# Enterprise Chat and Email Administrator's Guide to Chat and Collaboration Resources, Release 12.6(1)

**For Unified and Packaged CCE**

First Published: May 2021

**Americas Headquarters**
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
https://www.cisco.com
Tel: 408 526-4000 800
553-NETS (6387)
Fax: 408 527-0883

# Copyrights and Trademarks

# Contents

# Preface

- Related Documents
- Communications, Services, and Additional Information
- Field Alerts and Field Notices
- Documentation Feedback
- Document Conventions

Welcome to Enterprise Chat and Email (ECE), which provides multichannel interaction software used by businesses all over the world as a core component to the Unified Contact Center Enterprise product line. ECE offers a unified suite of the industry's best applications for chat and email interaction management to enable a blended agent for handling of web chat, email and voice interactions.

## Related Documents

The latest versions of all Cisco documentation can be found online at https://www.cisco.com

| Subject | Link |
| --- | --- |
| Complete documentation for Enterprise Chat and Email, for both Cisco Unified Contact Center Enterprise (UCCE) and Cisco Packaged Contact Center Enterprise (PCCE) | https://www.cisco.com/c/en/us/support/customer-collaboration/cisco-enterprise-chat-email/tsd-products-support-series-home.html |

## Communications, Services, and Additional Information

- To receive timely, relevant information from Cisco, sign up at Cisco Profile Manager.
- To get the business impact you're looking for with the technologies that matter, visit Cisco Services.
- To submit a service request, visit Cisco Support.
- To discover and browse secure, validated enterprise-class apps, products, solutions and services, visit Cisco Marketplace.

- To obtain general networking, training, and certification titles, visit Cisco Press.
- To find warranty information for a specific product or product family, access Cisco Warranty Finder.

### Cisco Bug Search Tool

Cisco Bug Search Tool (BST) is a web-based tool that acts as a gateway to the Cisco bug tracking system that maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. BST provides you with detailed defect information about your products and software.

## Field Alerts and Field Notices

Cisco products may be modified or key processes may be determined to be important. These are announced through use of the Cisco Field Alerts and Cisco Field Notices. You can register to receive Field Alerts and Field Notices through the Product Alert Tool on Cisco.com. This tool enables you to create a profile to receive announcements by selecting all products of interest.

Log into www.cisco.com and then access the tool at https://www.cisco.com/cisco/support/notifications.html

## Documentation Feedback

To provide comments about this document, send an email message to the following address: contactcenterproducts_docfeedback@cisco.com

We appreciate your comments.

## Document Conventions

This guide uses the following typographical conventions.

| Convention | Indicates |
|---|---|
| *Italic* | Emphasis. Or the title of a published document. |
| **Bold** | Labels of items on the user interface, such as buttons, boxes, and lists. Or text that must be typed by the user. |
| `Monospace` | The name of a file or folder, a database table column or value, or a command. |
| *Variable* | User-specific text; varies from one user or installation to another. |

# Chat & Collaboration Basics

- Navigating the Console
- Key Terms and Concepts for Chat and Callback
- Configuring the System for Chat and Callback

## Navigating the Console

The console interface has different functional areas that follow a hierarchical structure. These areas adjust and refresh as you click different options in the menus to navigate through the console. Thus, it is important to logically navigate the interface with a step-by-step approach in mind.

The functional areas are as follows:

1. **Partition and Department Menu:** Use this dropdown menu to switch between departments or to switch to the Partition space that allows you to adjust the configurations of multiple departments at once.
2. **Top Menu:** This is the top level navigation menu. Clicking an option in the Top menu changes the options that are available in the left menu.
3. **Left Menu:** Once you have selected an option from the Top menu, use this menu to navigate to through the different sections of the application you can configure.
4. **Workspace Area:** Upon selecting an option from the left menu, the space in the center of the screen refreshes. Here, you can make all your modifications and configurations to the application.
5. **Workspace Toolbar:** The buttons that appear for each workspace you have selected may vary, but you must remember to save your changes before moving on.

The console follows a particular logic path when it comes to navigation and displaying information. When working in the console, you should follow the process of navigating the console in the order of these different sections.

In most instances, you start in the upper left section of the desktop to verify that you are working in the correct department or at the partition level. Next, you want to select an option from the Top menu and navigate the Left menu to select the setting or configuration you wish to adjust. Doing so refreshes Workspace Area, where you can select the individual functions of the configuration and make your necessary adjustments. When you are finished with your changes, use the buttons in the Workspace Toolbar to finalize your modifications.

# Key Terms and Concepts for Chat and Callback

## Chat Activity

An activity created for a chat session between a customer and an agent. A chat is a real-time interaction between an agent and a customer where they exchange text messages. As part of a chat, agents can also push web pages to customers. The chat is routed to a queue, and a message is sent to Unified CCE. Unified CCE processes the activity and assigns the chat to an available agent. Configuring the System for Chat and Callback

## Callback Activities

An activity created for a web callback session between an agent and a customer. In callback activities, the customer provides personal details (including the phone number) in a customized web form, and the system automatically generates a call activity and assigns the activity to an agent. For more information, see About Call Variables.

## Delayed Callback Activities

An activity created for a delayed web callback session between an agent and a customer. In delayed callback activities, customer provides the personal details (including the phone number and a delay time, after which he wants to receive the call) in a customized web form, and the system automatically creates a call activity after the specified delayed time and assigns the activity to an agent. For more information, see About Call Variables.

## Template Sets

Template sets consist of HTML, CSS (cascading style sheets) and JS (JavaScript) files. The CSS files control the look and feel of the customer's chat and callback area. The JS files contain the logic used to render data in the chat and Callback area. Templates are also used to determine the type of information collected on the web form and used to identify the customer (for example, name, email address, phone number). You can also compose messages that the customer will see under certain circumstances (for example, if they request a chat session out of hours). The system comes with template sets: Aria for chats and Rainbow for callback and delayed callback activities. For more information, see About Chat Template Sets or About Callback Template Sets.

## Entry Points

An entry point defines the starting point from which customers initiate chat and web callback interactions. Every help link on a website is mapped to an entry point and each entry point has a queue associated with it. Different template sets can be used with the same entry point. The queue is used to route activities to agents while the template set determines the look and feel of the different pages displayed to chat customers. For more information, see About Entry Points.

## Chat Customer Single Sign-On

Secure Chat, also known as Chat Customer Single Sign-On, allows chat entry points to transfer customer context information from the company website to the application through SAML. This allows customers who are already recognized on the company website to use a SSO-enabled entry point to chat with a customer without having to provide redundant information. This feature is available for auto-login configuration only. To learn how to enable auto-login for chat, see Enabling Auto-Login. To learn how to configure entry points for Secure Chat, see Enabling Customer Single Sign-On for Chat Templates.

# Configuring the System for Chat and Callback

This section describes the objects that should be created to set up the system for chat, callback, and delayed callback activities.

## For Chat Activities

- An email address (a valid email address on your mail server): Contact your system administrator to get this configured.
- Quick Responses and Quick Links (from the Apps > Chat space): For details, see About Chat Articles.
- Users (from the Users space): For details, see the *Enterprise Chat and Email Administrator's Guide*.
- Queues (from the Business Rules > Queues space): For details, see the About Queues.
- Templates (from the web server): For details, see About Chat Template Sets.
- Entry points (from the Apps > Chat space): For details, see About Entry Points.
- Transfer codes (from the Business Rules > Codes and Classification space): For details, see the *Enterprise Chat and Email Administrator's Guide*.
- Share chat transcripts with other departments (from the Administration Console, Departments node): For details, see the *Enterprise Chat and Email Administrator's Guide*.
- Configuring the dynamic messages to be displayed for chats activities. For details, see Configuring DRASR.
- Enable the website for chat by inserting a code snippet.

## For Callback and Delayed Callback Activities

To be able to set up the system for these activities, the following objects must be created:

- Users (from the User Management space): For details, see the *Enterprise Chat and Email Administrator's Guide*.
- Queues (from the Business Rules > Queues space): For details, see About Queues.
- Templates (From the web server): For details, see About Callback Template Sets.
- Entry points (from the Apps > Chat space): For details, see About Entry Points.
- Enable the website for callback by inserting a code snippet.

## Data Masking for Chat

Data masking allows businesses to ensure that sensitive information, like credit card numbers, Social Security Numbers, bank account numbers, and so on, is not transmitted from the system to the customers and vice versa. If the customer and agent do add any sensitive data in the email content and chat messages, all such data is masked before it is displayed to customers and agents and before it is stored in the system.

For chat, you have the option to allow customers to disable data masking for selective messages. The Off-therecord feature allows agents and customers to exchange sensitive information as data masking rules do not apply to these messages. Any information exchanged while off-record is not stored in the system. Off-the-record conversations can be started and stopped only by customers. For details about setting up data masking, see About Data Masking.

> Data masking rules also apply to the message typed by customers in the callback request form.

## Secure Chat

Secure Chat allows chat entry points to transfer customer context information from the company website to the application through SAML. This allows website visitors who are already recognized on the company website to use a single sign-on enabled entry point to chat with an agent without having to provide redundant information. Entry points can be configured to allow website visitors to log in to secure chat upon initially accessing the chat entry point, or to allow agents to initiate the authentication process mid-chat.

To enable secure chat, the chat templates must be configured, an entry point must be configured to allow secure chat, and customer single sign-on must be configured.

- To learn how to enable secure chat for Aria chat templates, see Enabling Auto-Login and Enabling Customer Single Sign-On for Chat Templates.
- For details about configuring entry points to use secure chat and mid-chat authentication, see Creating Entry Points and Adding Chat and Collaboration Links to Websites.
- For more details about configuring Chat Customer Single Sign-On, see the *Enterprise Chat and Email Administrator's Guide*.

## Chat Attachments

Customers and agents can send files to each other during a chat. Once configured by an administrator, customers and agents can browse to a file and attach it to their chat messages. Customers can also drag and drop files into the chat text editor. For more details about the types of files that can be allowed or blocked, or how to set up chat attachments, see the *Enterprise Chat and Email Administrator's Guide*.

## Services for Chat and Collaboration

Make sure that the following services are configured properly and are in running state. For details about configuring these services, see the *Enterprise Chat and Email Administrator's Guide*.

- External Agent Assignment Service (EAAS) (For integrated emails, integrated chats, callback, and delayed callback activities only.)

- External Agent Messaging Service (EAMS) (For integrated emails, integrated chats, callback, and delayed callback activities only.)

## Settings for Chat and Collaboration

Make sure that the following partition and department level settings are configured properly. For details about these settings, see Chat Settings.

### Settings at the Partition Level

- Chat Watchdog Interval
- Starvation Time for Activities
- Chat Auto-Pushback Settings
- Enable auto-pushback of chats
- Expiry time for auto-pushback for chats (minutes)
- Make agent unavailable on auto-pushback of chats

### Settings at the Department Level

- Chat - Agent availability choice enabled
- Chat - Agent chat message maximum length
- Chat - Auto block check
- Chat - Auto spellcheck
- Chat - Daily service level sample set definition
- Chat - Disable typing area and page push area on customer exit
- Chat - Display timestamp in agent chat console
- Chat - Display timestamp in completed chat transcript
- Chat - Disable typing area and page push area on customer exit
- Chat - Enable Sound Alert Chat - Force activity categorization
- Chat - Force resolution code
- Chat - Inbox sort column
- Chat - Inbox sort order
- Chat - My Monitor - Activity refresh interval (seconds)

- Chat - SLA for Response Time (seconds)
- Reason for chat transfer

# Data Masking



## About Data Masking

Data masking allows businesses to ensure that sensitive information, like credit card numbers, Social Security Numbers, bank account numbers, and so on. is not transmitted from the system to the customers and vice versa. If the customer and agent do add any sensitive data in the email content and chat messages, all such data is masked before it is displayed to customers and agents and before it is stored in the ECE system.

Data masking is the process of scanning the content for sensitive information and applying regular expressions to mask the sensitive information and hide the original data with characters, like, * ^ #. Data is masked using patterns, which are defined using Javascript and Java regular expressions.

> Data masking is available for emails and chats.

Masking patterns are defined by the partition administrators and then are applied to the email and chat channel. The partition administrator can decide to manage the channels for all departments at the partition level, or can allow each department to manage their own configurations.

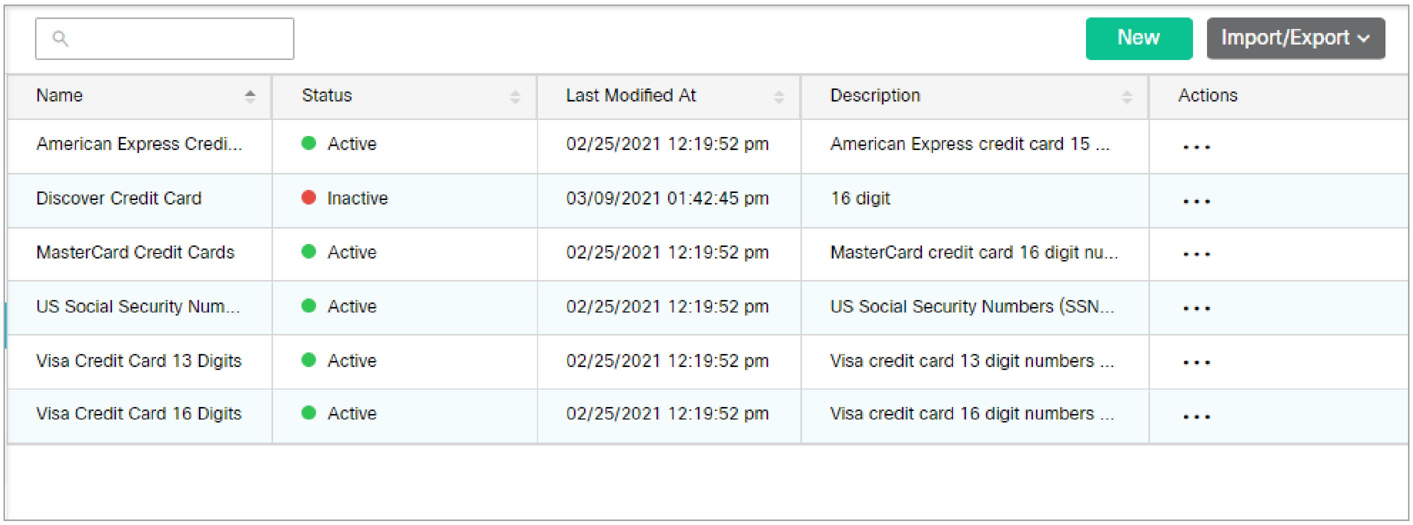
For chat, you have the option to allow customers to disable data masking for selective messages. The Off-the-record feature allows agents and customers to exchange sensitive information as data masking rules do not apply to these messages. Any information exchanged while off-record is not stored in the ECE system. Off the record conversation can be started and stopped only by customers.

> Data masking rules also apply to the message typed by customers in the callback request form.

# About Patterns

Patterns are definitions of data masking rules that you apply to the content of emails and chat messages to hide sensitive data. Patterns are defined using JavaScript and Java regular expressions. In the pattern definition, you also define the character to use for replacing the matching data (for example, *, ^, #). You can enable the Luhn algorithm for masking credit card numbers. This algorithm distinguishes the valid credit card numbers from a random sequence of numbers.

A partition administrator with the following actions can manage patterns, that is create, delete, edit, copy, import, and export patterns. You can either create a pattern from the user interface, or you can create patterns in an XML file and import the file using the import feature.




| Name | Status | Last Modified At | Description | Actions |
|---|---|---|---|---|
| American Express Credi... | Active | 02/25/2021 12:19:52 pm | American Express credit card 15 ... | ... |
| Discover Credit Card | Inactive | 03/09/2021 01:42:45 pm | 16 digit | ... |
| MasterCard Credit Cards | Active | 02/25/2021 12:19:52 pm | MasterCard credit card 16 digit nu... | ... |
| US Social Security Num... | Active | 02/25/2021 12:19:52 pm | US Social Security Numbers (SSN... | ... |
| Visa Credit Card 13 Digits | Active | 02/25/2021 12:19:52 pm | Visa credit card 13 digit numbers ... | ... |
| Visa Credit Card 16 Digits | Active | 02/25/2021 12:19:52 pm | Visa credit card 16 digit numbers ... | ... |

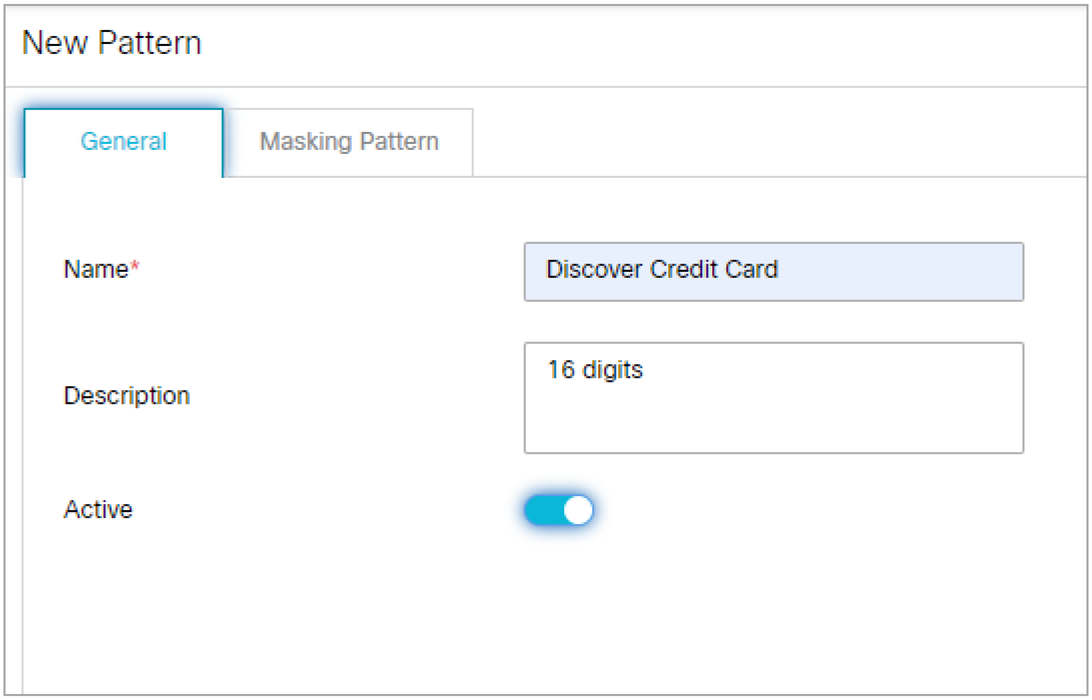
# Creating Patterns

You can either create a pattern from the user interface, or you can create patterns in an XML file and import the file using the import feature. Masking patterns can be created at the partition and at the department level.

### To create a pattern:

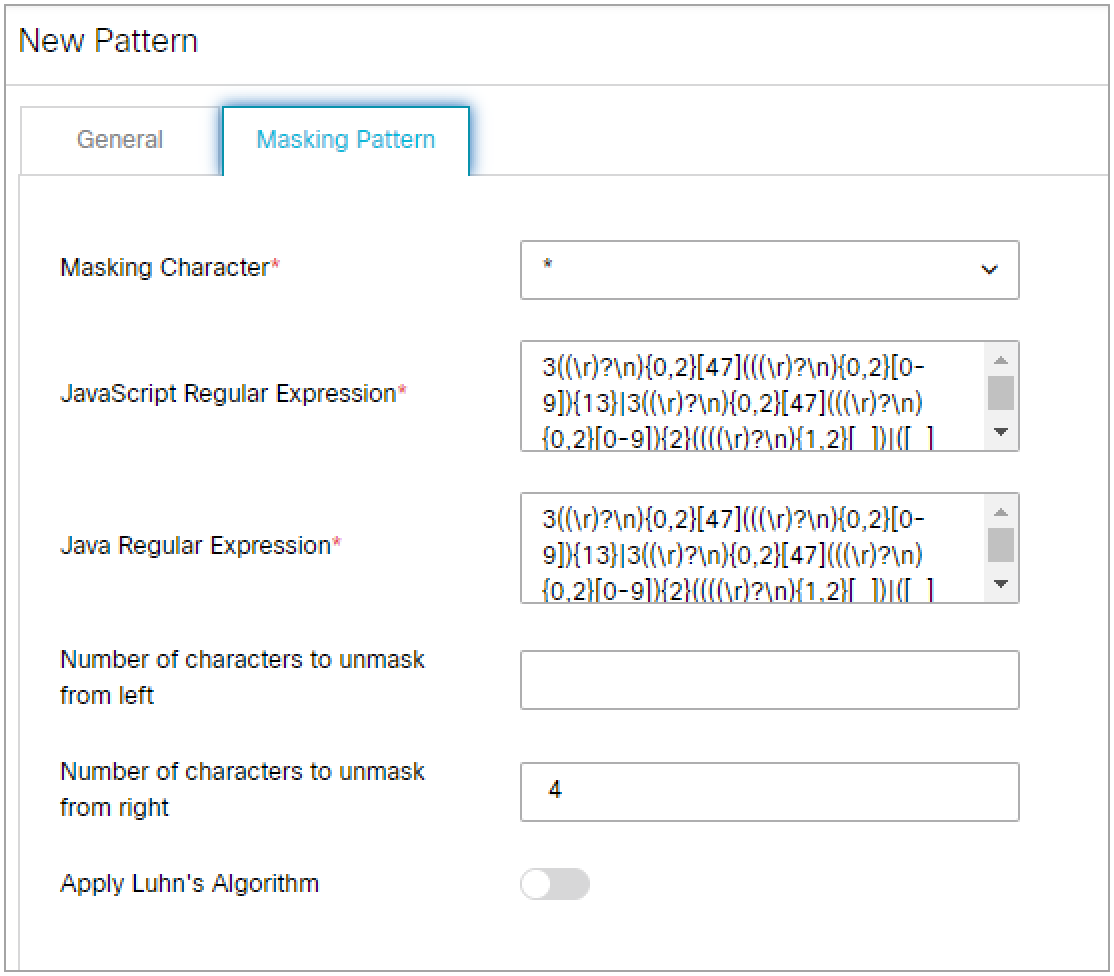
1. In the global-level Top Menu, click the **Security** option.
2. In the Left menu, navigate to **Data Masking > Patterns**.
3. Click the **New** button.
4. In the New Pattern space, on the General tab, set the following:
    - **Name:** Type a name for the pattern.

- **Description:** Provide a description for the pattern that explains what type of masking is done by the pattern.
- **Active:** Click the Toggle button to make the pattern active when it is ready for use. Only active patterns can be applied to channels. Once a pattern is made active and used in channels, it can be made inactive only after the association from the channels is removed.




5. On the Masking Pattern tab, set the following:

- **Masking Character:** From the dropdown list, select the character to be used to mask the data. The default value is *. Options available are: *, -, #, X, x.
- **JavaScript Regular Expression:** Provide the JavaScript regular expression for masking.
- **Java Regular Expression:** Provide the Java regular expression for masking.
- **Number of characters to unmask from left:** Provide the number of characters that should be ignored from left while masking. For example, if you are masking a 10 digit account number and you do not want to mask the first 4 numbers of the account number, the account number will show as 8765******
- **Number of characters to unmask from right:** Provide the number of characters that should be ignored from right while masking. For example, if you are masking the social security number and you do not want to mask the last 4 numbers of the SSN, the SSN will show as *****3545
- **Apply Luhn's Algorithm:** Select **Yes** to apply the Luhn's algorithms to credit card numbers.




6. Click the **Save** button.

# Creating Patterns in XML File

You can either create a pattern from the user interface, or you can create patterns in an XML file and import the file using the import feature.

While preparing a file for importing patterns, keep in mind that:

- Only XML files can be used to import patterns.
- You can name the file anything you want.
- Elements should be defined in the order specified in the pattern file exported from the application.
- Elements and values of elements in the XML file are case sensitive.
- For user created patterns, the isDefault element should be always set to **no**. Likewise, for default patterns, the isDefault element should be always set to **yes**.
- If you are importing a pattern that already exists in the system, your existing pattern will be overwritten by the import process.

