>
```

```
<div id="chatcontent">
  <div id="chatsidebar"></div>
  <div id="chatmain"></div>
</div>
<div id="chatframediv">
</div>
</body>
</html>
```

Creating the Agent Chat Page

The agent chat page is launched when an agent accepts a chat contact. The Connect Contact Center Chat Toolkit provides a default page that you can use as a starting place for developing your agent chat page. The page, `agentchat.html`, includes code that indicates the status of the connection, a window that displays the chat text, a window to select a chat response, and buttons to send text to the customer and disconnect from the session.

Complete the following steps to create the agent chat page:

1. Create a container html page.
2. In the container html page, embed the code found in the `agentchat.html` page.
3. The `agentchat.html` page is located in `\Apache Software Foundation\Tomcat\webapps\ECCChat`. Open the page using a text editor.
4. Save the html page.

Customizing the Web Pages

You can easily customize your customer and agent chat pages by modifying the `chat.css` file provided by the Connect Contact Center Chat Toolkit. This file is located in `\Apache Software Foundation\Tomcat\webapps\ECCChat`.

The `chat.css` file uses standard CSS formatting.

Enabling Web Callback

The process of enabling a site with web callback includes the following steps:

1. Create the web callback page used by the customer.
2. Customize the look and feel of the web page.

Creating the Web Callback Page

The web callback page allows your customers to use your web site to schedule a time to be called back. The Connect Contact Center Chat Toolkit provides a web callback page that you use as a starting place for developing your own web callback page. The page, `webcallback.html`, includes code for text entry fields and menus for customer information, for Submit and Reset buttons, and to connect with the Connect Contact Center system.

Complete the following steps to create the web callback page:

1. Create a container html page.
2. In the container html page, embed the code found in the `webcallback.html` page.
3. The `webcallback.html` page is located in `\Apache Software Foundation\Tomcat\webapps\WebCallback`. Open the page using a text editor. A detailed explanation of the `chat.html` page can be found in [The webcallback.html Page on page 30](#).
4. Save the html page.

The webcallback.html Page

The `webcallback.htm` page is a default web page, provided by the Connect Contact Center Chat Toolkit, that you can use as a starting place for developing your own web callback page. The page can be found in `\Apache Software Foundation\Tomcat\webapps\WebCallback`.

The contents of the `webcallback.html` file are:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8"/>
    <title>Web Callback</title>
    <link rel='stylesheet' type='text/css' href='chat.css'>
  </head>
  <body>
    <form method="post" action="http://xxx.xxx.xxx.xxx:8080/WebCallback/WebCallbackServlet">
      <div>
        First Name <input name="CallBack_Name"/>
        Surname <input name="CallBack_Surname"/>
      </div>
      <div>
        Phone Number:(
        <input name="CallBack_AreaCode" size="3"/>
        <input name="CallBack_Phone" size="7"/> (Area-Code) Number
      </div>
      <div>
        Callback Time
        <select name="CallBack_Hour">
          <option value="1">01</option>
          <option value="2">02</option>
```

```

        <option value="3">03</option>
        <option value="4">04</option>
        <option value="5">05</option>
        <option value="6">06</option>
        <option value="7">07</option>
        <option value="8">08</option>
        <option value="9">09</option>
        <option value="10">10</option>
        <option value="11">11</option>
        <option value="12">12</option>
    </select>
    <select name="CallBack_Min">
        <option value="0">00</option>
        <option value="5">05</option>
        <option value="10">10</option>
        <option value="15">15</option>
        <option value="20">20</option>
        <option value="25">25</option>
        <option value="30">30</option>
        <option value="35">35</option>
        <option value="40">40</option>
        <option value="45">45</option>
        <option value="50">50</option>
        <option value="55">55</option>
    </select>
    <select name="CallBack_AM_PM">
        <option value="AM">AM</option>
        <option value="PM">PM</option>
    </select>
</div>
<div>
    <!--
        Callback Offset<input name="CallBack_OffsetSeconds" value="-1"/>Seconds
        (negative is ignored, 0 is callback now)
    -->
    <label for="callbackNow">Callback Now</label>
    <input id="callbackNow" type="checkbox" name="CallBack_OffsetSeconds"
value="0"/>
</div>
<div>
    Please press "Submit" button after completing the form.
<br/>
    Press "Reset" to clear and start again.
</div>
<div>
    <input name="Submit" type="submit" value="Submit"/>
    <input name="Reset" type="reset" value="Reset"/>
    <input type="hidden" name="service_req" value="WebCallback"/>
    <!--
        <input type="hidden" name="agent_req" value="1000"/>
    -->
    <input type="hidden" name="CallBack_ResponsePage"
value="webcallback_response.html"/>

```

```

        <input type="hidden" name="CallBack_ResponseErrorPage"
value="webcallback_error.html"/>
    </div>
</form>
</body>
</html>

```

webcallback.html Parameters

The parameters of the webcallback.html page are:

Field Name	Description	Legal Values
FORM method=POST	Method should be post.	POST
action="http://tomcat.cs.com:8080/WebCallbackServlet"	The WebCallbackServlet on the Tomcat server.	
agent_req	The agent queue that Connect Contact Center will use to route callbacks. This agent queue must be defined in Connect Contact Center Director.	String
CallBack_Name	Name of the person to call.	String
CallBack_Surname	Surname of the person to call	String
CallBack_AreaCode	Area code of the destination phone	String with numbers
CallBack_OffsetSeconds	Number of seconds from when a customer requests a web callback and the outbound call is made.	Number
CallBack_Phone	Phone number to dial	String with numbers
CallBack_Hour	Time of the return call (Hours)	1-12
CallBack_Min	Time of the return call (Minutes)	00-59
CallBack_AM_PM	Call back time is AM or PM?	AM PM
CallBack_ResponsePage	Page that will be sent back to the customer after submitting the call back page.	webcallback_response.html
CallBack_ResponseErrorPage	Page that will be sent back to the customer if an error occurs.	webcallback_error.html
service_req	Service that Connect Contact Center will use to handle chat contacts. This service must be defined in Connect Contact Center Director and chat enabled.	string

Customizing the Web Callback Page

You can easily customize your web callback page by modifying the `chat.css` file provided by the Connect Contact Center Chat Toolkit. This file is located in `\Apache Software Foundation\Tomcat\webapps\ECCChat`.

The `chat.css` file uses standard CSS formatting.