The following table lists the names of the properties as they appear in the file and on the UI. For the description of each field, see Creating Patterns.

| Name on the UI | Name in the file |
|---|---|
| Name | name |
| Description | description |
| Active | isActive |
| Default | isDefault |
| Masking character | maskingCharacter |
| JavaScript Regular Expression | javascriptRegularExpression |
| Java Regular Expression | javaRegularExpression |
| Number of characters to unmask from right | numOfCharsToUnmaskFromLeft |
| Number of characters to unmask from left | numOfCharsToUnmaskFromRight |
| Apply Luhn algorithm | applyLuhnAlgorithm |

A sample pattern looks like:

new_pattern.xml

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<patterns xmlns="http://bindings.egain.com/security/masking">
    <pattern>
        <name>Pattern for masking Visa, MasterCard, and American
        Express credit card numbers</name>
        <description>Single pattern for three cards</description>
        <isActive>yes</isActive>
        <isDefault>no</isDefault>
        <maskingCharacter>*</maskingCharacter>
        <javascriptRegularExpression>
        ((?:(?:4\d{3})|(?:5[1-5]\d{2})|6(?:011|5[0-9]{2}))(?:-?|\040?)
        (?:\d{4}(?:-?|\040?)){3}|(?:3[4,7]\d{2})(?:-?|\040?)\d{6}(?:-?
        |\040?)\d{5})</javascriptRegularExpression>
        <javaRegularExpression>
        ((?:(?:4\d{3})|(?:5[1-5]\d{2})|6(?:011|5[0-9]{2}))(?:-?|\040?)
        (?:\d{4}(?:-?|\040?)){3}|(?:3[4,7]\d{2})(?:-?|\040?)\d{6}(?:-?
        |\040?)\d{5})</javaRegularExpression>
        <numOfCharsToUnmaskFromLeft>4</numOfCharsToUnmaskFromLeft>
        <numOfCharsToUnmaskFromRight>4</numOfCharsToUnmaskFromRight>
        <applyLuhnAlgorithm>yes</applyLuhnAlgorithm>
    </pattern>
</patterns>
```

# Exporting Masking Patterns

Patterns can be exported in XML format to share them across installations or if you wish to edit the patterns through an XML file. All the patterns configured in the system will be part of the exported XML file.

## To export patterns:

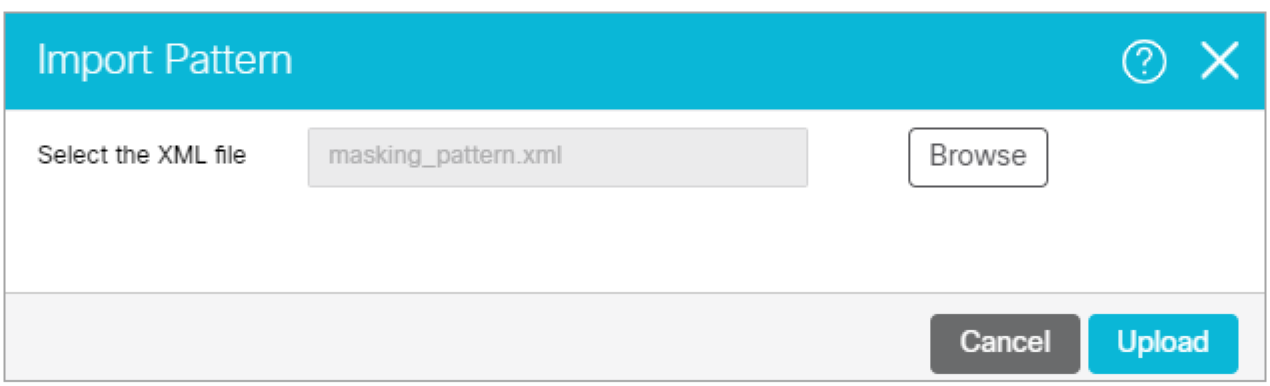
1. In the global-level Top Menu, click the **Security** option.
2. In the Left menu, navigate to **Data Masking > Patterns**.
3. Click the **Import/Export** button and select the **Export Pattern** option.
4. The patterns XML file is downloaded.

# Importing Masking Patterns

Only XML files can be used to import patterns.

## To import a pattern:

1. In the global-level Top Menu, click the **Security** option.
2. In the Left menu, navigate to **Data Masking > Patterns**.
3. Click the **Import/Export** button and select the **Import Pattern** option.
4. In the Import Patterns window, provide the location of the XML file. Click **OK**.




You are also notified if the import process will over-write existing patterns.

If the file has any issues, the import process is aborted and the user is notified about the same. Some of the issues with the file can be:

- Type of file is not XML.
- Size of the imported file is more than 10 MB.
- XML is malformed.

- The values of the name, description, JavaScript Regular Expression, Java Regular Expression fields are more than the allowed size.
- A custom pattern is defined as a default pattern.
- A default pattern is not defined as a default pattern.
- The JavaScript regular expression defined in the file is not correct.
- The Java regular expression defined in the file is not correct.
- You are deactivating a pattern that is in use.

# Copying Patterns

### To copy a pattern:

1. In the global-level Top Menu, click the **Security** option.
2. In the Left menu, navigate to **Data Masking > Patterns**.
3. In the Actions column of the desired pattern, click the **Options** ⋯ button and select the **Create Copy** option.

You are notified when the pattern is copied. All patterns are copied in the inactive state. You can make them active when you are ready to use the pattern.

# Deleting Patterns

Patterns cannot be deleted if they are associated with a channel. You must remove all associations before deleting the pattern.

### To delete a pattern:

1. In the global-level Top Menu, click the **Security** option.
2. In the Left menu, navigate to **Data Masking > Patterns**.
3. In the Actions column of the desired pattern, click the **Options** ⋯ button and select the **Delete** option.
4. In the window that appears, confirm the deletion.

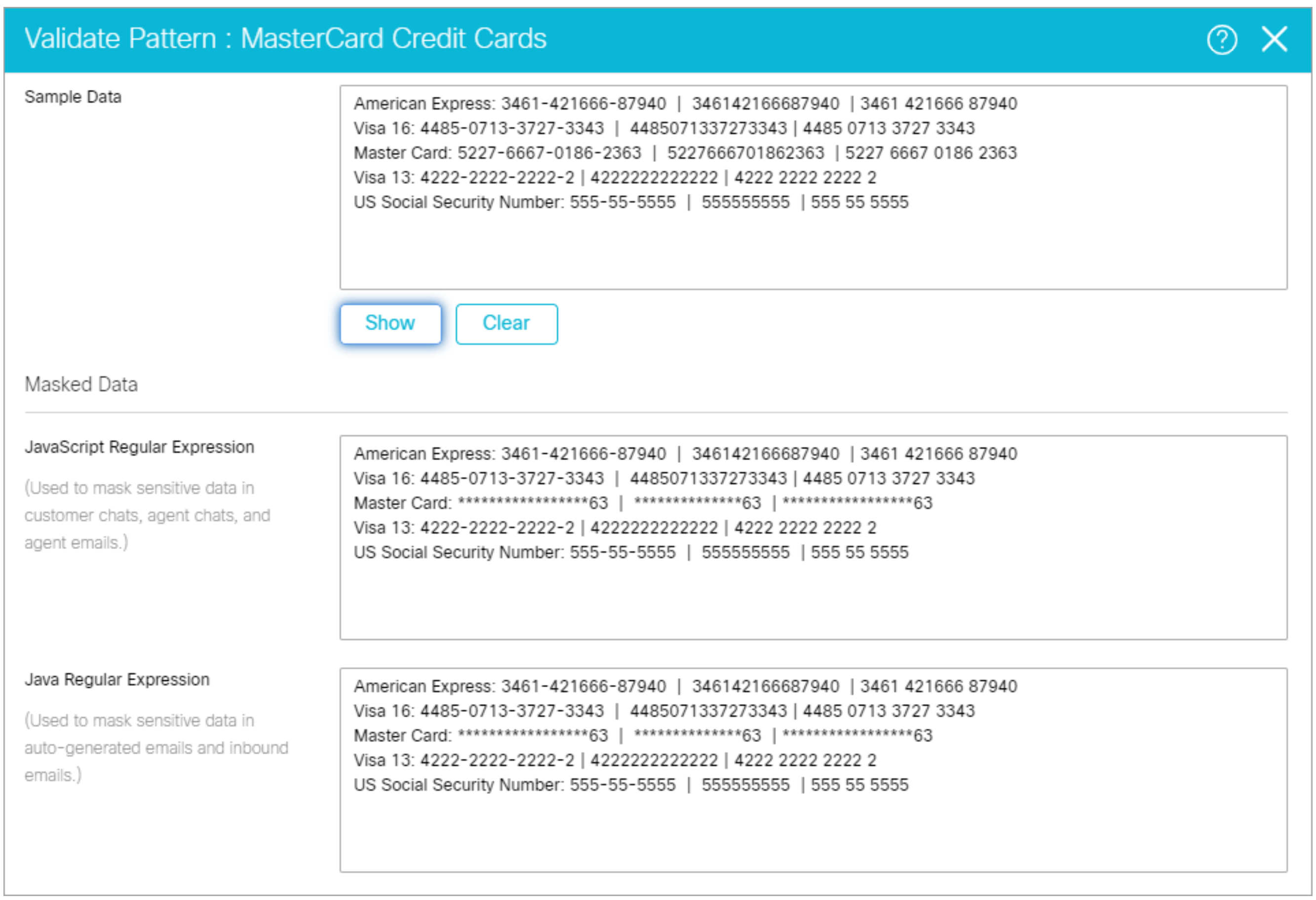
# Validating Masking Patterns

## Validating Individual Patterns

After you create a pattern, test it by using the validation option available for each pattern.

### To validate a pattern:

1. In the global-level Top Menu, click the **Security** option.
2. In the Left menu, navigate to **Data Masking > Patterns**.
3. Select the pattern you want to test.
4. Click the **Validate** button.
5. In the Validate Pattern window, do the following:
    a. In the Sample Data provide the text you want to use for testing the pattern and click the **Show** button.
    b. In the Masked Data section, you will see the JavaScript Regular Expression and Java Regular Expression applied to the sample data. All the settings configured in the Masking Pattern tab will be applied to the sample data.
    c. After you are done testing, click the **Close** button.




## Validating Masking Patterns Applied to Channels

In addition to validating individual patterns, you can validate the patterns selected for a channel and make sure that they work properly as a group and the order of the selected patterns is correct.

### To validate patterns applied to channels:

1. In the department-level Top Menu, click the **Apps** option.
2. In the Left menu, navigate to **Email > Data Masking** or **Chat & Messaging > Data Masking**
3. Click the **Validate** button.
4. In the Validate Pattern window, do the following:
   a. In the Sample Data provide the text you want to use for testing the pattern and click the **Show** button.
   b. In the Masked Data section, you will see all the selected patterns applied to the sample data.
   c. After you are done testing, click the **Close** button.

# Applying Patterns to Chat Channel

## At the Partition Level

A partition administrator with the following actions can perform this task:

- **Manage Application Security:** Allows you to view the patterns applied to channels and to apply patterns to channels.
- **View Application Security:** Gives a read-only view of the patterns applied to channels. Users with this action cannot change any configurations.

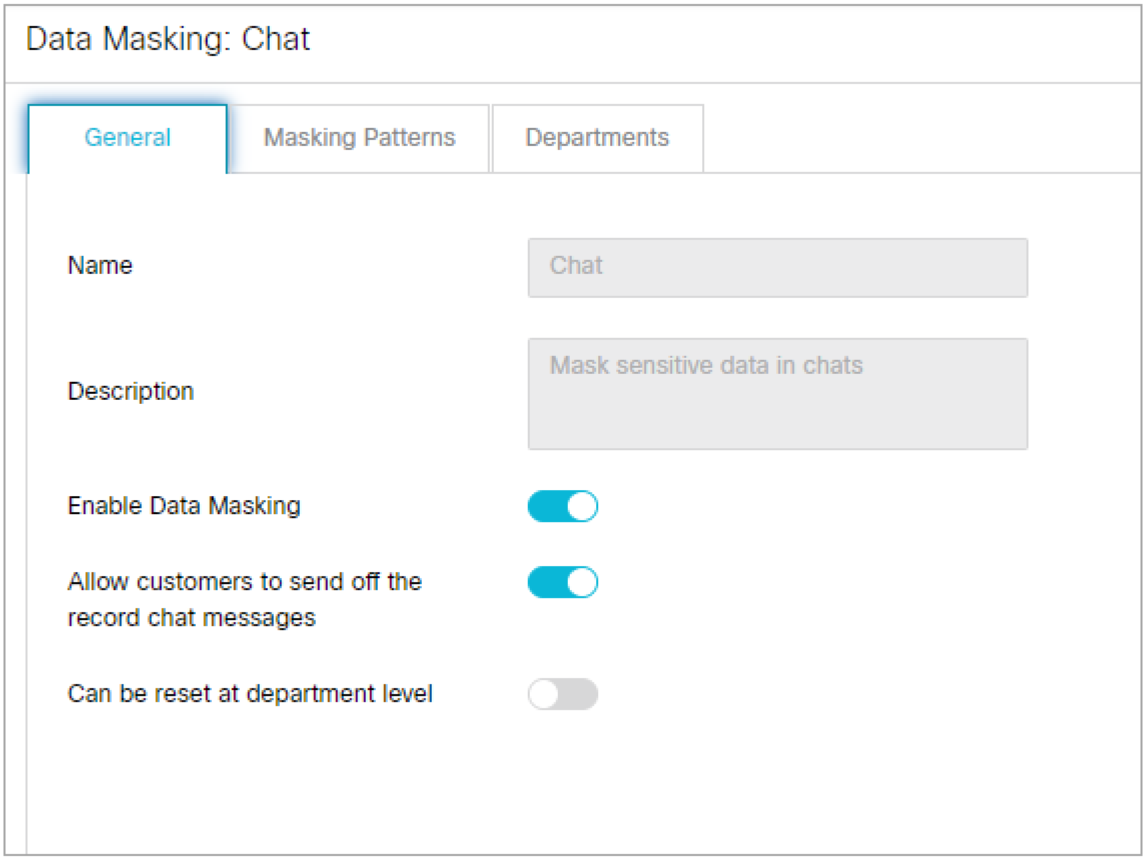
### What can the administrator do?

- Enable data masking for chat for all departments and manage all configurations from the partition level.
- Give control to the department administrators to configure their own settings. At this point, department administrators can choose to configure their own settings or can continue to use the settings configured by the partition administrators. Once a department administrator decides to configure their own settings, they are not affected by the changes made by the partition administrator.

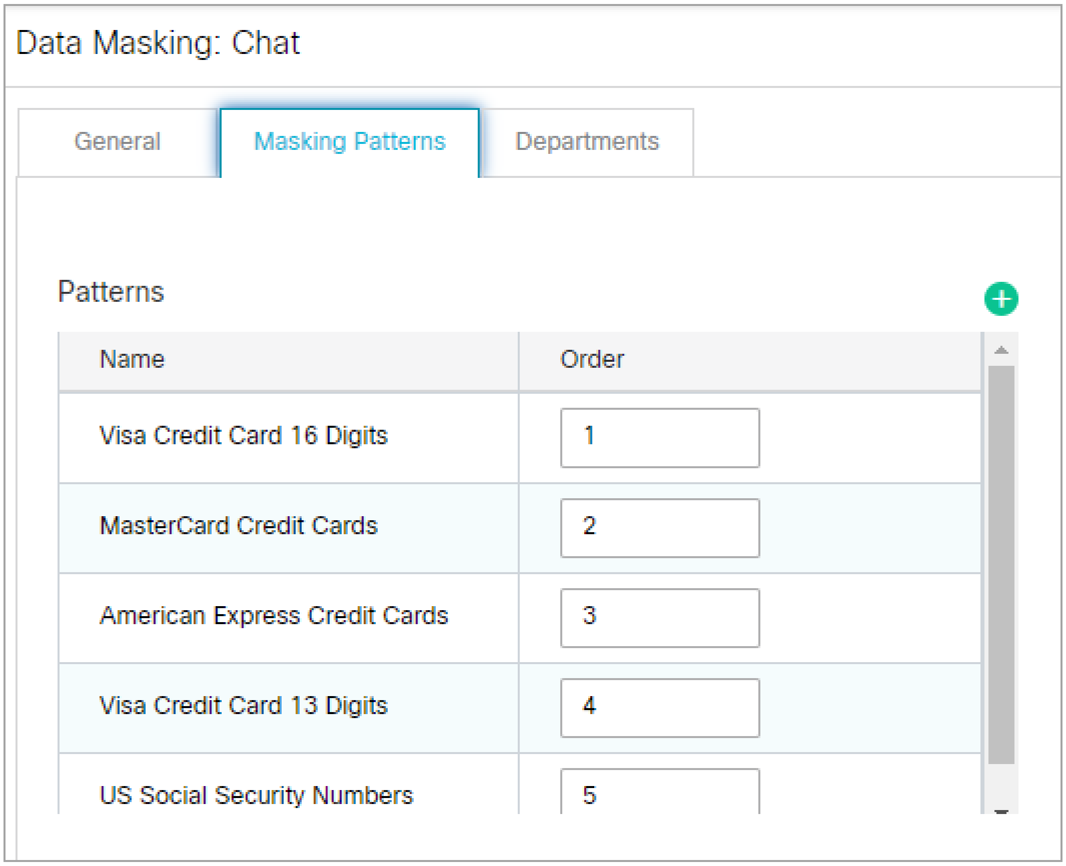
### To apply patterns to the chat channel:

1. In the Global-level Top menu, select the **Security** option.
2. In the Left menu, navigate to **Data Masking > Chat.**
3. In the Data Masking: Chat space, under the General tab, adjust the following fields:
   - **Name:** This field is disabled and cannot be changed.
   - **Description:** This field is disabled and cannot be changed.
   - **Enable Data Masking:** Click this toggle button to enable or disable data masking for chat.

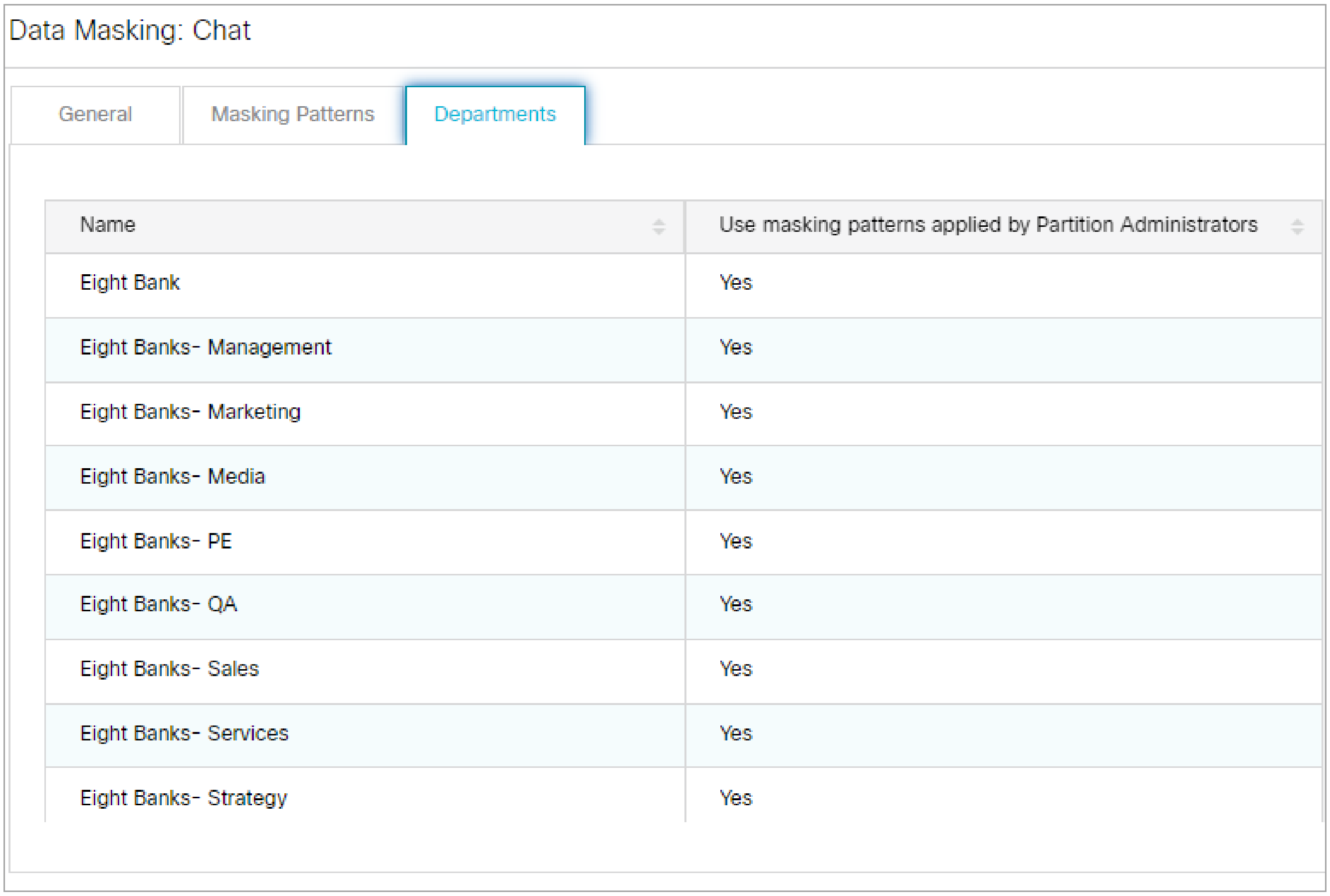
- **Allow customers to send off the recorded chat messages:** Click the Toggle button to enable "off-the-record" functionality for chats, allowing customers to send messages to agents in chats that ignore data masking patterns. Messages that are sent while off the record are not recorded or archived.
- **Allow Resetting at the department level:** Click the Toggle button to allow data masking settings to be configured at the department level. If this toggle remains disabled, the settings configured at the partition level apply to all departments.




4. On to the Masking Patterns tab, click the **Search and Add** button and select the patterns to be applied to the chat channel and define the order of the patterns. While defining the order make sure that the longest pattern is on top followed by the short patterns. This ensures that patterns that use smaller matches do not partially mask the text that would match the longer text. For example, if you are selecting both Visa Credit Card 16 Digits and Visa Credit Card 13 Digits, make sure the order is - Visa Credit Card 16 Digits and then Visa Credit Card 13 Digits. If the order is not correct, and let us say you mask the following content: Visa 16: 4485-0713-3727-3343 and Visa 13: 4222-2222-2222-2, it will get masked as Visa 16: ****************343 and Visa 13: ****************. You will notice that the 16 digit credit card did not get masked properly.



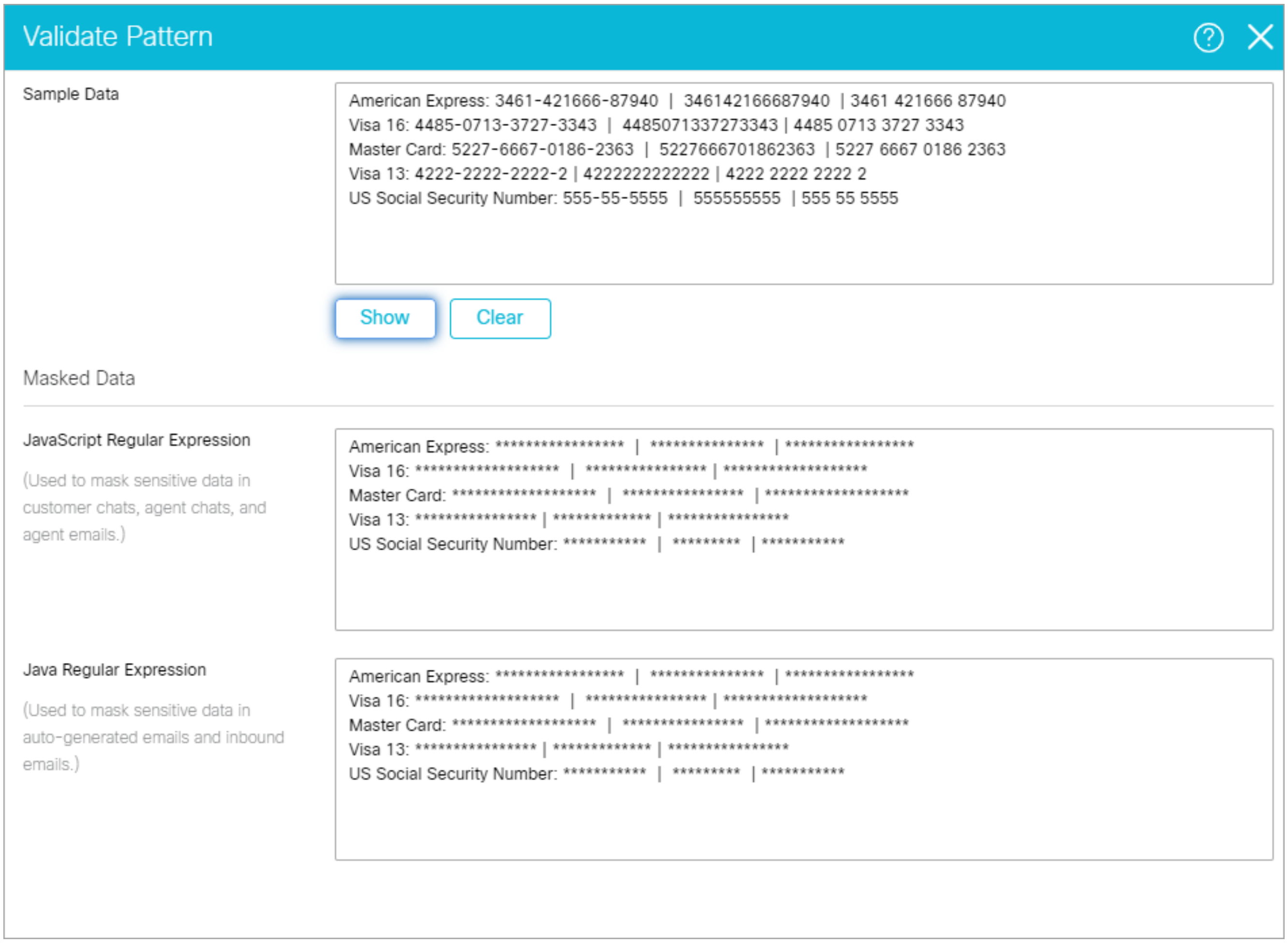

5. Next, go to the Departments tab to see a read-only view of the departments that are using the masking patterns applied by the partition administrator.


Data Masking: Chat

General | Masking Patterns | Departments

| Name | Use masking patterns applied by Partition Administrators |
| --- | --- |
| Eight Bank | Yes |
| Eight Banks- Management | Yes |
| Eight Banks- Marketing | Yes |
| Eight Banks- Media | Yes |
| Eight Banks- PE | Yes |
| Eight Banks- QA | Yes |
| Eight Banks- Sales | Yes |
| Eight Banks- Services | Yes |
| Eight Banks- Strategy | Yes |

6. Click the **Save** button.
7. After saving the changes, validate the patterns selected for the channel by using the Validate button.




## At the Department Level

A department administrator with the following actions can perform this task:

- **Manage Department Security:** Allows you to view the patterns applied to channels and to apply patterns to channels.
- **View Department Security:** Gives a read-only view of the patterns applied to channels. Users with this action cannot change any configurations.

## How much control do department administrators get?

- If the partition administrator has not given control to department administrators to configure their own settings, the department administrators get a read-only view of the settings configured by the partition administrator.
- If the department administrator has the option to configure their own settings, and they choose to do so, they are not affected by the changes made to the configurations by the partition administrators.

### To apply patterns to the chat channel:

1. In the department-level Top menu, select the **Apps** option.
2. In the Left menu, navigate to **Chat & Messaging > Data Masking.**

3. In the Data Masking: Chat space, select the **Enable Department Settings** toggle switch to enable data masking configuration for the department. If this toggle is enabled, you may configure and adjust the fields under the General and Masking patterns tabs.

4. On the General tab, set the following:

   - **Name:** This field is read-only.

   - **Description:** This field is read-only.

   - **Enable data masking:** Click the toggle to enable data masking for chat messages. By default this is set to **Off**.

   - **Allow customers to send off the record chat messages:** Click this toggle to allow customers and agents to exchange off the record messages. Data masking rules do not apply to such messages. During a chat, only the customer has the option to enable off-the-record feature. All messages exchanged in this mode are not stored in the ECE system. By default this is set to **Off.**

5. On to the Masking Patterns tab, click the **Search and Add** button and select the patterns to be applied to the chat channel and define the order of the patterns. While defining the order make sure that the longest pattern is on top followed by the short patterns. This ensures that patterns that use smaller matches do not partially mask the text that would match the longer text. For example, if you are selecting both Visa Credit Card 16 Digits and Visa Credit Card 13 Digits, make sure the order is - Visa Credit Card 16 Digits and then Visa Credit Card 13 Digits. If the order is not correct, and let us say you mask the following content: Visa 16: 4485-0713-3727-3343 and Visa 13: 4222-2222-2222-2, it will get masked as Visa 16: ****************343 and Visa 13: ****************. You will notice that the 16 digit credit card did not get masked properly.

6. Click the **Save** button.

7. After saving the changes, validate the patterns selected for the channel by using the Validate button.

# Call Variables



## About Call Variables

While sending new activity requests from a queue to Unified CCE, EAAS sends call variables and ECC variables to Unified CCE as task context. By default, the following activity attributes are sent to Unified CCE as ECC variables.

- For inbound, outbound, and email activities: activity_id
- For callback and delayed callback activities: activity_id, customer_name, cmb_param, cti_strategy

If you need to pass on other attributes of the activity as call variables or ECC variables to Unified CCE, you need to configure them in ECE. These variables can then be used in Unified CCE scripts to configure conditions. For details, see the Unified CCE scripts documentation. If you plan to configure these variables as ECC variables in ECE, you need to first create the ECC variables in Unified CCE. For details, see the Unified CCE documentation.

The following attributes are available out-of-the-box: `contact_point_data`, `delay_time_in_min`, `subject`, `is_escalated`, `conference_flag`, `case_type`, `last_action_reason`, `queue_id`, `description`, `assigned_to`, `user_last_worked`, `due_date`, `when_modified`, `when_created`, `activity_priority`, `customer_phone_no`, `activity_id`, `customer_name`, `cmb_param`, `cti_strategy`

You can create variables for custom activity attributes.

## Creating Call Variables

### To create a call variable:

1. In the department-level Top Menu, click the **Business Rules** option.
2. In the Left menu, navigate to **Call Variables**.
3. Click the **New** button.
4. Enter the required information into the fields listed below:
    - **Name:** Type the name for the call variable. Do not use any special characters in the names of call variables

- **Mapped Custom Attribute:** From the dropdown list, select a custom attribute created for the Activity Data.
- **Description:** Provide a brief description of the call variable.


5. Click the **Save** button.

> Once the variable has been save, you must restart the EAAS service process and instance.

# Deleting Call Variables

> Make sure you do not delete a call variable that is in use.

### To delete a call variable:

1. In the department-level Top Menu, click the **Business Rules** option.
2. In the Left menu, navigate to **Call Variables**.
3. In the Call Variables workspace, hover your mouse over the call variable you wish to remove and click the **Delete** button.
4. Click the **Yes** button when the system prompts you to confirm the deletion.

# Queues



## About Queues

Queues hold incoming customer service activities such as emails and chat sessions that are waiting to be assigned to agents. There is no limit to the number of queues that can be created in a department. A single queue can hold multiple activity types like email, call, chat, and so on. Agent access to queues is controlled through permissions.

Assume that a sports goods company receives email queries that can be categorized into one of the following: Orders, Replacements, Billing details, or Shipment details.

The administrator for that company can create four queues - Orders, Replacements, Billing, and Shipment, and using workflows, route each type of query to its respective queue. Agents with expertise in a particular area are given permissions to access the appropriate queue. In this example, an agent may have access to the Shipment queue, but not the Replacements one.

You can and should create different queues for each type of activity that is processed by the system. If the system is being used to handle email and chat type activities, create two independent queues, one for email, and one for chat, and configure permissions as required.

# Types of Queues

There are the following types of queues:

- **Enterprise Chat and Email:** Routing of activities from this type of queue handles email and chat activities. There are two methods in which activities are routed by standalone queues:
    - **Load balanced:** Activities are routed from the queue to available agents, who have least number of activities in their inbox.
    - **None:** Activities are not routed and remain in the queue to be picked or pulled from agents. Agents need permissions to pull activities from the queue.
- **Unified CCE:** This is a Unified CCE type of queue and the routing of activities from this queue is done by Unified CCE. This queue is mapped to a Unified CCE MRD and a script selector. When the MRD or script selector attributes are modified in Unified CCE, the modifications are automatically retrieved and synchronized in ECE when an administrator clicks the name of the queue in the Administration Console. If the MRD or script selector is deleted from Unified CCE, the queue in ECE is unmapped when a user clicks the name of the queue in the Administration Console, and the administrator is shown a message saying that the queue has been unmapped.

# Queues Created by the System

## Exception Queue

By default, an exception queue is created in every department. Activities are routed to the exception queue when:

- There are no active inbound workflows in the department.
- A workflow faces an error while processing activities.
- The queue used in a workflow is made inactive. All the activities coming to the inactive queue are routed to the exception queue.
- Emails are bounced back.
- Emails are not parsed by the retriever. If the email subject contains a closed case ID, the case matching the ID is reopened and the activity is associated with it. If the ID matches an open case, the email activity is associated with it. If the ID does not match any cases, a new case is created and associated with the activity.
- A `new_task_failure` message is returned by ICM.

> The exception queue cannot be deleted or made inactive.

## Supervisory Queues

By default, a supervisory queue is created for each MRD associated with either the email or the outbound media classes. This is a standalone queue in ECE. Administrators must use this supervisory queue as the end node of all outbound workflows that form supervisory loops for integrated users or user groups who are part of that MRD. Integrated users who are assigned the Supervisor or Administrator roles are automatically given permissions to the Supervisory queue. For more information about setting up supervisory loops, see About Workflows.

### Call Queue

A system provided queue for call activities. This queue is not in use.

### Chat Queue

A system provided queue for chat activities. This queue is not in use.

### Email Queue

A system provided queue for email activities. This queue is not in use.

## Creating Chat Queues

### Configure General Properties

#### To configure the general properties of the chat queue:

1. In the department-level Top Menu, select **Business Rules**.
2. In the Left menu, navigate to **Queues**.
3. Click the **New** button to open the Create Queue space. In the General tab, fill in the following information:
    - **Name:** Type a name of the queue. This is required information. The following characters are not allowed in the name: < , . ? : > $ * \ /
    - **Type**: Defines whether the queue is an **Enterprise Chat and Email** queue or a **Unified CCE** queue. Select the appropriate option from the dropdown menu.
    - **Media Routing Domain:** This field is disabled and does not apply to ECE queue types.
    - **Script Selector:** This field is disabled and does not apply to ECE queue types.
    - **Active:** Denotes if the queue is active or inactive.  To make a queue active, click the toggle switch.

        > If this queue is intended to be part of a workflow, then it is highly recommended not to set the queue status to inactive.

    - **Description:** Brief description of the queue
    - **Maximum Task Limit**: Sets the maximum amount of tasks that the queue can hold.  This field is disabled and does not apply to ECE queue types.
    - **Routing Priority:** Determines how the queue will sort and route priority tasks. There are two options:
        - **Due date:** Determines task priority based on the due date of the task.
        - **Longest waiting:** Prioritizes tasks that have been waiting in the queue for the longest amount of time.
    - **Queue Priority:** Marks the priority level for the queue. There are three options: Low, Medium, and High. This field is disabled and does not apply to ECE queue types.

- **Configure Solve:** Solve is available for Enterprise Chat and Email when Knowledge add-ons have been purchased and integrated into the system. For more information, see *eGain Solve for Cisco Companion Guide*.

4. Click the **Save** button. Select the following tabs that apply to this queue to configure more advanced properties. Note that sections are only visible for queues tied to MRDs with those channels, for example, the Chat tab is only visible for a Chat MRD queue

## Configure Chat Properties

### To configure the chat properties of the chat queue:

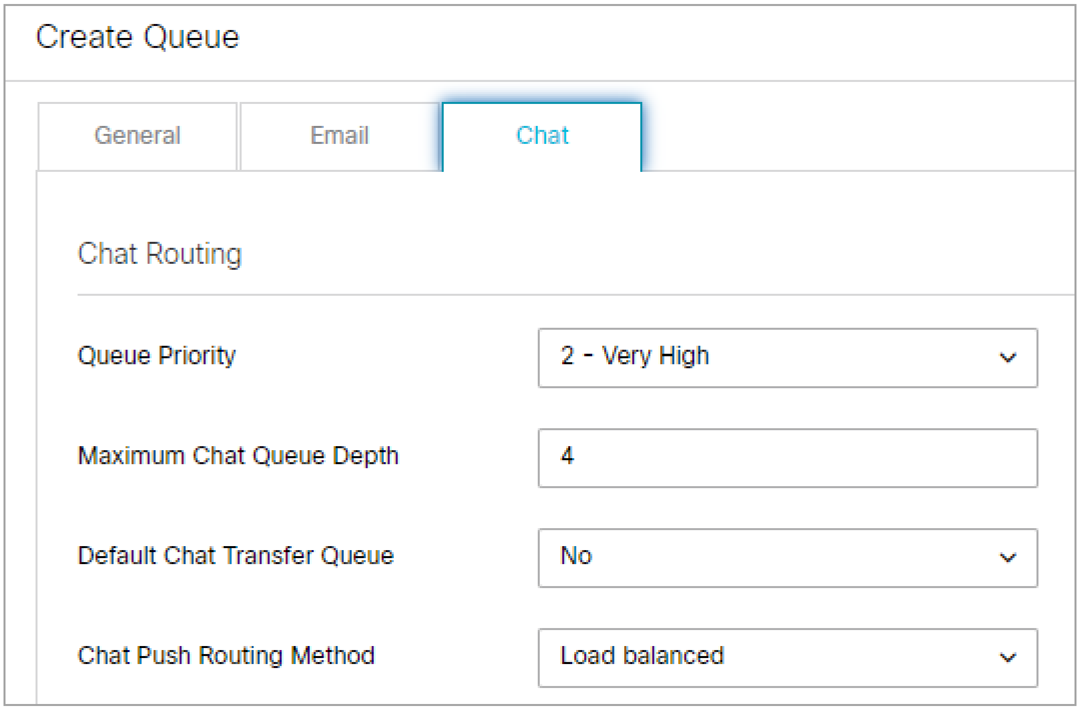
1. Select the **Chat** tab.
2. In the Chat Routing section, set the following:
   - **Queue priority for chats:** Set the priority of the queue. Chats from the highest priority queues will get assigned to agents first. For example, let us say there are three queues with different priorities and chats come in the queues at the times indicated in the following table and one agent is available for handling chats. The service order will be: Jack, Bob, Beth, Jane, Mary, Ron.

| Queue 1, High priority | Queue 2, Medium priority | Queue 3, High priority |
|---|---|---|
| Beth, 8:15 | Ron 8:20 | Jane 8:30 |
| Jack, 8:00 | Mary 8:05 | Bob 8:10 |

   - **Maximum chat queue depth:** Leave this field blank.
   - **Default chat transfer queue:** Once a queue is saved, you can mark it as the default queue for chat transfers by clicking the toggle switch. For details, see Routing chats transferred from other departments.
   - **Chat push routing method:** From the dropdown list, select the method you want to use. There are two options available:
     - **None:** If None is selected, chats are not pushed from the queue to any users. Users have to explicitly pull chats from the queue to service them.

       If you select the push-routing method for the queue as None, make sure that you give users pull  permission on the queue. If you do not give proper permissions on the queue, no user will be able to work on the activities coming in the queue.

     - **Load balanced:** If Load balanced is selected, chats get automatically assigned to the user with the minimum number of open chat activities. In case there is a tie between two or more agents with the same number of chats, the user who has been waiting the longest since any chat was assigned to him gets priority over others. If you select this option, you have to select the list of users and user groups to whom chats must be routed. This is done from the Relationships section in the Chat tab.

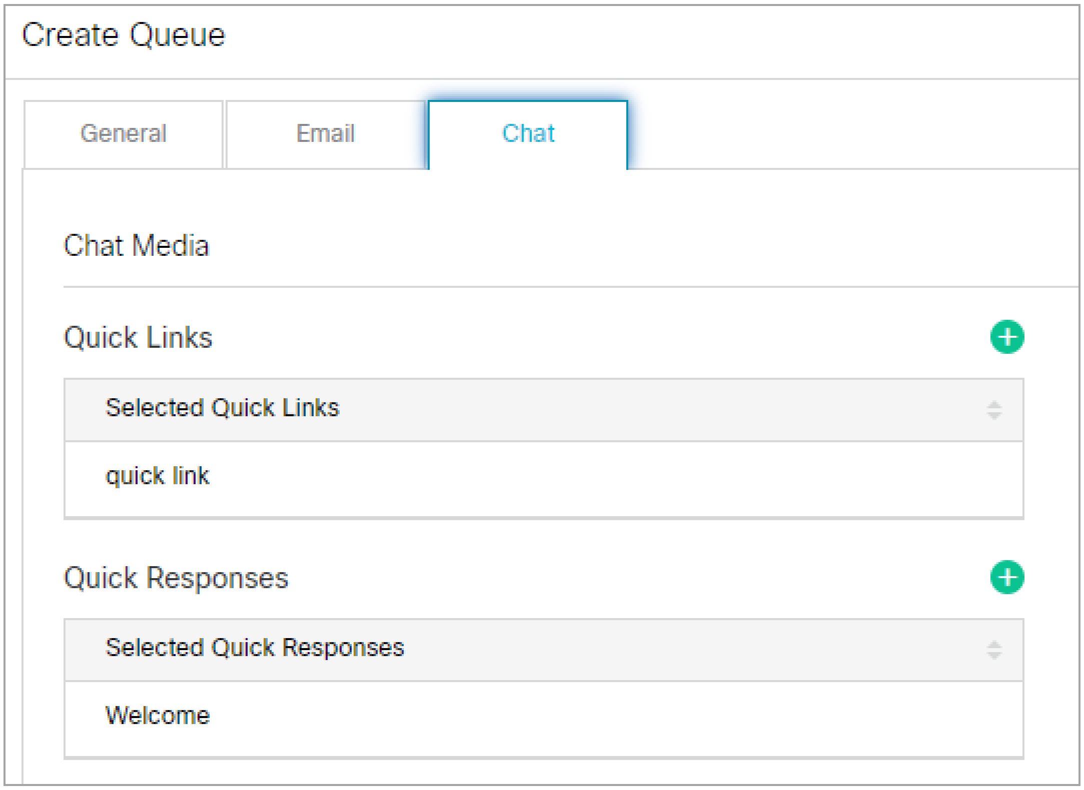



3. When the Chat Push Routing Method is selected as Load Balanced, then the go to the Users and User Groups section and select the users and user groups to whom chat activities are to be assigned from the queue. Chats are routed to agents only when they are available for handling chats.

   - In the User Groups section, click the **Search and Add** button and select the user groups from the available user groups list.
   - In the User section, click the **Search and Add** button and select users from the available users list.

4. In the Media section, set the following.

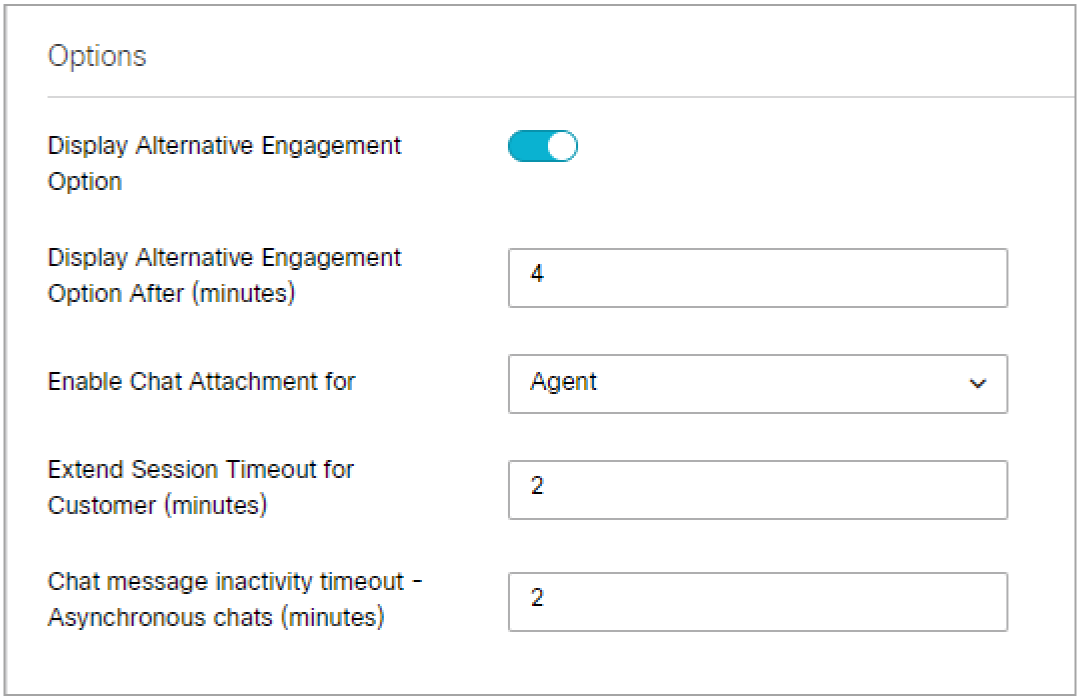
   - **Chat - Quick links:** Click the **Search and Add** button to display a list of quick links. From the available quick links, select the quick links to be associated with the queue.
   - **Chat - Quick responses:** Click the **Search and Add** button to display a list of quick links. From the available quick responses, choose the quick responses to be associated with the queue.




5. In the Options section, set the following:

   - **Display alternative engagement option:** Click the toggle button to display the alternate engagement options in the chat window while customers are waiting to be serviced by agents. By default, this feature is set to inactive. For details about configuring these options, see Configuring Alternative Engagement Options.
   - **Display alternative engagement option after (minutes):** Define the time after which the alternate engagement options will be presented to the customer. If you do not set a value, customers will see the options as soon as they start the chat.
   - **Enable chat attachments for:** If Chat attachments are enabled in the partition, you can select one of the following options. Image attachments sent during chat interactions appear in line. The allowed or blocked file types and the size limits of the chat attachments are configured at the partition level. For more information about configuring attachments settings, see About File Attachments.
     - **Both agents and customers**
     - **Neither agents nor customers**
     - **Agents only**
     - **Customers only**
   - **Extend session timeout for customers (minutes):** Define the period of time in which the system should attempt to reconnect with a customer in the event of a disconnection during the chat. If a chat between an agent and a customer suddenly disconnects, the application attempts to reconnect the agent and customer for the amount of time specified in this field and resets upon a successful connection.

- **Chat message inactivity timeout - Asynchronous chats (minutes):** Define the period of time that is allowed to pass before the system considers the activity inactive and times out for asynchronous chat activities.




6. Click the **Save** button.

# Editing Imported Queues For Chat

Queues can be created by importing the information from their designated MRD from Unified CCE. For more information about importing data, see Importing Data.

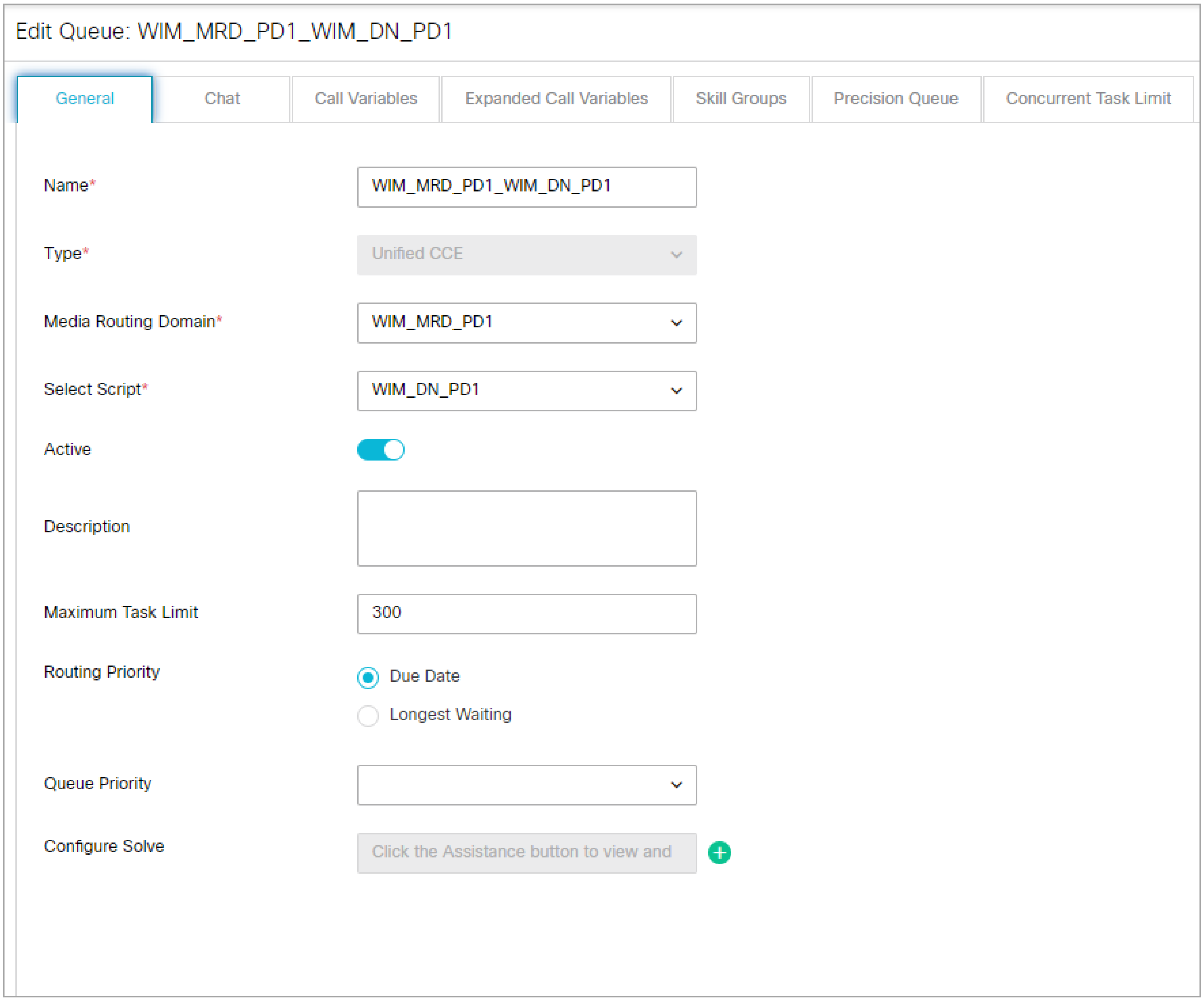
## Configure General Properties

### To configure general properties:

1. In the department-level Top Menu, select **Business Rules**.
2. In the Left menu, navigate to **Queues**.
3. In the Queues space, select the queue you wish to edit.
4. In the Edit Queue space, on the General tab, set the following properties of the queue:
    - **Name:** Type a name of the queue. This is required information. The following characters are not allowed in the name: < , . ? : > $ * \ /
    - **Type:** This is set to Unified CCE and can not be changed.
    - **Media Routing Domain**: Select the MRD to which the queue should map. One MRD can be mapped to only one queue. The list of MRDs is dynamically retrieved from Unified CCE. While creating queues for

outbound emails, make sure you select the MRD created for outbound emails. Similarly, while creating queues for inbound emails, make sure you select the MRD created for inbound emails.

- **Select Script**: From the dropdown list, select a script selector associated with the MRD to which the queue is mapped. It is the keyword that identifies the ICM routing script that is executed in Unified CCE on the activity originating from this queue. An MRD can have multiple script selectors associated with it, but you can select only one script selector for a queue.
- **Active:** When a queue is created, by default it is active. Click the toggle button to make it inactive. If you are using a queue in workflows or are intending to use it, it is advised that you do not make the queue inactive. If a queue is being used in workflows and it is made inactive, no new incoming activities are routed to the queue. All activities coming to an inactive queue are routed to the exception queue. Also, if the queue is being used in workflows, and agents have pull and transfer permissions on the queue, they can pull activities from, and transfer activities to inactive queues. If you want to restrict these actions, remove the pull and transfer permissions on the inactive queue.
- **Description:** Type a brief description.
- **Maximum Task Limit**: This is the maximum number of activities or tasks that can reside within the MRD at any point. It is also referred to as the Unified CCE queue depth or the Maximum calls per queue. The value in this field is set automatically based on the selected MRD and it cannot be changed.
- **Routing Priority**: Select the routing priority for activities waiting in the queue. The value chosen here determines the order in which activities are retrieved from the queue, and which new activity requests are sent for routing to Unified CCE. The options available are:
  - **Due date:** New activity request with the nearest due date is sent to Unified CCE.
  - **Longest waiting:** New activity request with the longest wait time is sent to Unified CCE.
- **Queue Priority**: Select the priority of the queue. While sending new activity requests from a queue, the priority of a queue is checked. If more than one queue has the same priority, than the queues are considered in alphabetic order. The EAAS instance retrieves activities from the highest priority queues first, and the lowest priority queues last. The queue priority can be:
  - Low
  - Medium
  - High
- **Configure Solve:** Solve is available for Enterprise Chat and Email when Knowledge add-ons have been purchased and integrated into the system. For more information, see *eGain Solve for Cisco Companion Guide*.




4. Click the **Save** button.

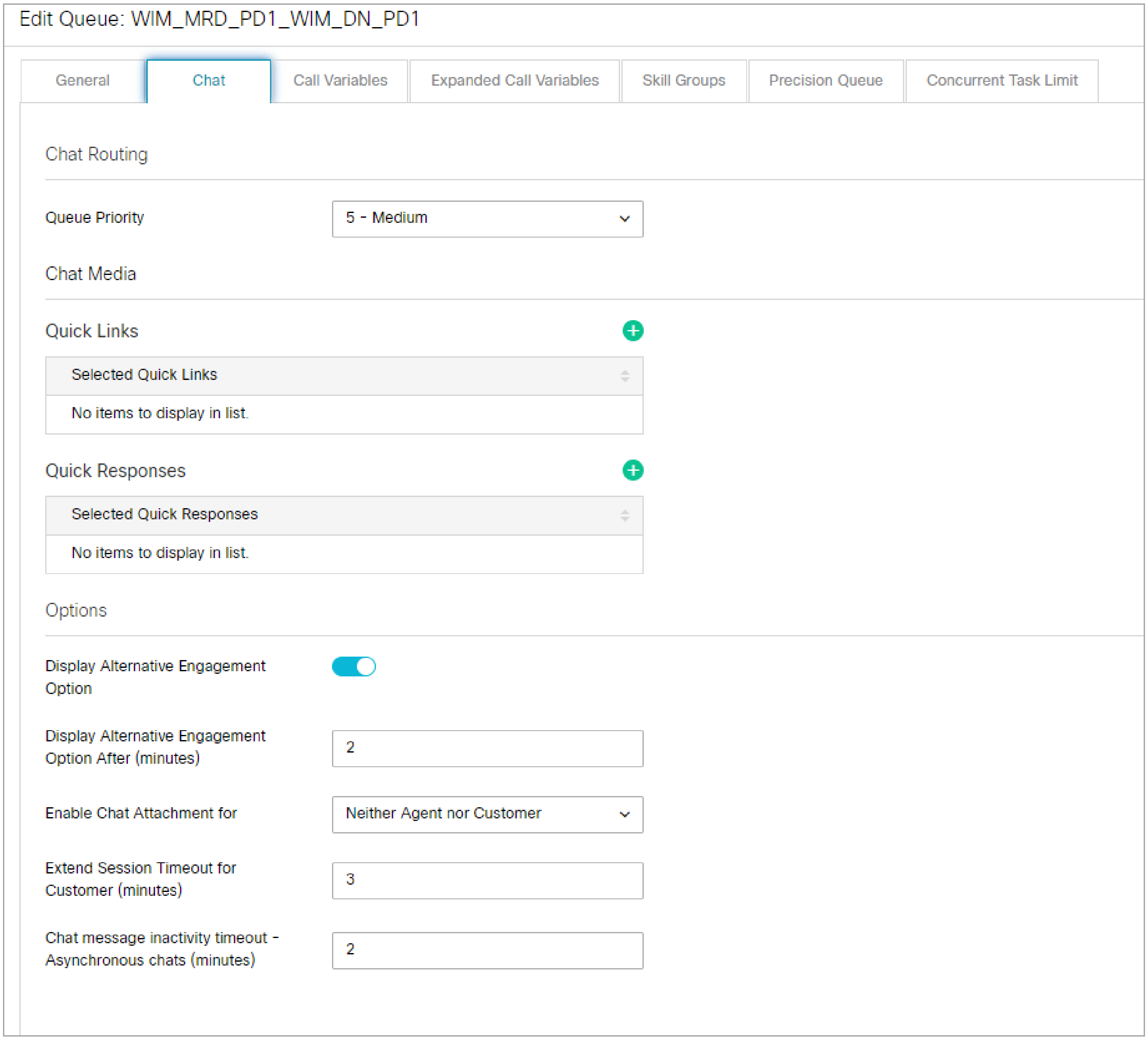
## Configure Chat Properties

Go to the Chat tab and set the following properties. This tab is enabled only for queues that belong to a chat MRD.

### To configure chat properties:

1. In the Chat Routing section, set the following:
    - **Queue Priority:** Set the priority of the queue. Chats from the highest priority queues will get assigned to agents first.
2. In the Media section, set the following.
    - **Chat - Quick links:** Click the **Search and Add** button to display a list of quick links. From the available quick links, select the quick links to be associated with the queue.
    - **Chat - Quick responses:** Click the **Search and Add** button to display a list of quick links. From the available quick responses, choose the quick responses to be associated with the queue.

3. In the Options section, set the following:
    - **Display alternative engagement option:** Click the toggle button to display the alternate engagement options in the chat window while customers are waiting to be serviced by agents. By default, this feature is set to inactive. For details about configuring these options, see Configuring Alternative Engagement Options.
    - **Display alternative engagement option after (minutes):** Define the time after which the alternate engagement options will be presented to the customer. If you do not set a value, customers will see the options as soon as they start the chat.
    - **Enable chat attachments for:** If Chat attachments are enabled in the partition, you can select one of the following options. Image attachments sent during chat interactions appear in line. The allowed or blocked file types and the size limits of the chat attachments are configured at the partition level. For more information about configuring attachments settings, see About File Attachments.
        - **Both agents and customers**
        - **Neither agents nor customers**
        - **Agents only**
        - **Customers only**
    - **Extend session timeout for customers (minutes):** Define the period of time in which the system should attempt to reconnect with a customer in the event of a disconnection during the chat. If a chat between an agent and a customer suddenly disconnects, the application attempts to reconnect the agent and customer for the amount of time specified in this field and resets upon a successful connection.
    - **Chat message inactivity timeout - Asynchronous chats (minutes):** Define the period of time that is allowed to pass before the system considers the activity inactive and times out for asynchronous chat activities.

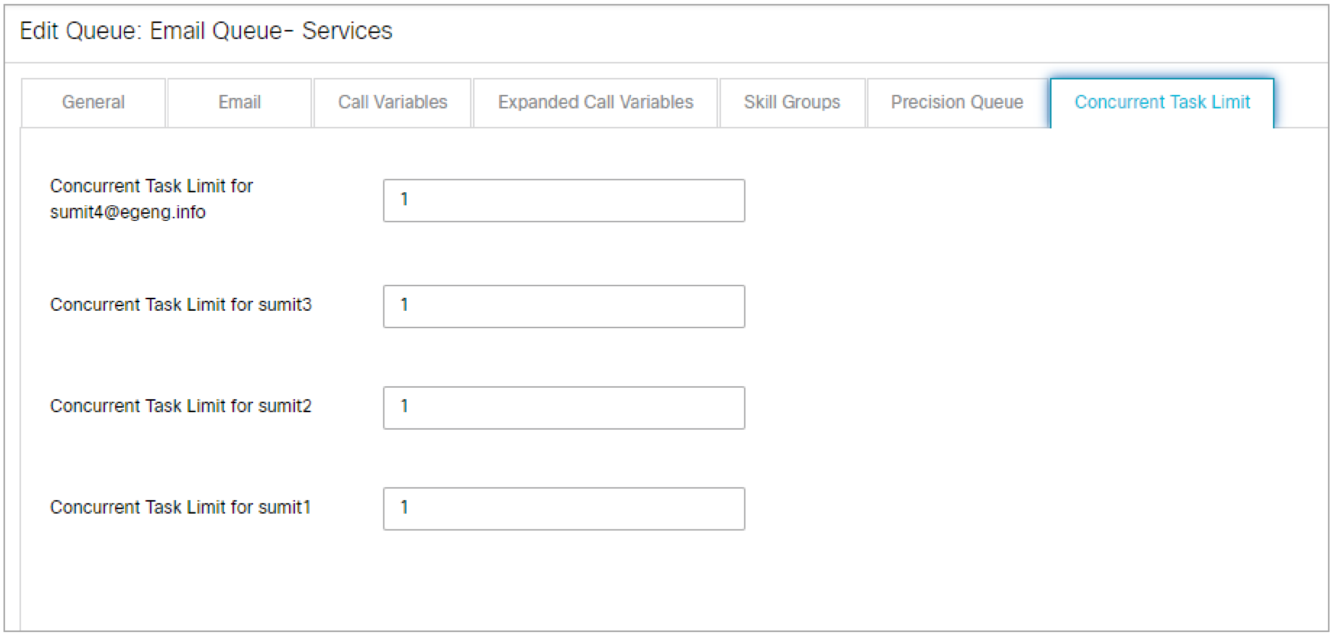



## Configure Concurrent Task Limit

### To configure concurrent task limit:

1. Go to the Concurrent Task Limit tab. For integrated queues for email and chat activities, set the maximum number of tasks that a given agent can handle at a time. This value is also known as the maximum concurrent task limit for an agent. There are two types of agents that are displayed here:
    - Agents who belong to the skill group associated with the Unified CCE MRD selected in the General tab.
    - Agent whose Attribute is used in PQ step and that PQ is mapped to the Unified CCE MRD selected in the General tab.
2. If more than one queue is mapped to the same MRD, then the highest value assigned to the agent for any of those queues will apply as the concurrent task limit. In such cases, users must go to the Concurrent Task Limit tab for each queue mapped to that same MRD to view the actual value that the system considers.

3. The Concurrent Task Limit for a user cannot be greater than the maximum task limit of the queue. The maximum task limit of a queue can be viewed from the **Maximum Task Limit** field on the General tab. The value for this field is configured in the **Max Calls In Queue** in the ICM Configuration Manager > Media Routing Domain List Tool.
4. The concurrent task limit for an agent is valid only as long as the agent remains in the same skill group. When the agent is reskilled, the value is lost. If the agent is added to the same skill group again, this field must be manually updated.




5. Click the **Save** button.

# Picking, Pulling, and Transferring Activities

Queue permissions are assigned automatically based on the MRDs that were imported. These permissions cannot be changed in ECE.

## Important Things to Note About Permissions to Pick and pull Activities

### Emails

- Agents can pick emails from other agents that belong to the same set of skill groups.
- Only agents who are part of a skill group that is associated with the queue can pick or pull from that queue.
- Only agents who match the attributes of a Precision Queue (PQ) that is associated with the ECE queue can pick or pull from that queue.
- Based on the **Maximum Task Limit** setting, agents who have reached their concurrent task limit can pick additional activities. The maximum number of activities is defined as part of the setting.

- When working on a non-interruptible chat or voice call:
  - Agents can pick or pull interruptible emails from queues and other agents.
  - Agents cannot pick or pull non-interruptible emails from queues or other agents.
- Agents with the **Administrator Role** or the **Supervisor Role** can pick from the Default exception queue.

> Emails with exception keywords that are routed to the Default exception queue should not be transferred to other queues. These emails cannot be picked or pulled upon being transferred to other queues.

### Chats

Agents are assigned chats by the system automatically. They cannot pull chat activities from queues. Pick does not apply to chats.

## Important Things to Note About Transferring Emails

- Multiple emails can be selected and transferred to another user or queue at the same time, so long as the emails are new and have no draft responses. If an email has any draft responses, or is not a new incoming email, it must be transferred individually.
- Outbound emails created by agents can only be transferred to users and not to queues.

> For installations that have upgraded from a version prior to 12.0(1), a routing script must be applied to the Pick/Pull node for the outbound MRD in order for agents to pick, pull, or transfer outbound emails. For more information about the script, see the *Scripting and Media Routing Guide for Cisco Unified CCE*.

- Disabled users are not listed in the list of users to whom you can transfer activities.
- You can transfer activities only if you have the **Transfer** action.

### Transferring to Queues:

- An email can be transferred to any queue that belong to the same media class. From there, the activity is routed based on the queue-to-script mapping.

### Transferring to Agents:

- Agents can transfer emails to other agents that belong to the same set of skill groups.
- Based on the **Maximum Task Limit** setting, agents can transfer additional activities to agents who have reached their concurrent task limit. The maximum number of activities is defined as part of the setting.
- Emails cannot be transferred to departments directly. If the **Allow Transfer of Activities to Integrated Queues in Other Departments** setting is enabled, agents can transfer activities to queues of other departments.
- If the **Allow email transfer to agents who are not available** setting is enabled, agents can transfer activities to other agents who are not available to work on new activities. To be able to transfer an email to an agent, the agent must be logged in to the application, should not have met the concurrent task limit, and should not be

working on a non-interruptible activity. If these requirements are not met, the agent is not displayed in the Transfer Activities window.

- If the **Allow email transfer to agents who are not logged in** setting is enabled, agents can transfer activities to other integrated agents who are not logged in to the application. To be able to transfer an email to an agent, the agent should not have met the concurrent task limit. If this requirement is not met, the agent is not displayed in the Transfer Activities window.
- An agent can transfer interruptible email activities to another agent. An agent cannot transfer non-interruptible email activities to another agent. The concurrent task limit of the agent is considered in these instances.

## Important Things to Note About Transferring Chats

- Only one chat activity can be transferred at a time.
- Only open chat activities, in which the customer has not left the chat session, can be transferred.
- Disabled users are not listed in the list of users to whom you can transfer activities.
- You can transfer activities only if you have the Transfer action. For more information about actions and permissions, see the *Enterprise Chat and Email Administrator's Guide to the Administration Console*.

### Transferring Chats to Queues:

- Only agents who match the attributes of a Precision Queue (PQ) that is associated with an ECE queue can transfer chats to that queue.
- Chats cannot be transferred to departments directly. If the **Allow Transfer of Activities to Integrated Queues in Other Departments** setting is enabled, agents can transfer activities to queues of other departments.
- A chat can be transferred to any queue that belong to the same media class. From there, the activity is routed based on the queue-to-script mapping.
- To be able to transfer a chat to a queue, at least one agent who can receive work from that queue must be logged in, must be available, and must not have met the concurrent task limit. The queue must also not be at its maximum task limit.

### Transferring Chats to Agents:

- Agents who do not meet these conditions are not displayed in the transfer window.
- Agents can transfer chats to other agents that belong to the same set of skill groups.
- Only agents who are part of a skill group that is associated with a queue can transfer chats to that queue.
- The receiving agent must be logged in to the application.
- The receiving agent must be available, depending on how the **Allow chat transfer to agents who are not available setting** is configured.
- The receiving agent should not have met the concurrent task limit, unless you are working on non-interruptible chat activities. This may be affected by the **Maximum assignment beyond concurrent task limit** setting.

- Based on the **Maximum assignment beyond concurrent task limit** setting, agents can transfer additional activities to agents who have reached their concurrent task limit. The maximum number of activities is defined as part of the setting.
- If the **Allow chat transfer to agents who are not available** setting is enabled, agents can transfer activities to other integrated agents who are not available to work on new activities. To be able to transfer a chat to an agent, the agent must be logged in to the application. Also, the agent should not be at the concurrent task limit (CTL), and the queue associated with the agent should not be at its maximum task limit (MTL). If the CTL and MTL for the agent have been reached, or if the agent is not logged in, the agent is not displayed in the Transfer Activities window.
- An agent can transfer chat activities to another agent who is working on an interruptible email activity or a non-interruptible chat activity. If the receiving agent is working on a non-interruptible voice call, only interruptible chat activities can be transferred to that agent. Agents working on non-interruptible voice calls cannot be transferred non-interruptible chats.

# Changing the Status of Queues

When a queue is created, by default it is active. If you are using a queue in workflows or are intending to use it, do not make the queue inactive. If a queue used in workflows is made inactive, no new incoming activities are routed to the queue. All activities coming to the inactive queue are routed to the exception queue.

If the queue used in workflows, and agents have pull and transfer permissions on the queue, they can pull activities from, and transfer activities to inactive queues. If you want to restrict these actions, remove the pull and transfer permissions for the inactive queue.

Queues used in chat entry points cannot be made inactive.

### To change the status of a queue:

1. In the department-level Top Menu, select **Business Rules**.
2. In the Left menu, navigate to **Queues**.
3. In the Edit Queue space, on the General tab, click the toggle button next to the **Active** field to make the queue active or inactive.
4. Click the **Save** button.

# Deleting Queues

You can only delete queues for which you have delete permissions.

You cannot delete queues that have open activities, or queues that are being used in workflows or chat entry points, or the queues that are associated with activity sub types. Also, the default queues provided with the system cannot be deleted.

### To delete a queue:

1. In the department-level Top Menu, select **Business Rules**.
2. In the Left menu, navigate to **Queues**.
3. In the Queues space, hover your mouse over the queue you wish to remove and click the **Delete** button.
4. Click the **Yes** button when the system prompts you to confirm the deletion.

> The **Delete** button is enabled only if you have permission to delete a queue.

# Assets For Chat



## Chat Settings

### Activity Assignment Settings

#### Chat Auto-Pushback Settings

The chat auto-pushback feature allows you to pushback chat activities to the queue, if the agents do not click on the new chats assigned to them in the configured time (default value is 2 minutes). You can also automatically mark the agents unavailable when chats are pushed-back from their inbox.

ENABLE AUTO-PUSHBACK OF CHATS

Use this setting to decide if new chats assigned to agents should be automatically pushed back from the agent's inbox if they do not click on the activity in the time defined in the Expiry time for auto-pushback for chats setting.

- **Access level:** Partition settings
- **Default value:** Enabled
- **Value options:** Enabled, Disabled

EXPIRY TIME FOR AUTO-PUSHBACK OF CHATS (MINUTES)

In this setting, define the time, in minutes, after which the new chat assigned to the agent will be automatically pushed back from the agent's inbox, if the agent does not click on the chat in the defined time.

- **Access level:** Partition settings

- **Default value:** 2
- **Minimum value:** 1
- **Maximum value:** 210

#### Make Agent Unavailable on Auto-Pushback of Chats

Use this setting to define if agents should be made unavailable after a chat is pushed back automatically from the agent's inbox. By default this setting is disabled.

- **Access level:** Partition settings
- **Default value:** No
- **Value options:** Yes, No

## Chat - Agent Availability Check Mechanism

This setting determines whether the value set in the "Chat - Agent Availability Buffer Value" setting is an absolute value or a percentage of the total number of agents that belong to the chat queue mapped to an MRD and have the required skill groups.

- **Access level:** Partition settings
- **Default value:** Percentage
- **Value options:** Absolute, Percentage
- **Editable at lower level:** Yes

## Chat - Agent Availability Buffer Value

This setting determines the minimum number or percentage of agents that have to be available for a chat queue mapped to an MRD before a chat offer is presented to a website visitor. When a website visitor becomes eligible for a chat offer, the system checks the number or percentage of available agents, whose assigned skill groups match those of the queue, against the value configured in this setting. If this condition is met, the chat offer is presented to the website visitor.

- **Access level:** Partition settings
- **Default value:** 5
- **Minimum value:** —
- **Maximum value:** —
- **Editable at lower level:** Yes

### Enable Chat Queueing

This allows customers to initiate new chats even when all agents are working at their maximum capacity. The chat requests are then queued in Unified CCE to wait for the next available agents. The maximum time for which a chat is queued is defined by the **Chat Watchdog Interval** setting.

- **Access level:** Partition settings
- **Default value:** Yes
- **Value options:** Yes, No
- **Editable at lower level:** Yes

## Inbox Settings

### Chat - Inbox Sort Column

In this setting, define the column that is used to sort items in the Chat Inbox in the Agent Console. Use the "Chat - Inbox sort order" setting to define whether the items are sorted in the ascending or descending order.

> If you specify a column that is not part of the agent's inbox list or if there is a tie between two activities with the same value for the sorting column, the inbox will then be sorted by the shortcut key.

- **Access level:** Partition settings, Department settings, User settings
- **Default value:** Key
- **Value options:** Key, Activity ID, Case ID, When Created, Customer name, Subject, Activity sub status, Queue name
- **Editable at lower level:** Yes

### Chat - Inbox Sort Order

Use this setting to define the order - ascending or descending, in which items appear in the Chat Inbox in the Agent Console. Use the "Chat - Inbox sort column" setting to determine the column by which items are sorted.

- **Access level:** Partition settings, Department settings, User settings
- **Default value:** Descending
- **Value options:** Ascending, Descending
- **Editable at lower level:** Yes

### Chat - My Monitor - Activity Refresh Interval (seconds)

In this setting configure the time interval (in seconds) at which the chat activities are refreshed in the My Monitor's folder of the supervisor's Agent Console. The following details of chat activities are refreshed - the list of activities for the queue or agent being monitored; the transcript of chats that the supervisor has not joined and is monitoring passively.

- **Access level:** Partition settings, Department settings

- **Default value:** 30
- **Minimum value:** 30
- **Maximum value:** 600
- **Editable at lower level:** Yes

### Chat - Enable Sound Alert

Use this setting to decide if you want play a sound alert to draw the agent's attention to the chat inbox when a new chat is assigned to the agent, or a new message is sent by the customer. The sound alert is played only when the Agent Console is minimized or not in focus. If the agent is already working in the Agent Console, the sound alert is not played.

- **Access level:** Partition settings, Department settings
- **Default value:** Yes
- **Value options:** Yes, No
- **Editable at lower level:** Yes

## Activty Handling Settings

### Chat - Force Resolution Code

Use this setting to ensure that agents assign resolution codes to each chat activity before completing it.

- **Access level:** Partition settings, Department settings
- **Default value:** No
- **Value options:** Yes, No
- **Editable at lower level:** Yes

### Chat - Force Activity Categorization

Use this setting to ensure that agents assign categories to each chat activity before completing it.

- **Access level:** Partition settings, Department settings
- **Default value:** No
- **Value options:** Yes, No
- **Editable at lower level:** Yes

### Chat - Agent Chat Message Maximum Length

Use this setting to determine the maximum length of messages sent by agents to customers.

- **Access level:** Partition settings, Department settings
- **Default value:** 800

- **Minimum value:** 60
- **Maximum value:** 2000

### Chat - Reason for Transfer

Use this setting to decide if you want agents to always assign a transfer code to chat activities while transferring chats to other users, queues, or departments.

- **Access level:** Partition settings, Department settings
- **Default value:** Optional
- **Value options:** Optional, Required
- **Editable at lower level:** Yes

### Show Smiley in Agent Chat Toolbar

The toolbar in the Chat pane has a Smiley button that can be used to add emoticons in the chat messages. Use this setting to determine if this Smiley button should be available to the agents.

- **Access level:** Partition settings, Department settings
- **Default value:** Yes
- **Value options:** Yes, No
- **Editable at lower level:** Yes

### Chat - Disable Typing Area and Page Push Area on Customer Exit

Use this setting to disable Page Push and the typing area of the Chat pane for agents and supervisors, when a customer leaves the chat session.

- **Access level:** Partition settings, Department settings
- **Default value:** No
- **Value options:** Yes, No
- **Editable at lower level:** Yes

## Common Chat Settings

### Chat - Display Timestamp in Agent Chat Console

Use this setting to decide if the timestamp should be displayed with the chat messages in the Agent Console. This setting applies to open chat activities only.

- **Access level:** Partition settings, Department settings
- **Default value:** No

- **Value options:** Yes, No
- **Editable at lower level:** Yes

### Chat - Display Timestamp in Completed Chat Transcript

Use this setting to decide if the timestamp should be displayed with the chat messages in the Agent Console. This setting applies to completed chat activities only.

- **Access level:** Partition settings, Department settings
- **Default value:** Yes
- **Value options:** Yes, No
- **Editable at lower level:** Yes

## Chat Service Level Settings

### Chat - SLA for Response Time (Seconds)

This setting is required for the, Chat - Current service level (%) and Chat - Daily service level (%), queue-monitoring attributes, viewed from the Supervision Console. With this setting you can decide the threshold interval (in seconds) that all in-progress sessions are checked against, to measure what percentage had a wait time lesser than the threshold. Any session picked up after a wait time lesser than this threshold is counted as having met the service level. The service level is shown as an aggregate percentage based on how many sessions have met the service level and gives an indication of the timely pick-up of sessions by agents. If this value is set to blank, then the “Chat - Current service level (%)” and “Chat - Daily service level (%)” attributes will show a value of 100% for all queues. The default value is 600.

- **Access level:** Partition settings, Department settings
- **Default value:** 600
- **Minimum value:** 0
- **Maximum value:** 3600
- **Editable at lower level:** Yes

### Chat - Daily Service Level Sample Set Definition

This setting defines if the abandoned chat activities should be considered while calculating the daily service level for chats.

- **Access level:** Partition settings, Department settings
- **Default value:** All chats handled including abandoned
- **Value options:** All chats handled including abandoned, All chats handled excluding abandoned
- **Editable at lower level:** Yes

### Chat - Daily Service Level Timezone

Use this setting to select the time zone to be used for the daily service level for chats.

- **Access level:** Partition settings, Department settings
- **Default value:** (GMT-05:00) Easter Standard Time (US and Canada)
- **Editable at lower level:** Yes

## About Chat Articles

Articles are powerful tools that can be crafted and added to an agent's toolset when handling chats. Articles save agents time by providing pre-written information that can be used to quickly respond to customer inquiries without having to individually type out a standard response or URL. The types of chat articles available are:

- **Quick Links:** Articles that contain URLs that can be immediately available for the agent to send to a customer, such as frequently used online websites or resources.
- **Quick Responses:** Brief messages that are frequently used for common interactions with customers, such as greetings, good-byes, requests for patience, and so on.

## Creating Quick Links

### To create a quick link:

1. In the department-level Top Menu, click the **Apps** option.
2. In the Left menu, navigate to **Chat & Messaging > Quick Links**.
3. In the Quick Links workspace, click the **New** button.
4. In the Create Quick Link space, provide the following:
    - **Name:** A name for the quick link. This is what is displayed in the agent's desktop when the agent selects a quick response.
    - **Description:** A description of the article.
    - **Language:** The language of the article. This is set to the default language and cannot be changed. For more information about changing the language, see Setting a Primary Language.
    - **Content:** Provide the content for the article. Since this is a quick link, this should be a URL web address. You can also use the **Find and Replace** button to perform a quick search of the content and make any necessary changes.
5. You can add notes to the article as well.
6. Click the **Preview** button to preview the presentation of the article. When you are satisfied, click the **Save** button.

# Creating Quick Responses

### To create a quick response:

1. In the department-level Top Menu, click the **Apps** option.
2. In the Left menu, navigate to **Chat & Messaging > Quick Responses**.
3. In the Quick Responses workspace, click the **New** button.
4. In the Create Quick Response space, under the General tab, provide the following:
    - **Name:** A name for the quick response. This is what is displayed in the agent's desktop when the agent selects a quick response.
    - **Description:** A description of the article.
    - **Language:** The language of the article. This is set to the default language and cannot be changed. For more information about changing the language, see Setting a Primary Language.
5. Click the **Content** tab. In the authoring area, provide the content for the response. You can use the **Find and Replace** button to perform a quick word search and replace them with the desired terms.
6. You can add notes to the article as well.
7. Click the **Preview** button to preview the presentation of the article. When you are satisfied, click the **Save** button.

# Copying Articles

### To copy an article:

1. In the Chat or Email article workspace, in the Actions column, click the **Options** ⋯ button for the article you wish to copy.
2. Select the **Create Copy** option.
3. Edit the Name, Description, and Content of the article as needed and click the **Save** button when finished.

# Editing Articles

### To edit an article:

1. In the Chat or Email article workspace, click the article you wish to edit
2. In the Edit space, adjust the Name, Description, and Content of the article as needed. For more information about authoring articles and what type of content to provide, see About Chat Articles and About Email Articles.
3. Click the **Preview** button to preview the presentation of the article. When you are satisfied, click the **Save** button.

# Deleting Articles

### To delete an article:

1. In the Chat or Email article workspace, in the Actions column, click the **Options** ⋯ button for the article you wish to delete.
2. Select the **Delete** option.
3. In the window that appears, click the **Yes** button to confirm the deletion.

# Setting the Language of Articles

### To set the language of articles:

1. In the Chat or Email article workspace, click the **Actions** button.
2. Select the **Change Language** option.
3. In the Change Language window, select a language from the **Select Knowledge Base Language** dropdown.
4. When you have made your selection, click the **Done** button. The language you have selected is now applied to all new chat articles that you author.

# Knowledge Settings

## eGain Knowledge System

> This setting only applies to systems that have purchased Knowledge add-ons.

Use this setting to identify an external knowledge system as the source of knowledge for configuration with digital channels, such as chat or email. The URL for the knowledge system must be provided here in order to configure queues to use external knowledge bases for Solve.

- Access Level: Partition settings; Department settings
- Default value: —
- Minimum: —
- Maximum: —
- Editable at lower level: Yes

## KB Primary Language

Designates the primary language for the Knowledge Base (KB). This setting does not appear in the Language Tools section and must be set here. To add additional languages from the language pack, click the **Search and Add** button, then select the desired language from the popup window that appears.

- **Access level:** Partition settings, Department settings
- **Default value:** —
- **Value options:** English (US), English (UK), Arabic, Chinese (Simplified), Chinese (Traditional), Czech, Danish, Dutch, Finnish, French, German, Greek, Hungarian, Italian, Japanese, Korean, Norwegian (Bokmal), Norwegian (Nynorsk), Portuguese, Portuguese (Brazilian), Romanian, Spanish, Swedish, Turkish
- **Editable at lower level:** Yes

## Custom Language Label

This setting allows you to add a custom language to the list of languages available in the KB primary language setting.

- **Access level:** Partition settings, Department settings
- **Default value:** Custom
- **Minimum:** 0
- **Maximum:** 225
- **Editable at lower level:** Yes

# Aria Chat Templates

# About Aria Chat Template Sets

Chat templates are derived from a predefined set of base templates to create pages.

Templates are built using AngularJS framework. They make use of HTML5, Javascript, CSS, and Less. A template consists of a set of pages that are designed to work together.

A page is rendered based on a layout of containers and components configured in those container. It is the final view that the end user sees with a URL that allows a user to access it. Types of pages include: the login page (pre-chat page), interaction page, off-hours page, survey page (post-chat page), thanks message page, error page, and so on.

The CSS files control the look and feel of the Chat Console. The JS files contain the business logic used to render data in the Chat Console. Templates are also used to determine what information is requested to identify the customer (such as name, email address, phone number). Messages that the customer sees under certain circumstances (for example, if they request a chat session out of hours) can be composed as well.

## What Makes up an Aqua Template?

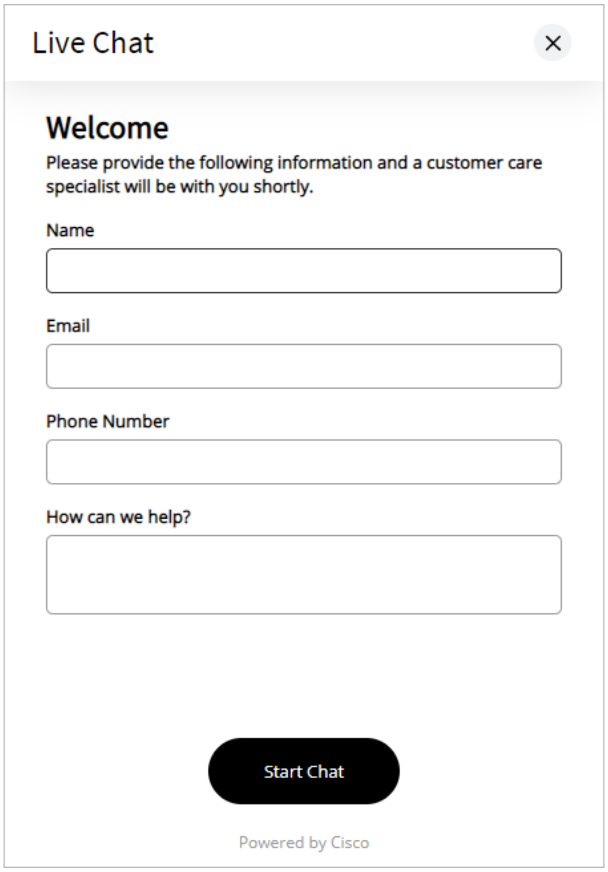
A template consists of a set of pages that are designed to work together. Typically, a chat template has the following pages:

- A pre chat form
- The chat area
- Post chat survey form
- The off-hours page

All these pages can be customized to meet your business needs.

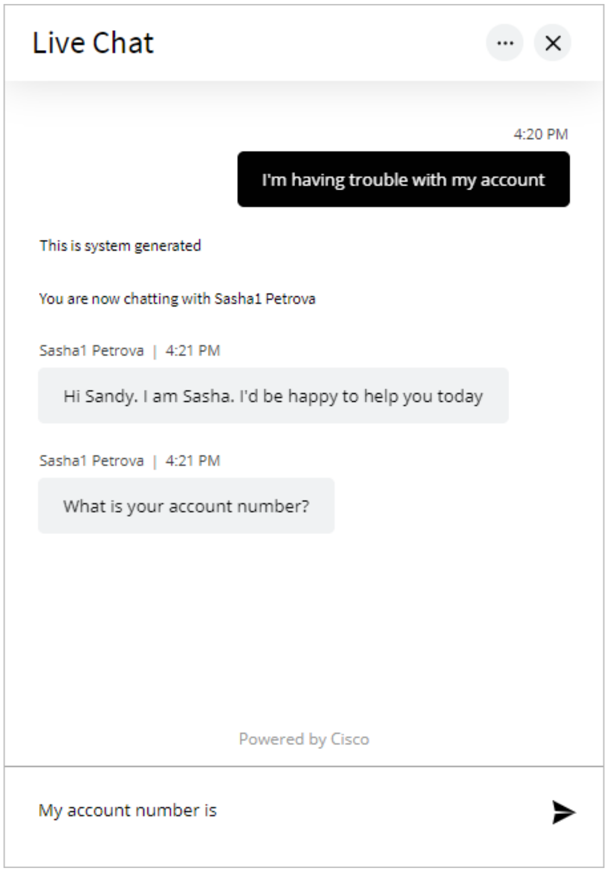
### Pre Chat Form

Customers fill up this form to start a chat.




## Chat Area

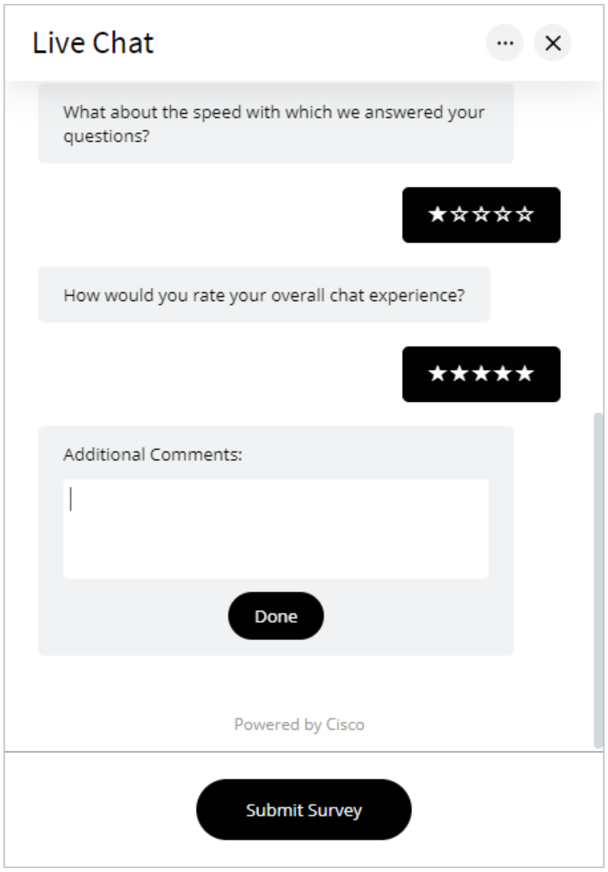
Agent and customer exchange chat messages in the chat area.




## Post Chat Survey

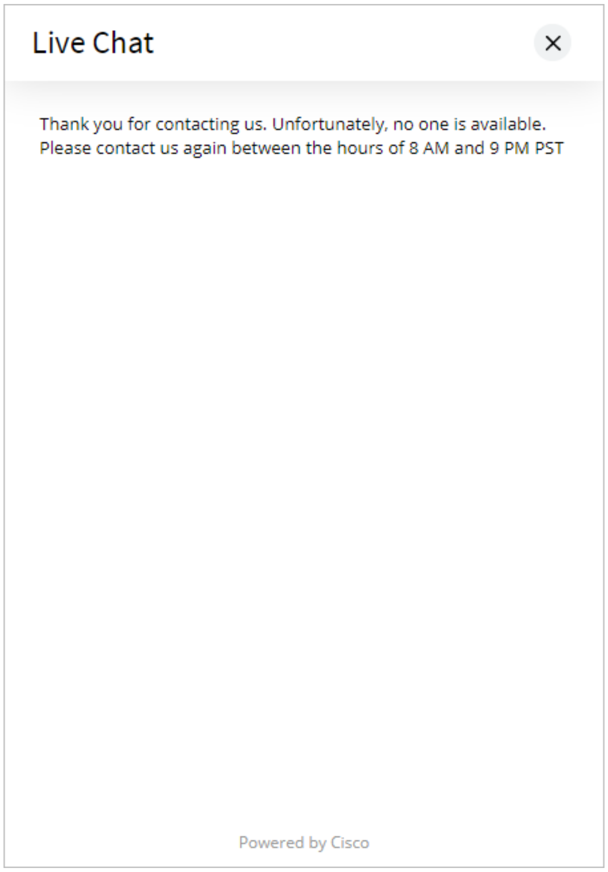
> If the installation is integrated with Cisco WXM, this post chat survey is not displayed when the chat ends and is instead replaced with the WXM survey form. For more information about integrating with WXM, see About Certificate Management.

Customer is presented a survey form at the end of the chat.




## Off Hours Message

An off-hours page is displayed to customers, typically when the customer tries to chat during business off hours.




## Do I Need to Create New Templates or Just Use the Default One?

The application comes with base templates, which can be used as the basis for the chat templates. A new one can be created by copying a base template to be customized to look like the company website. The chat templates can be customized by overriding various parts of the base templates. This allows for the adjusting of the look and style of the chat template as needed.

> The system supplied templates should not be deleted.

A single template set can be used for all the chat entry points to provide a consistent look and feel. Alternatively, using different templates allows the look and feel of each one to be targeted towards a specific audience.

An entry point can be mapped to multiple template sets and multiple entry points can use the same template set. The template sets are shipped with the application and can readily be mapped to any number of entry points.

## How Easy are the Templates to Modify?

> The template sets are managed from the web server and not from the Administration Console.

Templates are easy to modify. All the properties of the templates are modified using the files available on the web server. It is possible to make a number of layout and wording changes as well as branding changes to match them with the look and feel of the website.

It's important to have familiarity and knowledge of the HTML and CSS languages to make changes in the HTML and CSS files is required.

Templates have configuration at various levels:

**Global** configuration is used to control behavior at the global level, for example, positioning of the chat button or window on the screen, configuration of login parameters or post-chat survey, and so on.

**Component-level** configuration lists properties that can modify the behavior of components, or how data is displayed in the component, such as the number of options to display in a list on a common page shown in the chat window.

For more information about creating and modifying custom template sets, see:

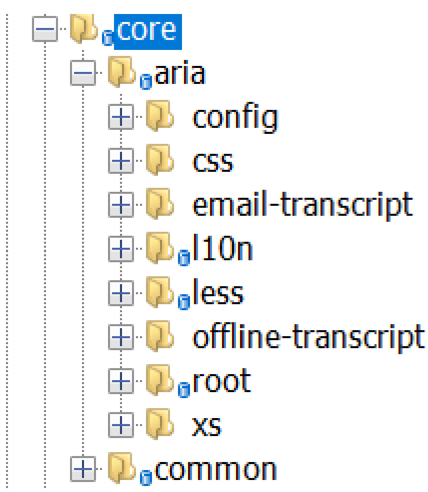
- Creating Template Sets
- Modifying Custom Template Sets

## File Organization

Templates are located on the web server, in the folder *Cisco_Home*`/eService/templates/chat`. By default, this root folder contains the out-of-the-box template, Aria, and the Core folder.

> This folder also contains the Aqua out-of-the-box template. however, it is an older generation of chat template and information regarding customizing Aria chat templates doesn't apply to it. Aqua templates are only used in systems upgrading to 12.6(1) from previous versions, as it is not supported in new installations. For more information about Aqua templates, see About Aqua Chat Template Sets.

The Core folder bundles the default implementation of templates shipped with the product and creates a physical separation between product implementation and custom code in chat templates. Its sub-folders contain the base template (named Aria) and the common resources in the `common` folder. The base template folders have resources that are specific to the Aria template. These include config json, CSS, and l10n files that drive the look-and-feel of the Aria template. It also has the `template.json` file that describes the default definition of that template.




In the `root` folder, at the same level as `core`, is the out-of-the-box Aria template. Its folder has very few files as all the implementation is bundled in the base template in the `core` folder. The file `template.json` in the out-of-the-box Aria template contains the reference to the base template, thus linking it to its default implementation.

The `core` folder is overwritten during a product update. Therefore, any changes must not be made to the contents of the `core` folder.

# Creating Template Sets

Creating a new template set requires copying a template that is pre-configured in the application. Access to the web server is required to perform this task. Custom chat templates are stored at the same level in the file system as out-of-the-box chat templates.

> Anytime you update or create new templates, you must copy the templates files on all web servers in your installation.

Create a separate template set for the following types of activities using the **Aria** template:

- Docked chat activities
- Undocked chat activities

### To create a new template set:

1. On the Web Server, browse to ***eGain_Home***`\eService\templates\chat.`
2. Create a copy of the **Aria** folder and rename the folder.
3. The new template set is created. Custom changes can now be applied to files in the copied folder.

> Never modify any files in the ***Cisco_Home***`\eService\templates\chat\core` folder.

# Modifying Custom Template Sets

Before making any changes to a template set, make sure that the template has been copied from an existing template. For more information, see Creating Template Sets.

Once a custom template has been created, changes can be applied to the `template.json` file in the copied folder. Custom changes always go into the `template.json` file in the custom chat template. Changes to global configurations like positioning of chat button and window are made directly in the custom section in custom chat template's `template.json`.

If the customization requires adding custom implementation or configuration for template entities (such as l10n, CSS, component configuration, and so on), the custom implementation and configuration files should be stored in the chat template folder itself.

> Custom code should never be stored at any other location.

The references of these custom implementation and configuration files are then added in a single `custom` section in the custom chat template's `template.json` for them to be effective. During chat template initialization, when the two `template.json` files from custom chat template and base template are merged, the custom definition from the `template.json` file in the custom chat template overrides the default definition from base template. Hence, it is very important that syntax and structure in the custom section of custom chat template's `template.json` accurately matches that in the `template.json` of the base template.

There are also localization (l10n) files, which are language-specific json files that hold localized text for various labels or messages used in the templates. For more information about customizing these files, see Configuring L10N Files.

## Modifying Global Configurations

### To modify global configurations for a custom template:

1. Ensure the `template.json` in the custom chat template identifies the base template name and version.

   For example, if the base template name is aria and version is 21.0.0 open the *Cisco_Home*`\eService\templates\chat\core\aria\root\21.0.0\template.json` file in a text editor. In the text editor, find the reference to the particular key or property you wish to change. This is specified in json structure under `configs >> allConfigs >> preferences`. For example, if you wanted to enable anonymous chats for the chat template, you would locate the reference to the `loginParameters` section.

2. In the custom chat template's `template.json` file, add the custom section to the json structure at the same level as 'template' section, if it is not already there, and create the property json structure as `custom >> configs >> allConfigs >> preferences`.

   For example, if you wanted to enable anonymous chats for the chat template, you would paste the `loginParameters` section into the custom chat template's `template.json` file, it would look something like this:

```
{
    "template": {
        "base": {
            "version": "21.0.0",
            "name": "aria",
            "description": "Aria Chat Template"
        }
    },
    "custom": {
        "configs": {
            "allConfigs": {
                "preferences": {
                    "loginParameters": []
                }
            }
        }
    }
}
```

3. Make your modifications to the section in the custom `template.json` file and save your changes.

# Modifying CSS

Before making any changes to the CSS, find out the default styles that are being applied to the custom chat template:

- Ensure the `template.json` in the custom chat template identifies the base template name and version. For example, if the template name is aria and version is 21.0.0. go to `core/aria/root/21.0.0/template.json` and find the reference to default CSS for the template.
- This would be specified in the json structure as: `template > css > allCss > default > cssPath`.

After identifying the various places in the template in which CSS changes are required, the relevant parts of the default CSS can be copied and modified.

## To modify CSS for a custom template:

1. Create a customized CSS file specifying only those classes that you intend to override. All custom styling changes should be bundled into one CSS file.
2. Save the custom CSS file in the custom chat template folder under a folder named `custom`.
3. Add a `custom` section in custom chat template's `template.json` and specify the path to custom CSS file using the custom keyword. For example:

```
{
    "template": {
        "base": {
            "version": "21.0.0",
            "name": "aria",
            "description": "Aria Chat Template"
        }
    },
    "custom": {
        "css": {
            "allCss": {
                "custom": {
                    "cssPath": "custom/mystyles.css"
                }
            }
        }
    }
}
```

4. Save the changes to the `template.json` file

## Modifying Images and Icons

Custom folders can be specified to load the images and icons that are used in custom templates. All default images and icons, along with the new images and icons, should be added in a subfolder in the custom chat template folder. The reference to this subfolder must be be provided in chat template's `template.json` using the `custom` keyword. For example, if a sub-folder was created for images in the in the custom chat folder it might look like:

```
{
    "template": {
        "base": {
            "version": "21.0.0",
            "name": "aria",
            "description": "Aria Chat Template"
        }
    },
    "custom": {
        "media": {
            "allMedia": {
                "custom": {
                    "folderPath": "custom/mymedia"
                }
            }
        }
    }
}
```

# Deleting Template Sets

Repeat these tasks on all web servers in your installation.

> The system supplied templates should not be deleted.

### To delete a template set:

1. On the web server, browse to *Cisco_Home*`\eService\templates\chat`.
2. Delete the template folder that you do not want to use. If you are using these templates for any entry point, make sure to update the HTML code on the web pages with the new template sets you want to use.

# Configuring L10N Files

The reference to localization files (l10n) is defined in base template's `template.json` file. Search for `l10ns` in the file to locate the referenced file and supported languages.

To override the default localization text or to add a new key-value pair, a new l10n file for the languages is required.

### To modify the localization of the custom template:

1. Create a new l10n file for the languages in which the change is required. Only the changed or newly added key-value pairs need to be specified in the custom l10n files. All other data from the default l10n files dos not need to be copied over if it is not to be overridden.
2. Save the new l10n file in the template folder under a folder named `custom`.
3. Add a custom section in chat template's `template.json` if it is not already there and specify the path to custom l10n file using the `custom` keyword.

## Providing Additional Attributes for L10n files

Additional attributes called `languages` and `messaging` are added for specifying which custom `<language>-<locale>.json` and `messaging_<language>-<locale>.json` l10n files should be loaded during template launch. This eliminates the step of adding the changed key-value pairs in all languages, even if the chat template is not accessed in those languages. For example, the custom section in chat template's `template.json` would look like:

```
{
    "template": {
        "base": {
            "version": "21.0.0",
            "name": "aria",
            "description": "Aria Chat Template"
        }
    },
    "custom": {
        "l10ns": {
            "allL10ns": {
                "custom": {
                    "folderPath": "custom/myl10n",
                    "languages": {
                        "en-US": true,
                        "xx-XX": true
                    },
                    "messaging": {
                        "en-US": true,
                        "xx-XX": true
                    }
                }
            }
        }
    }
}
```

# Enabling Anonymous Chats

Anonymous chats allow customers to chat with agents without providing any personal details like name, email address, phone number, and so on. When the customer clicks the chat link or button, he is not displayed the login page.

### To enable anonymous chats for chat entry points:

1. Locate the `loginParameters` key defined in `template.json` of the base template and copy the section into the custom chat template's `template.json` file. For more information about performing this step, see Modifying Custom Template Sets.
2. Delete all the parameters between the square brackets. The property will look like:

```
{
    "template": {
        "base": {
            "version": "21.0.0",
            "name": "aria",
            "description": "Aria Chat Template"
        }
    },
    "custom": {
        "configs": {
            "allConfigs": {
                "preferences": {
                    "loginParameters": []
                }
            }
        }
    }
}
```

3. Save the changes.

# Configuring Login Page Parameters

The web form customers must complete to initiate a chat can be configured. The system uses the customer's contact details (for example, email address, phone number, and so on) to identify existing customers and add the chat session to the customer records automatically. It also creates a customer record for new customers.

The following can be configured:

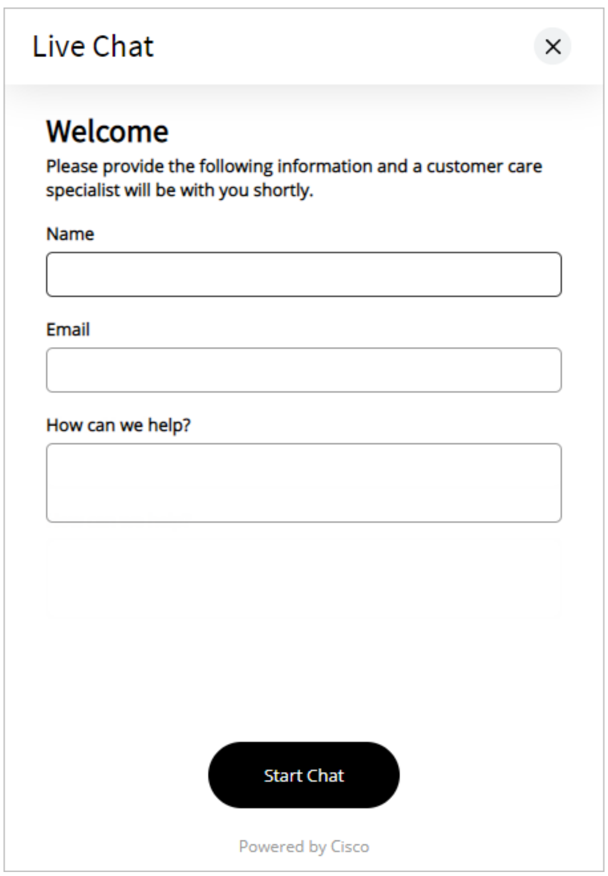
- Change the text of the options
- Remove fields
- Change the order of fields
- Change error messages
- Change the format of the field

### To configure the login page:

1. Locate the `loginParameters` key defined in `template.json` of the base template and copy the section into the custom chat template's `template.json` file. For more information about performing this step, see Modifying Custom Template Sets.
2. To remove an existing field, select the code for the field and delete it from the file. For example, to remove the phone number from the login page, delete the following lines:

```
{
    "eGainAttributeName": "phone_number",
    "eGainParentObject": "casemgmt",
    "eGainChildObject": "phone_number_data",
    "primaryKey": false,
    "controlType": "text",
    "label": "EG_PHONE_NUMBER",
    "required": false,
    "minLength": "0",
    "maxLength": "18",
    "validationPattern": "([0-9 \\-\\(\\)])*$"
},
```

3. To change the required fields, locate the attribute which you want to change and do the following:
   - For the attribute, locate the `required` property and change the value to **true** to make it a required field, or change the value to **false** to make it an optional field.
   - If you are setting a field as required, you must set the minimum length allowed for the field.
   - In the attribute section, locate the property **minLength** and set a minimum length for the field. You can also set the maximum length by configuring the **maxLength** setting.




4. By default, the email address is set as the primary key for the login page. Also, note that only one field can be set as the primary key. The attribute that is set as the primary key needs to be marked as a required field.

5. You can change the text of the fields displayed on the login page. Locate the following properties and provide new labels for the fields in customized `language-locale.json` file.
   Please refer to Configuring L10N Files for more details.
   - **EG_FULLNAME:** Change the label of the name field.
   - **EG_EMAIL_ADDRESS:** Change the label of the email field.
   - **EG_PHONE_NUMBER:** Change the label of the phone number field.
   - **EG_QUESTION:** Change the label of the text field.
6. Save the changes.

## Adding New Fields to the Login Page

Along with full name, email address, and phone number, custom attributes can be used on the login page as well. Custom attributes created for the following objects can be added to the login screen.

- Customer Data
- Contact Point Data
- Activity Data

The following table lists the fields that can be used on the login page. It also lists the attribute names and object names for each field. You will need this information while adding new fields to the login screen.

| For | Parent Object | Child Object | Attribute Name |
|---|---|---|---|
| Full Name | casemgmt | individual_customer_data | full_name |
| Email Address | casemgmt | email_address_contact_point_data | email_address |
| Phone Number | casemgmt | phone_number_data | phone_number |
| Subject Custom attributes added to activity data | casemgmt | activity_data | subject<br>If the custom attributes added to the object has two words, replace the space with underscore (_). For example, if the attribute created in the application is `More details`, the attribute name will be `more_details`. |

| For | Parent Object | Child Object | Attribute Name |
|---|---|---|---|
| Custom attributes added to customer data | casemgmt | customer_data | If the custom attributes added to the object has two words, replace the space with underscore (_). For example, if the attribute created in the application is `Family name`, the attribute name will be `family_name`. |
| Custom attributes added to contact point data | casemgmt | contact_point_data | If the custom attributes added to the object has two words, replace the space with underscore (_). For example, if the attribute created in the application is `Mobile number`, the attribute name will be `mobile_number.` |

### To add new fields:

1. Open the custom chat template's `template.json` file in a text editor and add the reference to `loginParameters`, if not already there. This is specified in json structure as `configs >> allConfigs >> preferences`. For more information about performing this step, see Modifying Custom Template Sets.
2. Locate the `loginParameters` property and copy a section for one of the existing parameters and paste it where you want to add the custom field.
3. Edit the following properties.
    - `maxLength`: Set the maximum allowed characters for the field. The value should not be set more than the value set in the application while creating the custom attribute.
    - `eGainAttributeName`: Provide the attribute name. See the table for details about the attribute name you need to provide.
    - `eGainParentObject`: Provide the parent object name. See the table for details about the object name you need to provide.
    - `eGainChildObject`: Provide the child object name. See the table for details about the object name you need to provide.
    - `primaryKey`: Set the value as **false.**
    - `controlType`: Set the format for the field. For a single line text field, set the value to **text**. For a multi-line text field, set the value to **textarea**. For a single-select dropdown, set the value to **singleselect**. For a multi-select option, set the value to **multiselect**.
    - `label`: Provide the label from the l10n language file for the field. For example, for the phone number field, use `APP.EG_PHONE_NUMBER`.

- `required`: If you want to set the field, as optional set the value as **false.** To make the field mandatory, set the value as **true.**
- `minLength`: Set the minimum required characters for the field.
- `validationPattern`: Provide the expression against which the information provided by the customer is to be checked. For example, for phone number you can give an expression like, `^\s*\(?\s*\d{3}\s*\)?\s*[-.]?\s*\d{3}\s*[-.]?\s*\d{4}\s*$`. And, for email you can use the expression like, `^[0-9a-zA-Z\.\-\_]+\@[0-9a-zA-Z\.\-]+$` - this expression refers to the format X@Y. Z (john@mycompany.com). For fields that require an integer value, you need to at least set the validation string as `\\d+$`.

4. Save the changes.

# Enabling Auto-Login

When the chat link is provided in an authenticated section of the website, the auto-login feature can be used to automatically login the customer for chat. When the customer clicks the chat link or button, he is not displayed the login page.

> A template where auto-login is enabled should not be used for regular chats.

### To enable auto-login for chats:

1. Locate the `enableAutoLogin` key defined in `template.json` of the base template and copy the section into the custom chat template's `template.json` file. For more information about performing this step, see Modifying Custom Template Sets.
2. Set the value to **`true`** to enable auto-login. For example, the custom chat template's `template.json` file would look like:

```
{
    "template": {
        "base": {
            "version": "21.0.0",
            "name": "aria",
            "description": "Aria Chat Template"
        }
    },
    "custom": {
        "configs": {
            "allConfigs": {
                "preferences": {
                    "enableAutologin": true
                }
            }
        }
    }
}
```

3. Save the changes.

# Enabling Customer Single Sign-On for Chat Templates

Chat entry points can be configured to transfer customer context information from the company website to the application through SAML. For more information about Secure Chat for customers, see About Customer Single Sign-On.

## To configure chat templates for customer single sign-on:

1. Locate the `EnableAutologin` key defined in `template.json` of the base template and copy the section into the custom chat template's `template.json` file. For more information about performing this step, see Modifying Custom Template Sets.
2. Enable auto-login by setting the value of the `EnableAutologin` parameter to **true**.
3. Configure the attributes that will be transferred to the application along with the chat creation request. You can do this by configuring the `loginParameters` property for each of the attributes as described in Configuring Login Page Parameters.
4. If any of the attributes will be transferred to the application in the SAML assertion, specify the following additional properties for the attribute in `loginParameters`:
    - `providerAttributeName`: Name of the attribute in the SAML assertion

- `secureAttribute`: You have multiple options of what to set here. Set the value to **1** or **SECURE** to indicate that this attribute will be transferred in the SAML assertion. For example, if the below configuration is done for full_name, then this field is not shown on login page and application searches for and extracts information for this attribute from SAML assertion:

```
{
"providerAttributeName”: ”my_name”,
"secureAttribute": "1",
"eGainAttributeName": "full_name",
"eGainParentObject": "casemgmt",
"eGainChildObject": "individual_customer_data",
"primaryKey": false,
"controlType": "text",
"label": "APP.EG_FULLNAME",
"required": true,
"minLength": "1",
"maxLength": "120",
"validationPattern": ""
}
```

- Set the value to **2** or **UNAUTH_AND_SECURE** to indicate that this attribute will be transferred in the SAML assertion and it will also be displayed on login page for authentication at start. For example, If the below configuration is done for full_name, then this field is shown on login page and application also searches for and extracts information for this attribute from SAML assertion.

```
{
"providerAttributeName”: ”my_name”,
"secureAttribute": “2”,
"eGainAttributeName": "full_name",
"eGainParentObject": "casemgmt",
"eGainChildObject": "individual_customer_data",
"primaryKey": false,
"controlType": "text",
"label": "APP.EG_FULLNAME",
"required": true,
"minLength": "1",
```

```
"maxLength": "120",

"validationPattern": ""
}
```

- If `secureAttribute` is missing or is set to **0** or **UNAUTHENTICATED**, the application does not search for and extract information for that attribute from SAML assertion, and the full_name field appears on the login screen. For this configuration, you must specify the login parameters the application searches for and extracts from SAML assertion in a similar way as that of a 1 or SECURE attribute.
- If the `EnableAutologin` property value is false and all the login parameters are configured with the `secureAttribute` value as **1** or SECURE, then the chat is treated as an Anonymous chat. If `EnableAutologin` property value is true and all the login parameters are configured with the `secureAttribute` value as 1 or **SECURE** along with the SAMLToken, then the chat is treated as a secure chat.

# Changing Chat Button and Window Position

There may be a requirement to change the position of chat button and chat window in a template.

There are following supported positions to display chat window and chat button.

- right-bottom
- right-center
- left-bottom
- left-center
- center-bottom

### To change the position of the chat window:

1. Locate the `maximizedChatWindowPosition` key defined in `template.json` of the base template and copy the section into the custom chat template's `template.json` file. For more information about performing this step, see Modifying Custom Template Sets.
2. Provide one of the supported positions in the parameter field. For example, if the position is `left-bottom`, the custom chat template's `template.json` file would look like:

```
{
    "template": {
        "base": {
            "version": "21.0.0",
            "name": "aria",
            "description": "Aria Chat Template"
        }
    },
    "custom": {
        "configs": {
            "allConfigs": {
                "preferences": {
                    "maximizedChatWindowPosition": "left-bottom"
                }
            }
        }
    }
}
```

3. Save the changes.

## To change the position of the chat button:

1. Locate the `chatButtonPosition` key defined in `template.json` of the base template and copy the section into the custom chat template's `template.json` file. For more information about performing this step, see Modifying Custom Template Sets.
2. Provide one of the supported positions in the parameter field. For example, if the position is `left-bottom`, the custom chat template's `template.json` file would then look like:

```
"custom": {
"configs": {
"allConfigs": {
"preferences": {
"chatButtonPosition": "left-bottom"
}
}
}
}
```

3. Save the changes.

# Changing the Size of the Chat Frame

For more information about how to change the placement of the chat window, see Changing Chat Button and Window Position.

## To change the size of the chat frame:

1. Open the *Cisco_Home*`\eService\templates\chat\`***custom_chat_template_name***`\css\iframe-style.css` in a text editor. For more information about editing custom CSS files, see Modifying Custom Template Sets.
2. Edit the values for the following property: `#egain-chat-wrapper.egain-maximized`. This property represents the maximized chat window. Set height and width for the maximized chat frame here.
3. Save the changes

# Configuring Chat Input Text Area

In the Chat transcript view, the message input text area is loaded at the bottom of the page.

## To configure the chat input text area:

1. Ensure the template.json in the custom chat template identifies the base template name and version. For example, if the base template name is aria and version is 21.0.0 open the *Cisco_Home*`\eService\templates\chat\core\aria\root\21.0.0\template.json`.
2. Find the reference to the default component configuration in the file in the json structure: `template > config > allConfigs > defaultComponentConfig > configPath`.
3. Open the default component configuration json file as found above from the *Cisco_Home*`\eService\templates\chat\core` folder in a text editor.
4. Find the `message-input` section and the settings within it for which the configuration needs to be overridden and create a custom `component-config.json` file containing only the overridden settings. It is not required to copy the rest of the data from the file.

   For example, if you want to override the `maxCharacters` setting of the `message-input` component to `800`, then the contents of the custom component config file would look like:

   ```
   "message-input": {
   "Settings": {
   "maxCharacters": "800"
   }
   }
   ```
5. The following settings can be modified:
   - **`showRemainingCharactersBelow:`** This property determines at which point the the Remaining Characters message should be displayed. For example, If `showRemainingCharactersBelow=10` and

maxCharacters=100, if the user has typed 90 characters already, then the Remaining Character message is displayed.

- **showRemainingCharactersDuration**: This property determines the number of milliseconds the remaining characters alert stays on screen.
- **numberOfMessageTypingRows**: This property determines the number of rows for the input text area.
- **maxCharacters**: This property determines the maximum number of characters that can be typed in the input text area.
- **customerNoTypingTimeout**: This property determines the number of milliseconds after which the customer stops typing that the "Customer is Typing" message is sent.
- **sendDroppedAttachmentsDirectly**: If this property is true, dropped attachments are sent directly without showing the list of attachments in attachment list.
- **textChoicesStretch**: This property determines whether the choice buttons should be stretched to full-width or not.
- **textChoicesDisplay**: This property determines whether the choice elements should be displayed with horizontal scroll buttons or to wrap material to the next line.
- **textChoicesScrollLogic**: This property determines how many choices should be scrolled when clicked once on left or right scroll button. This only applies when the textChoicesDisplay property is set to scroll.
- **imageChoicesStretch**: This property determines whether the image choice elements should be stretched to full-width or not.
- **imageChoicesDisplay**: This property determines whether the image choices should be displayed with horizontal scroll buttons or to wrap material to the next line.
- **imageChoicesScrollLogic**: This property determines how many image choices should be scrolled when clicked once on left or right scroll button.
- **showTextWithImageChoices**: This property determines whether text should be displayed with image choices or not.
- **flexibleHeightTextarea**: This property determines whether the text area should have a fixed height or flexible height.

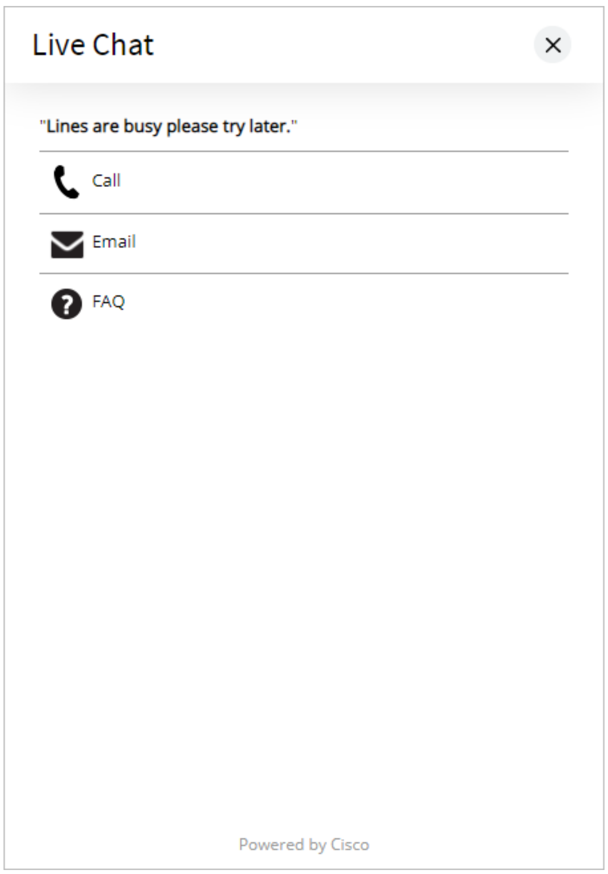
6. Save the changes in the custom component-config.json file.
7. Save the custom component-config.json file in a folder named custom in the custom chat templates folder.
8. Add the path to the custom component-config.json file in the custom chat template's template.json in a custom section, for example:

```
"custom": {
"configs": {
"allConfigs": {
"customComponentConfig": {
```

```
"configPath" : "custom/my-component-config.json"
}
}
}
}
```

# Configuring Alternate Engagement Options

When a chat session is initiated and opens to the transcript view, the user is presented with other options if an agent is not immediately available. These options are shown as buttons.




Templates can be reconfigured for alternate engagement options to allow:

- Changing the text of options
- Changing the order of options
- Changing the URL which is opened when an option is clicked
- Adding and removal of options

### To configure the alternate engagement options:

1. Locate the `alternateContactOptions` key defined in `template.json` of the base template and copy the section into the custom chat template's `template.json` file. For more information about performing this step, see Modifying Custom Template Sets.
2. To remove an existing field, select the code for the field and delete it from the file. For example, to remove the phone option from the alternate engagement component, delete the following lines:

```
"phone": {
    "label": "EG_ALT_CONTACT_PHONE",
    "image": "icon_phone_call_1",
    "url": "http://customer.egain.net
},
```

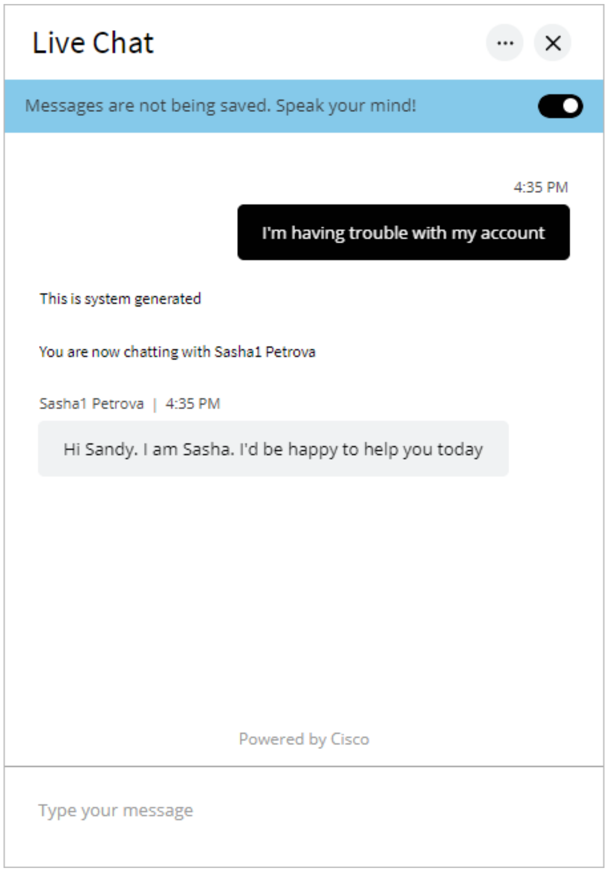
3. You can change the text of the engagement option. To change it, locate the `label` property in the customized `language-locale.json` file and provide new labels for the fields. For more details, see Configuring L10N Files.
4. The URL in the engagement option can be any URL that should be opened after a user clicks the button. This can be an external URL, an entry point, and so on. There are no specific requirements for this.
5. Save the changes.

## Configuring the Off-the-Record Text

When data masking is enabled, masking rules prevent agents and customers from sending sensitive information like credit card numbers, social security numbers, and so on, during the chat session. The Off-the-record feature allows agents and customers to exchange sensitive information as data masking rules do not apply to these messages. Any information exchanged while off-record is not stored in the system. Off-the-record conversations can be started and stopped only by customers. This feature is enabled while configuring data masking for the chat channel. For more information about data masking, see About Data Masking.

### To change the text of off-the-record messages:

1. Open the ***Cisco_Home***`\eService\templates\chat\`***custom_chat_template_name********`l10n\language-locale.json`* and modify the `EG_OFF_RECORD` and `EG_ON_RECORD` labels in customized `language-locale.json` file. For more information, see Configuring L10N Files.




2. Save the changes.

# Changing Color of Buttons

Changes outlined here apply the colors for **Start Chat**, **Send**, and **Close** buttons.

## To change button colors:

1. Open the *Cisco_Home*`\eService\templates\chat\`*custom_chat_template_name*`\css\iframe-style.css` in a text editor. For more information about editing custom CSS files, see Modifying Custom Template Sets.
2. Override the CSS for following classes to apply customized styles to various buttons:
   - **`#eg-chat-pre-chat-submit .eg-button-1`**: Override styles of this class to change **Start Chat** button style.
   - **`#eg-chat-header .eg-header .eg-header-btn-wrapper .eg-header-btn`**: Override styles of this class to change the **Close** or other header button styles.
3. Save the changes.

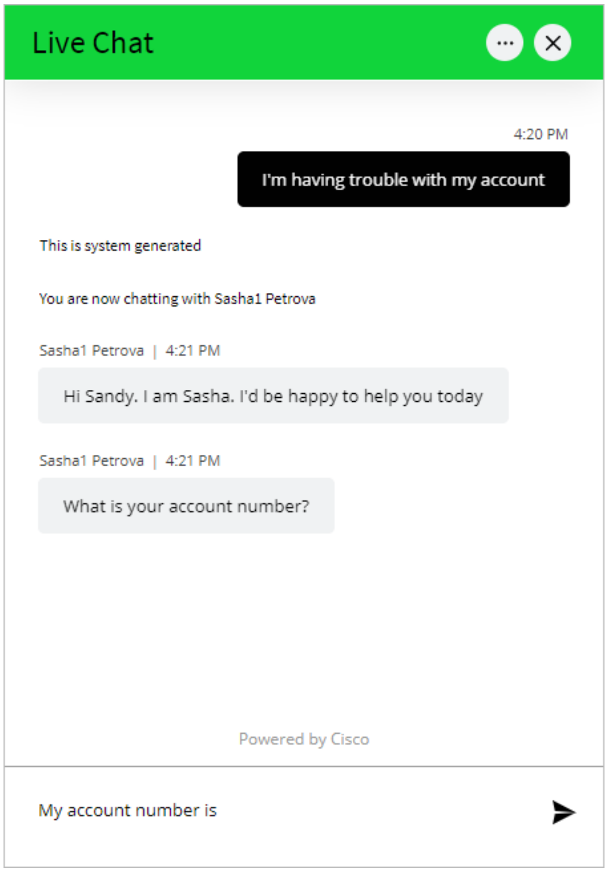
4. Adjusting the **Send** button requires changes be made to the images or icons that are used, as well as the references in the custom `template.json` files. To modify the **Send** button:
   - Create the image folder in your custom chat template folder and add your images there. For more information, see Modifying Custom Template Sets.
   - After adding custom image files to the media folder, go toOpen the *Cisco_Home*`\eService\templates\chat\`*custom_chat_template_name*`\template.json` file in a text editor and locate the `custom >> media >> allMedia >> fileMap section`. Create `custom>>media>>allMedia>>fileMap` section if not already created .
   - Locate the `icon_send` property in the `template.json` file. Search for the section: `media >> allMedia >> fileMap`. In this example, the value in `icon_phone_call_1`, is name of the file which is currently in use. Change the `icon_send` property to the name of the image file. For example, instead of `"icon_send": "icon-send.svg"`, it would be `"icon_send": "new_image.svg"`.
5. Save the changes.

# Changing Header and Footer Colors

Changes outlined here the headers and footers of all pages in the Chat Customer Console: the Login page, the Chat page, and the Survey page.

## To change the header and footer colors:

1. Open the *Cisco_Home*`\eService\templates\chat\`*custom_chat_template_name*`\css\iframe-style.css` in a text editor. For more information about editing custom CSS files, see Modifying Custom Template Sets.
2. Override the CSS for following classes to apply customized styles to the header:
   - **`#eg-chat-header`**: Override styles of this class to apply a customized style to the header. To change the `background-color` the `background-image` property must be set to `none`.

3. 


4. Override the CSS for following classes to apply customized styles to the footer:
   - **`#eg-chat-footer`**: Override styles of this class to apply a customized style to the footer.
   - **`#eg-branding`**: Override styles of this class to apply a customized style to the branding section of the footer that has the 'Powered by' message.
5. Save the changes.

# Configuring Surveys

At the end of the chat session a survey form is presented to customers to measure the usefulness of the chat. The survey form has survey questions and a text box for providing additional comments. There are different types of surveys that can be configured. The default survey overwrites the chat transcript view, interactive surveys display questions in the chat transcript view itself. External surveys open an external URL. The text of these questions can be changed, however, questions cannot be added or removed from the survey form.

If a survey is not necessary, it can be turned off as well. For details, see Turning Off Surveys.

> If the installation is integrated with Cisco WXM, this post chat survey is not displayed when the chat ends and is instead replaced with the WXM survey form. Therefore, this setting does not apply to WXM survey forms if the installation is integrated with Cisco WXM. For more information about integrating with Cisco WXM, see About Certificate Management.

### To change the survey type:

1. Locate the `surveyType` key defined in `template.json` of the base template and copy the section into the custom chat template's `template.json` file. For more information about performing this step, see Modifying Custom Template Sets.
2. The survey type can be changed by configuring the `surveyType` value as `default`, `interactive`, and `external`. If you wish to use external survey then, the `externalSurveyURL` field and value need to be added to the custom chat template's `template.json` file as well. For example, the custom `interactive` chat template's `template.json` file might look like:

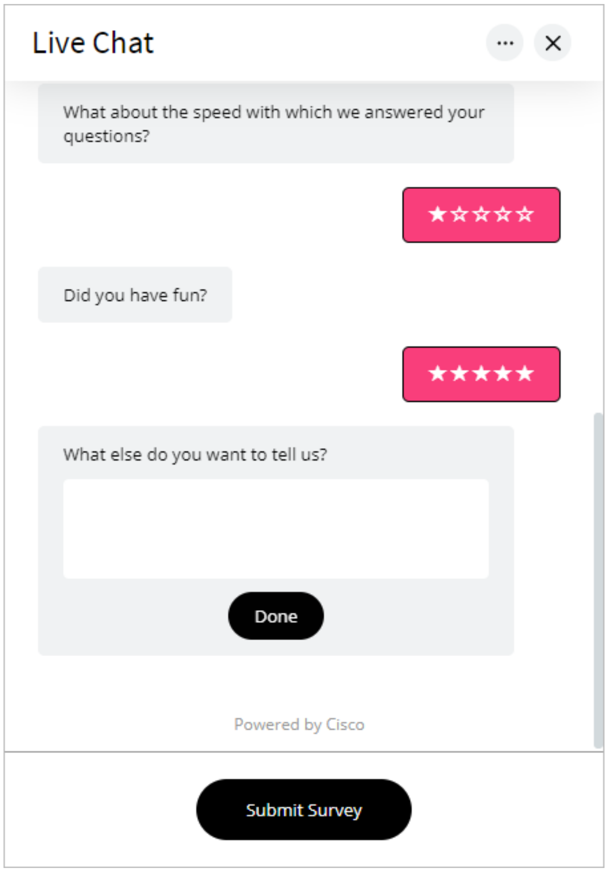
```
"custom": {
"configs": {
"allConfigs": {
"preferences": {
"surveyType": "interactive",
"externalSurveyURL": "",
"surveyQuestions": [
{
"questionText": "EG_SURVEY_QUESTION_1",
"controlType": "STAR",
"minRating": 1,
"maxRating": 5,
"ratingTooltip": [
"EG_POOR",
"EG_FAIR",
"EG_GOOD",
"EG_VERY_GOOD",
"EG_EXCELLENT"
]
},
{
"questionText": "EG_SURVEY_QUESTION_2",
"controlType": "STAR",
"minRating": 1,
"maxRating": 5,
"ratingTooltip": [
"EG_POOR",
"EG_FAIR",
"EG_GOOD",
"EG_VERY_GOOD",
"EG_EXCELLENT"
]
},
{
"questionText": "EG_SURVEY_QUESTION_3",
```

```
"controlType": "STAR",
"minRating": 1,
"maxRating": 5,
"ratingTooltip": [
"EG_POOR",
"EG_FAIR",
"EG_GOOD",
"EG_VERY_GOOD",
"EG_EXCELLENT"
]
},
{
"questionText": "EG_SURVEY_QUESTION_4",
"controlType": "TEXTAREA"
}
]
}
```

3. Save the changes

### To change the text of survey questions:

1. Locate the `surveyQuestions` key defined in `template.json` of the base template and copy the section into the custom chat template's `template.json` file. For more information about performing this step, see Modifying Custom Template Sets.
2. Locate the attribute which you want to change and provide the appropriate value to the field. For example if you wished to change the `controlType` property, you need to locate the property and change the value to `STAR`, `TEXTAREA`, `SINGLESELECT`, or `MULTISELECT` based on the requirement.
3. To change the text of the survey questions, locate the properties EG_SURVEY_QUESTION_1, EG_SURVEY_QUESTION_2, EG_SURVEY_QUESTION_3 and EG_SURVEY_QUESTION_4 and provide new labels for the fields in customized `language-locale.json` file. Please refer to Configuring L10N Files for more details.




4. Save the changes.

# Turning Off Surveys

If a survey is not necessary, it can be turned off. For more information on configuring survey types and text, see Configuring Surveys.

> If the installation is integrated with Cisco WXM, this post chat survey is not displayed when the chat ends and is instead replaced with the WXM survey form. Therefore, this setting does not apply to WXM survey forms if the installation is integrated with Cisco WXM. For more information about integrating with Cisco WXM, see About Certificate Management.

### To turn off surveys:

1. Locate the `surveyType` key defined in `template.json` of the base template and copy the section into the custom chat template's `template.json` file. For more information about performing this step, see Modifying Custom Template Sets.
2. In the `surveyType` field, provide the value `none`. For example, the custom chat template's `template.json` file would look like:

```
{
    "template": {
        "base": {
            "version": "21.0.0",
            "name": "aria",
            "description": "Aria Chat Template"
        }
    },
    "custom": {
        "configs": {
            "allConfigs": {
                "preferences": {
                    "surveyType": "none"
                }
            }
        }
    }
}
```

3. Save the changes.

# Configuring Chat Templates for WXM Surveys

If the installation is integrated with Cisco WXM, the WXM survey form is displayed at the end of the chat session to allow the customer to provide feedback.

For more information about integrating with Cisco WXM, see About Certificate Management.

## To configure chat templates for WXM surveys:

1. Locate the `surveyType` key defined in `template.json` of the base template and copy the section into the custom chat template's `template.json` file. For more information about performing this step, see Modifying Custom Template Sets.

2. Change the `surveyType` value to `external`. For example, the custom section in chat template’s `template.json` would look like:

```
"custom": {
"configs": {
"allConfigs": {
"preferences": {
"surveyType": "external",
}
}
}
```

```
}
```

3. Save the changes. Once this change has been saved, the application can automatically redirect users to the WXM survey form.

# Changing General Chat Template Settings

Changing general settings for the chat template requires modifying the l10n files. For more information, see Configuring L10N Files.

## To change the general properties for templates:

1. Open the ***Cisco_Home***`\eService\templates\chat\`***custom_chat_template_name***`\`*`l10n\language-locale.json`* and modify the following key value pairs:

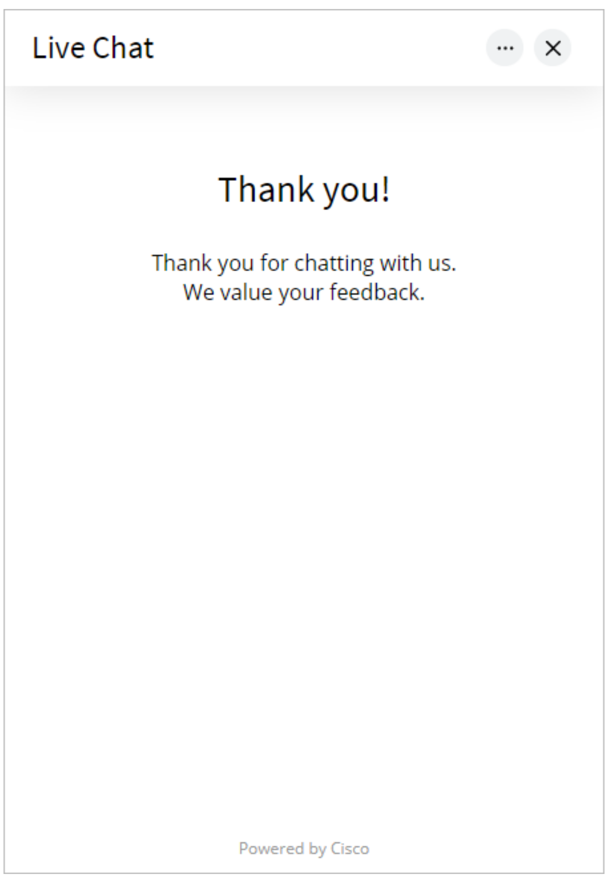
   `TITLE`: Change the title of the chat window. The default value is Live Collaboration by Cisco.

   `EG_CONFIRM_CLOSE_CHAT`: Change the message displayed to the agent when the customer closes the chat session.

   `EG_START_CHAT`: Change the text of the button for starting chat. The default value is Start Chat.

   `EG_SEND`: Change the text of the button for sending the first chat message. The default value is Send.

   `EG_SUBMIT`: Change the text of the button for sending chat messages. The default value is Submit.



2. Save the changes

# Changing Chat Transcript Options

During a chat session and at the end of a chat session, customers have access to the **Save Transcript** button. Customers can use this to save the chat transcript for future reference. The tool-tips can be modified. The header, footer, and title for the transcript window can also be modified. These changes are done in the l10n files. For more information, see Configuring L10N Files.

## To change the chat transcript options:

1. Open the ***Cisco_Home***`\eService\templates\chat\`***custom_chat_template_name*******`l10n\language-locale.json` and modify the following key value pairs:
   - `EG_TOOLBAR_SAVE_TRANSCRIPT`: Change the tool-tip of the **Save Transcript** button.
   - `EG_TRANSCRIPT_HEADER`: Provide the header for the transcript page.
   - `EG_TRANSCRIPT_START_TIME`: Change how you want the start time of the transcript to be displayed.

     Do not change the string `{{chatStartTime}}`.

   - `EG_SAVED_TRANSCRIPT_HEADING`: Change the heading of the saved transcript.
   - `EG_TRANSCRIPT_FOOTER`: Change the footer of the saved transcript.
   - `EG_POWERED_BY_BRAND`: Change the label for the brand that appears in the "Powered by" section of the footer.
   - `EG_PRINT`: Change the **Print** button title in saved transcript.
2. Save the changes.

# Configuring Sub-Domain Navigation Options

The chat template, by default, allows for website visitors to chat with agents while browsing pages hosted on a single domain. To allow for visitors to chat with agents while navigating to other web pages hosted on other subdomains, further configuration is required. The template must be configured to store chat session data in cookies, thus allowing visitors to freely navigate to other sub-domains without ending the chat session.

> It is not recommended to use multiple chat templates on a domain if only one template has been configured for sub-domain navigation.

### To configure templates for sub-domain navigation:

1. Locate the `multiSubDomainSupport` key defined in `template.json` of the base template and copy the section into the custom chat template's `template.json` file. For more information about performing this step, see Modifying Custom Template Sets.
2. Adjust the following properties:
   - `IsRequired`: By default, the values is set to **false**. Set the value to **true** to enable storage of chat session data in cookies.
   - `CookieParentDomain`: Provide the parent domain in which the cookies are stored. For example, `help.com`. This is required if `IsRequired` is set to **true**.
   - `CookiePath`: Provide the common URL path of all the sub-domains that should be supported for continuing the chat, for example: `/chat`. This prevents the cookie data from being sent over to the server with every request. This is required if `IsRequired` is set to **true**.
   - `IsCookieSecured`: Determine whether the cookies should only be used for HTTPS domains. Set to **true** to allow cookies only to be used in HTTPS connections, or set to **false** to allow cookies to be used in HTTP and HTTPS connections.

   For example, the custom chat template's `template.json` file might look like:

```
{
    "template": {
        "base": {
            "version": "21.0.0",
            "name": "aria",
            "description": "Aria Chat Template"
        }
    },
    "custom": {
        "configs": {
            "allConfigs": {
                "preferences": {
                    "multiSubDomainSupport": {
                        "IsRequired": true,
                        "CookieParentDomain": "help.com",
                        "CookiePath": "/",
                        "IsCookieSecured": true
                    }
                }
            }
        }
    }
}
```

3. Save the changes.

# Callback Templates



## About Callback Template Sets

Template sets consist of CSS (cascading style sheets) and JS (JavaScript) files. The CSS files control the look and feel of the Callback Console. The JS files contain the business logic used to render data in the Callback Console. Templates are also used to determine what information is requested to identify the customer (e.g. name, email address, phone number). You can also compose messages that the customer will see under certain circumstances (e.g. if they request a callback session out of hours).

An entry point can be mapped to multiple template sets and multiple entry points can use the same template set. The template set, Cisco Rainbow, is shipped with the application and can readily be mapped to any number of entry points.

> The system supplied templates should not be deleted.

The template sets are managed from the web server and not from the Administration Console.

## What Makes up a Template

A template consists of a set of pages that are designed to work together. Typically, a callback template has three pages - a callback or delayed callback request form, the call status page, and the off-hours page. All these pages can be customized to meet your business needs.

- **Callback Request Form:** Customers fill up this form to start a callback session.
- **Callback Status Area:** Status of call session is displayed here.
- **Off Hours Page:** An off-hours page is displayed to customers, typically when the customer tries to call during business off hours.

## Do I Need to Create New Templates or Just Use the Default One?

The template set, Cisco Rainbow, is shipped with the application. You must create a copy of the default template set and customize it look like your website.

You can choose to use one template set for all your callback entry points for a consistent look and feel. Alternatively, you can use different templates, so that the look and feel of each one is targeted towards a specific audience.

## How Easy are the Templates to Modify?

Templates are easy to modify. All the properties of the templates are modified using the files available on the web server. It is possible to make a number of layout and wording changes as well as branding changes to match them with the look and feel of your website.

Some important things to note:

- You must have a good knowledge of the HTML and CSS languages to make changes in the HTML and CSS files.
- Always create a backup copy of the files before making any changes.

The following table describes the files that can be changed. All the files are available on the web server at *Cisco_Home*`\eService\templates\callback\`*Template_Name*.

| Use | To |
|---|---|
| `\L10n\custom_callback_`*locale*`.properties` | ▪ Update the text displayed on the callback pages.<br><br>Note: All custom messages should be added in the `custom_callback_`*locale*`.properties` file. Do not make any changes to the `callback_`*locale*`.properties` |
| `\eGainLiveConfig.js` | ▪ Change the fields displayed on the login page. |

| Use | To |
| --- | --- |
| | ▪ Enable auto login.<br>▪ Configure the height of the callback window.<br>▪ Configure the maximum length allowed for the customer message. |
| `\css\callback.css` | ▪ Configure the look and feel of the callback pages. |
| `\call.html` | ▪ Change the layout of the callback page. |
| Images folders:<br>▪ *Template_Name*`\images` | ▪ Change the images used.<br>Note: The size of the custom images should match the size of the default images. However, if you are changing the size of the callback windows, you can adjust the images size to match the new windows. |

# Creating Template Set Folders

You need access to the web server to perform this task.

> Anytime you update or create new templates, you must copy the templates files on all web servers in your installation.

Create a separate template set for each of the following types of activities:

- Callback activities
- Delayed callback activities

### To create a new template set:

1. On the Web Server, browse to *Cisco_Home*`\eService\templates\callback`.
2. Create a copy of the **Rainbow** folder and rename the folder.
3. The new template set is created. You can now edit the files in the template folder to configure your template settings and to customize the look of your templates. You can change any files except for files in the following folders:
   - `script`
   - `libs`

After creating the template set, some of the things that you can do are:

- Configure the login page parameters
- Change the off hours message

# Changing General Callback Settings

### To change the general properties for templates:

1. Open the *Cisco_Home*\eService\templates\callback*Template_Folder*\properties\callback_*Locale*.properties file in a text editor.
2. Add the following properties and set the values as required:
   - **L10N_DOCUMENT_TITLE:** Change the title of the callback window. The default value is **ClickToCall by Cisco**.
   - **L10N_BROWSER_CLOSE_MESSAGE:** Change the message displayed to the customer when he closes the callback window.
   - **L10N_CLICK_TO_CALL:** Change the text of the button for initiating callback session. The default value is **Send**.
   - **L10N_CLOSE:** Change the text of the button for closing the callback window. The default value is **Close**.
3. Save the changes.
4. Clear the browser cache and access the callback link to test the changes.

# Changing Header Image

This section talks about changing the header image for the callback templates.

### To change the header image:

1. Change the `logo_click_to_call.png` image file in the *Cisco_Home*\eService\templates\callback*Template_Folder*\image folder with your custom image.
2. Clear the browser cache and access the callback link to test the changes.

# Changing Color of Buttons

This section talks about changing button colors for **Call Me** and **Close** buttons.

### To change button colors:

1. Open the *Cisco_Home*`\eService\templates\callback\`*Template_Folder*`\css\callback.css` file in a text editor.
2. To change the **Call Me** button colors, do the following:
   - To change the button color, locate the `call_button_input_class` class and change the value of the `background-color` property.
   - To change the hover color, locate the `call_button_input_class:hover` class and change the value of the `background-color` property.
3. To change the **Close** button colors, do the following:
   - To change the button color, locate the `close-button-class` class and change the value of the `background-color` property.
   - To change the hover color, locate the `close_button_class:hover` class and change the value of the `background-color` property.
4. Save the changes.
5. Clear the browser cache and access the callback link to test the changes.

## Changing the Size of Callback Windows

This section talks about defining the size of the windows.

### To change the size of the callback windows:

1. Open the *Cisco_Home*`\eService\templates\callback\`*Template_Folder*`\eGainLiveConfig.js` file in a text editor.
2. Edit the values for the following properties:
   - **windowWidth**: Set the width of the callback and delayed callback windows. The default value is 425 pixels.
   - **windowHeightCallback**: Set the height of the callback page. The default value is 600 pixels.
   - **windowHeightDelayedCallback**: Set the height of the delayed callback page. The default value is 680 pixels.
3. Save the changes.
4. Clear the browser cache and access the callback link to test the changes.

# Configuring Call Request Page Parameters

Build the web form which customers must fill in to initiate a web callback interaction. The system uses the customer's contact details (for example, email address, phone number, and so on) to identify existing customers and add the session to their customer records automatically. It also creates customer records for new customers.

Phone number is a required attribute for callback and delayed callback activities. Schedule call is a required attribute for delayed callback activities and should not be removed from templates configured for delayed callback activities.

You can:

- Change the text of the options
- Remove fields
- Change the order of fields
- Change the primary key

### To configure the call request page:

1. Open the *Cisco_Home*`\eService\templates\callback\`*Template_Folder*`\eGainLiveConfig.js` file in a text editor.
2. Locate the `loginParameters` property, and do the following.
   a. To remove an existing field, select the code for the field and delete it from the file. For example, to remove the question field from the page, delete the following lines:

```
{
paramName : 'L10N_YOUR_QUESTION_PROMPT',
objectName : 'casemgmt::activity_data',
attributeName : 'subject',
primaryKey : '0',
required : '0',
minLength : '1',
maxLength : '800',
fieldType : '2',
validationString : '',
errorMessage : '',
className : 'textarea_field'
}
```

> Do not remove a field that is set as the primary key.

b. To change the order of the fields, remove the code for the field you want to move and paste it at the new location. Make sure to copy the complete set of lines for a field and move them to the new location.

c. To change the required fields, locate the attribute which you want to change and do the following:

   a. For the attribute, locate the property `required` and change the value to 1 to make it a required field, or change the value to 0 to make it an optional field. If you are setting a field as required, you must set the minimum length allowed for the field.

   b. In the attribute section, locate the property `minLength` and set a minimum length for the field. You can also set the maximum length by configuring the `maxLength` setting. These options are not available for the **Schedule Call** field.

d. By default, the email address is set as the primary key for the login page. You can choose to change the primary key to phone number. No other field can be set as the primary key. Also, note that only one field can be set as the primary key. To change the primary key, do the following:

> Only the customer phone number or email address can be set as the primary key.

   a. Locate the section for email address by searching for `attributeName : 'email_address'`. In this section, locate the `primaryKey` property and set the value to 0.

   b. Next, locate the section for phone number by searching for `attributeName : 'phone_number'`. In this section, locate the `primaryKey` property and set the value to 1. Also, the attribute that is set as the primary key needs to be marked as a required field.

   c. In the phone number section, locate the property `required` and change the value to 1.

3. Open the *Cisco_Home*`\eService\templates\`*Template_Folder*`\l10n\custom_callback_Locale.properties` file in a text editor.

   a. You can change the error message for the fields displayed on the call request page. Add the following properties and provide new error messages:

      - **L10N_ERROR_NAME:** Change the error message of the name field.
      - **L10N_ERROR_EMAIL:** Change the error message of the email field.
      - **L10N_ERROR_PHONE:** Change the error message of the phone field.
      - **L10N_ERROR_DELAYED_TIME:** Change the error message in case delay time is not valid.
      - **L10N_ERROR_SUBJECT:** Change the error message of the question field.
      - **L10N_ERROR_DELAYED_TIME:** Change the error message in case delay time is not valid.

   b. You can also change the text of the fields displayed on the call request page. Add the following properties and provide new labels for the fields.

      - **L10N_NAME_PROMPT:** Change the label of the name field.
      - **L10N_EMAIL_PROMPT:** Change the label of the email field.
      - **L10N_PHONE_NUMBER_PROMPT:** Change the label of the phone number field.

- **L10N_YOUR_QUESTION_PROMPT:** Change the label of the text field.
- **L10N_CALL_NOW:** Change the label of **Call Now** in the dropdown. This applies to delayed callback only.
- **L10N_CALL_LATER:** Change the label of **Call Later** in the dropdown. This applies to delayed callback only.
- **L10N_SCHEDULE:** Change the label of **Schedule Call** field. This applies to delayed callback only.

4. Save the changes.
5. Clear the browser cache and access the callback link to test the changes.

# Adding New Fields

Along with full name, email address, and phone number, you can use custom attributes on the login page.

Custom attributes created for the following objects can be added to the login screen.

- Customer Data
- Contact Point Data
- Activity Data

The following table lists the fields that can be added to the login page. It also lists the attribute names and object names for each field. You will need this information while adding new fields to the login screen.

| For | Parent Object | Attribute Name |
|---|---|---|
| Full Name | casemgmt::individual_customer_data | full_name |
| Email Address | casemgmt::email_address_contact_point_data | email_address |
| Phone Number | casemgmt::phone_number_data | phone_number |
| Subject, Custom attributes added to activity data | casemgmt::activity_data | subject<br>If the custom attributes added to the object has two words, replace the space with underscore (_). For example, if the attribute created in the application is `More details`, the attribute |

| For | Parent Object | Attribute Name |
|---|---|---|
| | | name will be `more_details.` |
| Custom attributes added to customer data | casemgmt::customer_data | If the custom attributes added to the object has two words, replace the space with underscore (_). For example, if the attribute created in the application is `Family name`, the attribute name will be `family_name`. |
| Custom attributes added to contact point data | casemgmt::contact_point_data | If the custom attributes added to the object has two words, replace the space with underscore (_). For example, if the attribute created in the application is `Mobile number`, the attribute name will be `mobile_number.` |

## To add new fields:

1. Open the *Cisco_Home*`\eService\templates\callback\`*Template_Folder*`\eGainLiveConfig.js` file in a text editor and do the following:

    a. Locate the `loginParameters` property, and copy a section for one of the existing parameters and paste it where you want to add the custom field. Edit the following properties.

    - **paramName:** Create a parameter name using the format `L10N_`*Name*`_PROMPT.`
    - **objectName:** Provide the object name. See the table for details about the object name you need to provide.
    - **attributeName:** Provide the attribute name. See the table for details about the attribute name you need to provide.
    - **primaryKey:** Set the value as **0.**
    - **required:** If you want to set the field, as optional set the value as **0.** To make the field mandatory, set the value as **1.**
    - **minLength:** Set the minimum required characters for the field.
    - **maxLength:** Set the maximum allowed characters for the field. The value should not be set more than the value set in the application while creating the custom attribute.

- **fieldType:** Select the field type
  - For single line text fields, set the value as **1**.
  - For multi-line text fields, set the value as **2**.
  - For dropdown fields, set the value as **3**.
  - For multi-select dropdown list, set the value as **4**.
- **validationString:** Provide the expression against which the information provided by the customer is to be checked. For example, for phone number you can give an expression like, `^\s*\(?\s*\d{3}\s*\)?\s*[-.]?\s*\d{3}\s*[-.]?\s*\d{4}\s*$`. And, for email you can use the expression like, `^[0-9a-zA-Z\.\-\_]+\@[0-9a-zA-Z\.\-]+$` - this expression refers to the format X@Y. Z (john@mycompany.com). For fields that require an integer value, you need to atleast set the validation string as `\\d+$`.
- **className:** Provide the `css` class you want to use for the field. These are defined in the `callback.css` file available in the `css` folder.

b. Save the changes.

2. Open the *Cisco_Home*`\eService\templates\callback\`*Template_Folder*`\L10n\custom_callback_`*Locale*`.properties` file in a text editor and do the following:

   a. Add the following line in the properties file: `L10N_`*Name*`_PROMPT = "`*Display name for the attribute*"

   `Where: L10N_`*Name*`_PROMPT` should match the value set in **paramName** in step 1.

   b. Save the changes.

3. Clear the browser cache and access the callback link to test the changes. The login page should show the new fields.

# Adding New Countries

By default United States and United Kingdom are the two countries available on the callback form. You can add or remove countries from this list.

## Adding New Countries

### To add a new country:

1. Open the *Cisco_Home*`\eService\templates\callback\`*Template_Folder*`\eGainLiveConfig.js` file in a text editor.

2. Locate the `countryCodes` property, and copy a section for one of the existing country and paste it where you want to add the new country. Edit the following properties:
   a. **name:** Create a name like `l10n_`*Name*
   b. **value:** Provide the country code.
3. Open the *Cisco_Home*`\eService\templates\`*Template_Folder*`\l10n\custom_callback_Locale.properties` file in a text editor and do the following:
   a. Add the following line in the property file: `L10N_`*Name*`=` "*Country_Name*" Where: `L10N_`*Name* should match the value set in Step 2.
   b. Save the changes.
4. Clear the browser cache and access the call link to test the changes. The call request page should show the new country.

## Removing Countries

### To remove a country:

1. Open the *Cisco_Home*`\eService\templates\callback\`*Template_Folder*`\eGainLiveConfig.js` file in a text editor.
2. Locate the `countryCodes` property, and select the code for a country and delete it from the file. For example, to remove United States, and delete the following lines:

```
{
name: 'l10n_US',
value: '1'
}
```

3. Clear the browser cache and access the call link to test the changes. The call request page should not show the country removed from the file.

## Changing Country Order

The order of the country codes in the call request form is determined by the order in which the fields are configured in the `eGainLiveConfig.js`. To change the order, just move the section for a country in the file to the new location.

### To change the order of countries:

1. Open the *Cisco_Home*`\eService\templates\callback\`*Template_Folder*`\eGainLiveConfig.js` file in a text editor.
2. Locate the `countryCodes` property, and rearrange the country codes. For example, to move United Kingdom before United States change the order as:

```
countryCodes : [
{
name: "L10N_UK",
value: "44"
},
{
name: "L10N_US",
value: "1"
}
]
```

3. Clear the browser cache and access the call link to test the changes. The call request page should show the new order of country codes.

# Adding Country Codes

Callback templates can be set to automatically enter the country code for the phone number entered by the customer using an entry point. Be aware, if a customer includes their country code in their phone number while the parameter is enabled, the country code is still automatically added to the number, causing a misdial. Disable the country code parameter to allow customers to enter their own country codes.

### To change the country code parameter:

1. Open the *Cisco_Home*`\eService\templates\chat\`*Template_Folder*`\application\application-chat-defaults.js` file in a text editor.
2. Locate the `useCountryCode` parameter and change it to one of the following:
   - 0: The phone number used to place the call does NOT use the country code.
   - 1: The phone number used to place the call uses the country code.
3. Save the changes.
4. Reload the Callback / Delayed Callback URL entry point web site from the browser.

# Enabling Auto-Login for Callback

When the callback link is provided in an authenticated section of the website, the auto-login feature can be used to automatically login the customer for call. When the customer clicks the callback link or button, he is not displayed the login page.

A template where auto-login is enabled should not be used for regular callback.

### To enable auto-login:

1. Open the *Cisco_Home*\eService\templates\callback*Template_Folder*\eGainLiveConfig.js file in a text editor.
2. Locate the autoLogin property and set the value to 1 to enable auto-login.
3. Save the changes.
4. Change the entry point code so that the customer information collected at the time of login on the website is passed on to the ECE application when the customer initiates a callback session.

## Changing the Off Hours Message

An off hours message is displayed to customers when ECE is not able to connect to Unified CCE to place a call.

### To change the off hours message:

1. Open the *Cisco_Home*\eService\templates\callback*Template_Folder*\l10n\custom_callback_Locale.properties file in a text editor.
2. Add the property L10N_OFF_HOURS and change the message to be displayed to customers.
3. Save the changes.
4. Clear the browser cache and access the callback link to test the changes.

## Deleting Template Sets

You must delete the templates files from all web servers in your installation.

### To delete a template set:

1. On the web server, browse to *Cisco_Home*\eService\templates\callback.
2. Delete the template folder that you do not want to use. If you are using these templates for any entry point, make sure to update the HTML code on web pages with the new template sets you want to use.

# Entry Points



## About Entry Points

An entry point defines the starting point from which customers initiate chat or callback interactions. Every chat help link on a website is mapped to an entry point. Each entry point has a queue associated with it and the queue is used to route chats to agents. A default entry point is provided in each department.

> Multiple help links on a website can point to the same entry point.

Chat Activities and Callback Activities use different starting template sets: Aria, and Callback Rainbow. For more information about the template sets and how you can configure them, see About Template Sets.

You must create separate entry points with the Aria template set for each of the following types of activities:

- Chat activities

You must create separate entry points with the Callback Rainbow template set for each of the following types of activities:

- Callback activities
- Delayed callback activities

# Creating Entry Points

Before creating an entry point, create the queues and template sets to be used for the entry points. For details on creating queues, see About Queues.

Chat Activities and Callback Activities use different starting template sets. For details on creating chat templates, see About Template Sets.

### To create an entry point:

1. In the department-level Top Menu, click **Apps.**
2. In the Left menu, click **Chat & Messaging > Entry Points**.
3. Click the **New** button.
4. The workspace refreshes to show the Create Entry Point page. In the General tab, provide the following details.
   - **Entry Point Active**: Click the toggle switch to turn the entry point on or off.

     > For a customer to be able to chat, the entry point mapped to the chat help link should be made active, else the customer would be displayed a no service page.

   - **Entry Point Name:** Type the name of the entry point. This is required information.
   - **Routing Type:** If necessary, set the routing method for the entry point. By default, this is set to Unified CCE.
   - **Subactivity:** Select the types of activities this entry point is routing from the dropdown. Options include: Chat, Callback, Delayed Callback.
   - **Queue:** Select a queue from the dropdown. This is required information. The queues available can be active or inactive. If the entry point is marked as active, then an active queue must be selected for the entry point.

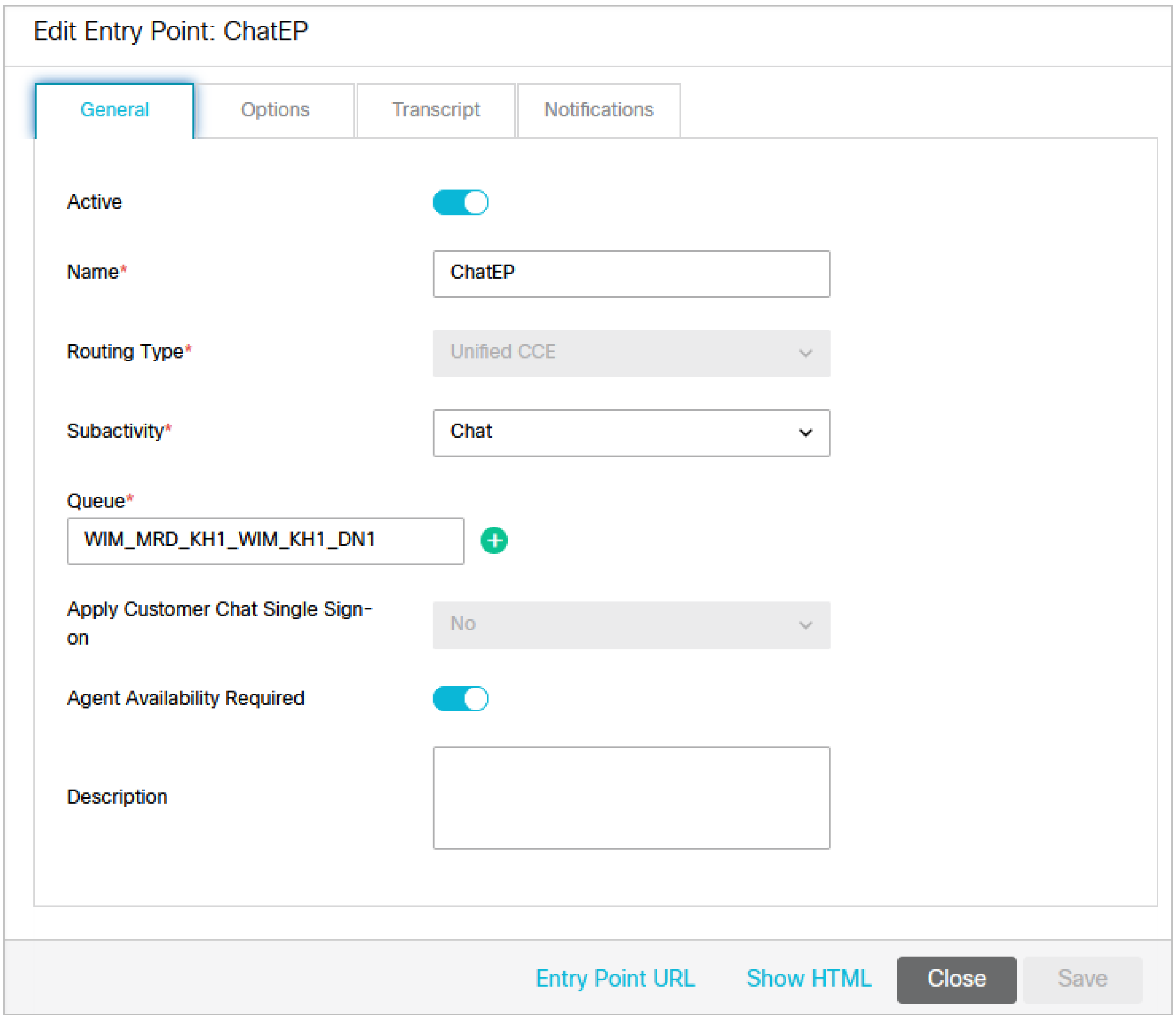
     > An active entry point cannot be mapped to an inactive queue.

   - **Apply customer chat single sign-on:** Select one of the following options:
     - **No:** Disable the entry point for Secure Chat.
     - **At the start of chat:** Enable Secure Chat for the entry point and allow customers to authenticate their accounts using Chat Customer Single Sign-On before the chat activity is first created.
     - **During chat:** Enable Secure Chat for the entry point and allow customers to authenticate their accounts using Chat Customer Single Sign-On after the chat activity has begun and while interacting with an agent.

       The chat creation request must contain a SAML assertion. If the SAML assertion is missing, or is not valid for the entry point, chat requests are denied for that entry point. For details about secure chat configuration, see About Customer Single Sign-On.

Secure Chat, utilizing customer single sign-on functionality, allows chat entry points to transfer customer context information from the company website to the application through SAML. This allows customers who are already recognized on the company website to use a SSO-enabled entry point to chat with a customer without having to provide redundant information.

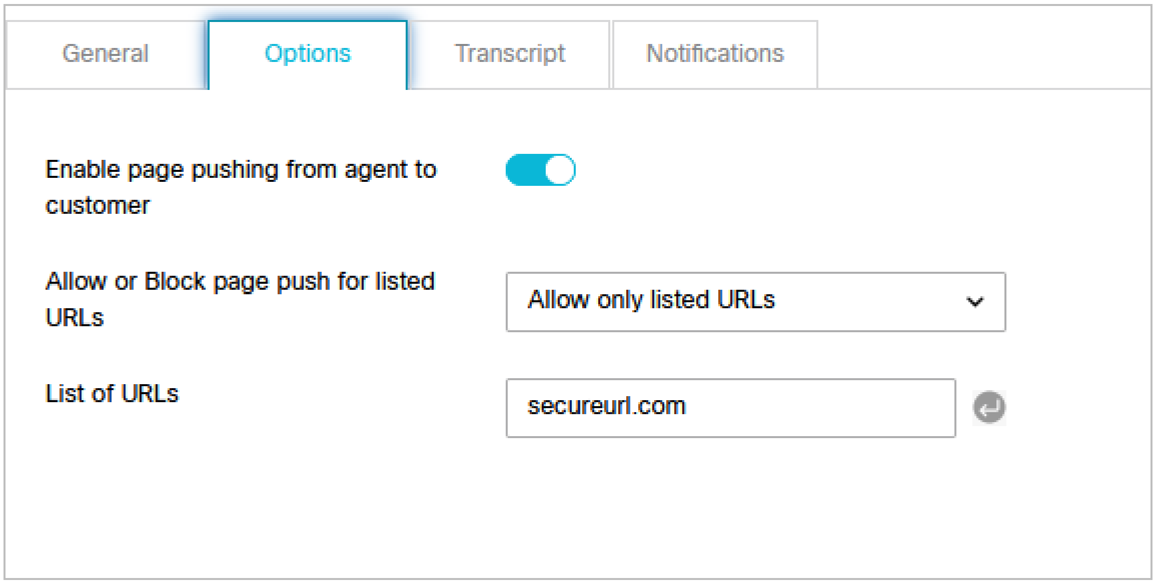
- **Agent Availability Required:** With this option you can decide if the agent availability is to be checked, when a customer initiate a chat session. When the switch is toggled **On**, customers will be able to initiate a chat session only if an agent is available (Availability setting from the agent desktop is selected) for handling chats.  If no agent is available, the off hours message is shown to customers. If the switch is toggled **Off**, Customers will be able to initiate chat sessions even when no agent (Availability setting from the Agent Console is not selected) is available for handling chats. Customers are not shown the off hours message and the login page is displayed.
- **Description:** Provide a description of the queue.




5. In the Options tab, provide the following:

- **Enable page pushing from agent to customer:** Click the toggle switch to allow agents to send web pages to customers. When page pushing is not allowed, the Page Push button in the Agent Console is hidden.

- **Allow or Block page push for listed URLS:** Select one of the options to determine if the URLs you are providing are to be **Allow only listed URLs**, **Block listed URLs**, and **Allow all URLs**.
- **List of URLs:** Provide the URLs that you are allowing or blocking agents from sending by typing the URL and pressing Enter. Repeat these steps to add more URLs.




6. Click the **Save** button. With the entry point saved, set the transcript emails and notifications. Then, test the entry point, and add the help link to your website.

# Setting Up Transcripts

> Transcripts must be set up as part of the creation process for a chat entry point.

You can email chat transcripts to customers. Transcripts can be sent for both serviced and abandoned chats. Serviced chats are those chat sessions where the agent joins the chat session at least once (activity substatus changes to "In Progress") before the customer exits the session. Serviced chats also include chats that were not completed successfully because of some error. Abandoned chats are those chat sessions where the customer exits the chat before the activity substatus changes to "In Progress" at least once. This means the customer leaves the chat before an agent could attend to him.

In the transcript email that is sent, the chat transcript is placed between the greeting article and the signature article.

> Before you start, make sure you create the header, footer, signature, and greeting KB articles to be used in the chat transcript email. For more information, see Creating Email Articles.

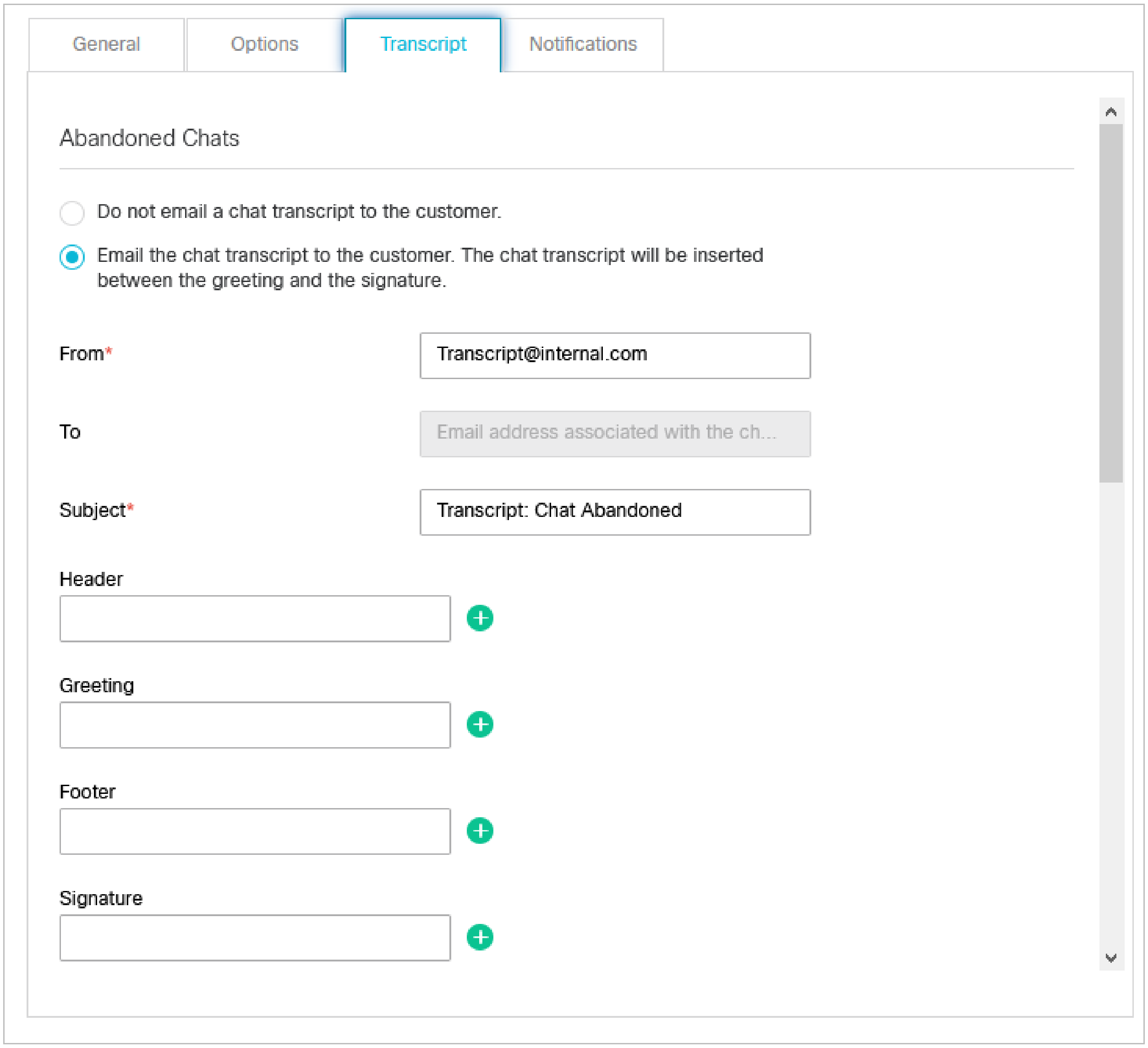
### To email chat transcripts to customers:

> These features are not available for callback and delayed callback activities.

1. In the department-level Top Menu, click **Apps.**
2. In the Left menu, click **Chat & Messaging > Entry Points**

3. Click the **New** button to create a new chat entry point.
4. Complete the steps outlined in the General and Options tabs
5. On the Transcript tab, go the Serviced chats or the Abandoned chats section and provide the following details.
    - **Do not email chat transcript to the customer:** If you want do to send a chat transcript, select this option. When you select this option, all other options will be disabled. This option is selected by default.
    - **Email the chat transcript to the customer. The chat transcript will be inserted between the greeting and the signature:** If you want to send a chat transcript, select this option. The chat transcript is inserted between the greeting and the signature.
    - **From:** Type the email address which you want to show in the **From** field in the email sent to the customer. This is required information. Any reply sent by the customer, in response to the chat transcript email, goes to the address specified in the **From** field. You can configure a standard email address such as "support@mycompany.com" as the **From** email address and configure it as the email alias to which customer reply emails will be sent.
        - **To:** The transcript email is sent to the email ID provided by the customer while logging in to the chat. If the customer has not provided an email ID, then the contact point will be picked up from the existing email field for the customer of the chat. If a customer has multiple email IDs, the last updated email ID of the customer will be considered. However, if no email ID is found at all, no transcript mail is sent out.

          > This field cannot be edited.
        - **Subject:** Type the subject of the email.
        - **Header:** Click the **Search and Add** button and select the header article you want to use from the menu. For more information about creating headers, see Creating Email Articles.
        - **Greeting:** Click the **Search and Add** button and select the greeting article you want to use from the menu. For more information about creating greetings, see Creating Email Articles.
        - **Signature:** Click the **Search and Add** button and select the signature article you want to use from the menu. For more information about creating signatures, see Creating Email Articles.
        - **Footer:** Click the **Search and Add** button and select the footer article you want to use from the menu. For more information about creating footers, see Creating Email Articles.
        - **Chat transcript content:** Specify what you want to include in the transcript. The following options are available.
            - **Include chat messages and URLs exchanged**
            - **Include only URLs exchanged**
            - **Include only chat messages exchanged**




4. Click the **Save** button.

# Setting Up Notifications

You must set up notifications as part of the creation process for a chat entry point.

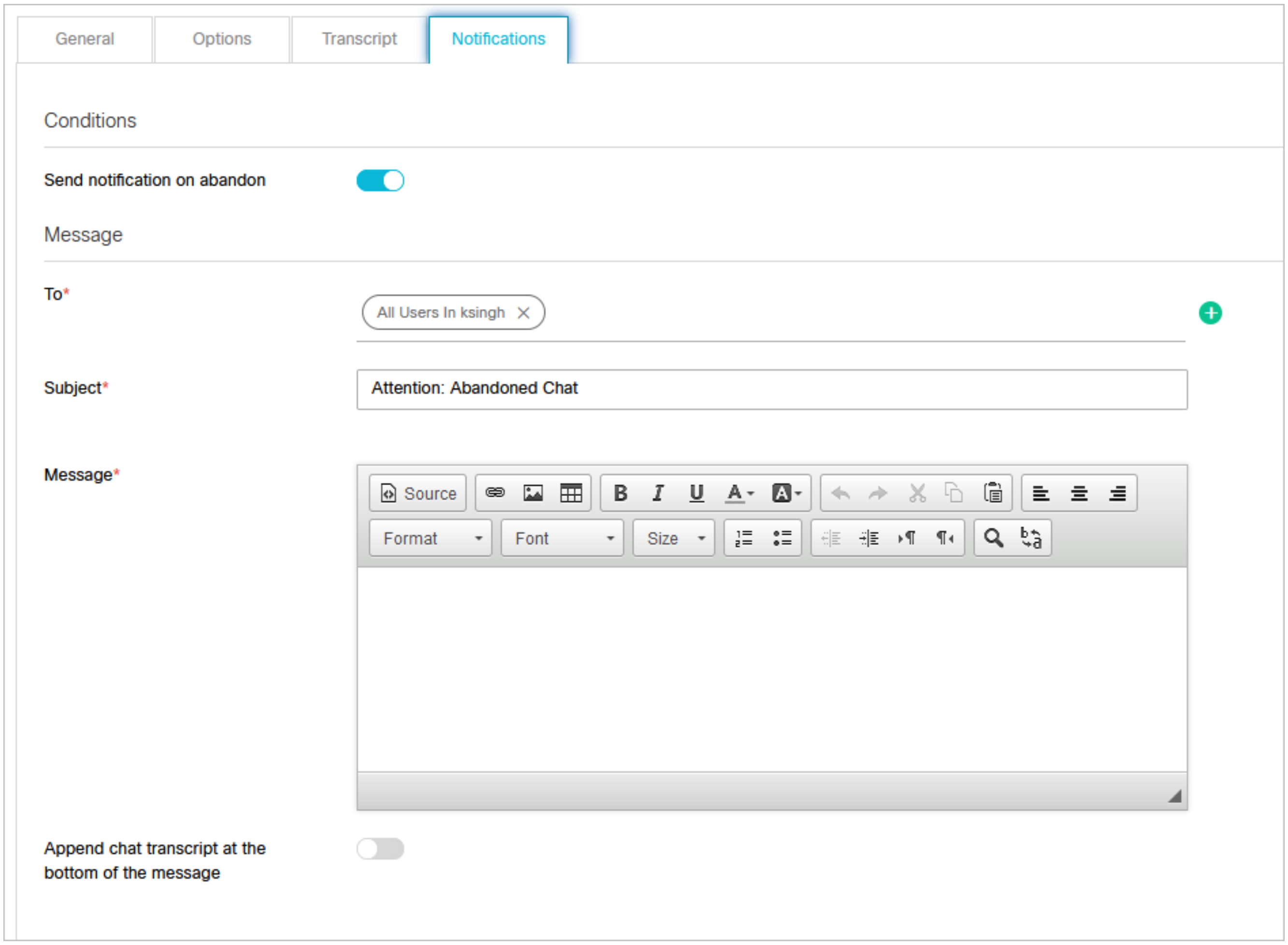
Notifications are messages sent to administrators using the messaging infrastructure. You can send a notification when a chat gets abandoned. Along with notifications, the transcript of a chat can also be sent. Notifications can be sent to internal users as well as to external email addresses.

The difference between a transcript and a notification is that a transcript is sent to the customer with whom the chat session is held and a notification is sent to administrators when a chat is abandoned. You can create an entry point without selecting the option of sending transcript or notification.

This feature is not available for callback and delayed callback activities.

## To set up notifications:

1. In the department-level Top Menu, click **Apps.**
2. In the Left menu, click **Chat & Messaging > Entry Points**.
3. Click the **New** button.
4. Complete the required fields in the General, Options, and Transcript tabs. This step is required.
5. On the Notification tab, go the Conditions section and click the **Send notification on abandon** switch to toggle if you want to send notifications for abandoned chats.
6. Next, on the Notification tab, go to the Message section, and provide the following details.
   - **To:** Select to whom you want to send the notifications by clicking the **Search and Add** button and performing the following:
     - In the Add user window, use the Users, User Group, Department, and External Email Addresses tabs to select recipients for notifications.
     - In the Message mode field for the selected recipient, click the dropdown to select whether to send notifications to internal, external, or both addresses for the recipients. This is required information.
     - Click the **Done** button.
   - **Subject:** Type a subject for the notification to be sent.
   - **Message:** Type a message in the text box. You can use the text box toolbar to edit the text.
   - **Append chat transcript at the bottom of the message:** Click this toggle to send the chat transcript with the notification.




7. Click the **Save** button.

# Testing Entry Points

After configuring an entry point, you can test it immediately to see how the entry point looks like.

The UI does not display the custom chat or callback templates or languages added for chat. Before you begin this task, get the list of template names and language files names that you have created on the web server: *Cisco_Home*`\eService\templates\chat` and *Cisco_Home*`\eService\templates\callback`.

## To test an entry point:

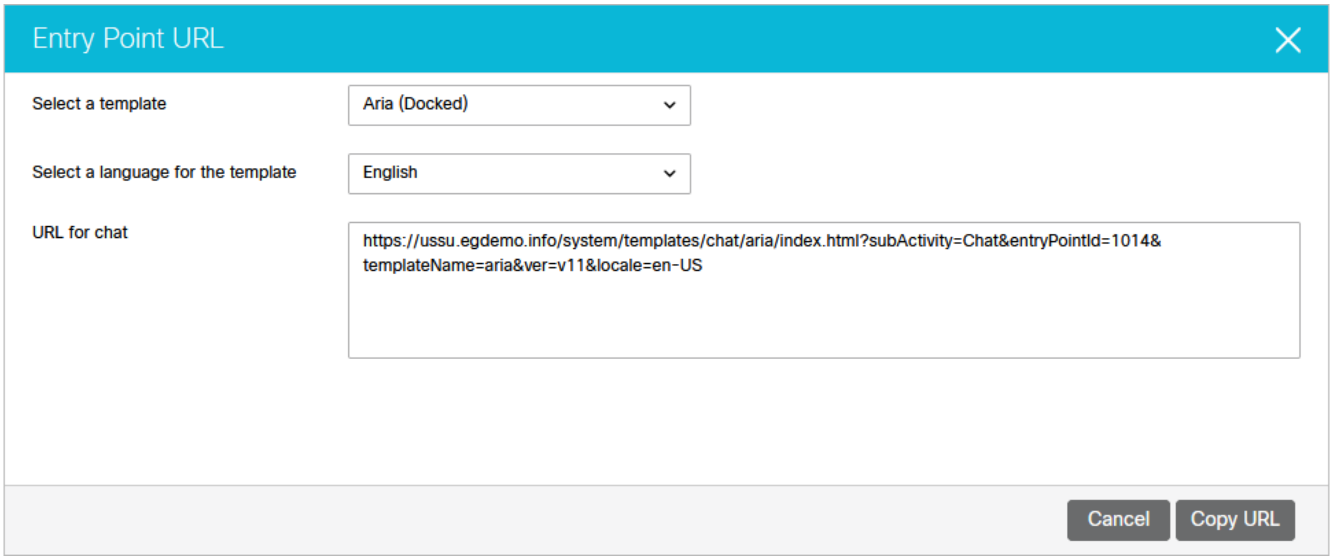
1. In the department-level Top Menu, click **Apps.**
2. In the Left menu, click **Chat & Messaging > Entry Points**.
3. Select an active entry point from the list of entry points available.
4. Click the **Entry Point URL** option.

> If you are creating a new entry point and want to test it, you have to save the entry point to be able to test it.

5. The Entry Point URL window opens. From the window, select the template you want to use for the entry point. This window only displays out-of-the-box templates and languages. If you have created custom templates or languages, you will need to edit the entry point URL.

> **Note**: If you are using a custom template made for a docked chat entry point, the entry point URL option does not display how the chat window appears on a webpage.

6. When you have made your selections, click the **Copy URL** button**.**




7. If you are using custom templates or languages, change the following parameters in the URL:

   Replace both occurrences of `aria, or rainbow` with the name of your custom template. For example, `PurpleNile.`

   Change the language code and country code in `languageCode=en&countryCode=US`. For example, `languageCode=it&countryCode=IT`.

   The sample chat URL will look like:

   http://demo1/pm/templates/chat/PurpleNile/chat.html?subActivity=Chat&entryPointId=1000&templateName=PurpleNile&languageCode=it&countryCode=IT&ver=v11

8. Open a new browser window and access the entry point URL and test it. To test the entry point with other templates available in the system, repeat the earlier steps and select a different template folder.

# Deleting Entry Points

Before you delete the entry point make sure you remove all the chat links related to the entry point from your website.

> The default entry point and active entry points cannot be deleted.

### To delete an entry point:

1. In the top menu, click **Apps**.

2. In the left menu, click **Chat & Messaging** > **Entry Points**.
3. On the Entry Point List page, locate the entry point you want to delete. Hover over the entry point, and click the **Delete** button.

# Turning Off Active Entry Points

Use this option to temporarily turn off all entry points for your department.

## To turn off active entry points:

1. In the department-level Top Menu, click **Apps.**
2. In the Left menu, click **Chat & Messaging > Entry Points**.
3. Click the **Stop New Chats** button.
4. Click **Yes** to confirm the decision. All the active entry points are turned off.
5. Click the **Enable New Chats** button, to turn on the entry points. When you click the button, it turns on the active entry points.

# Adding Chat and Collaboration Links to Websites

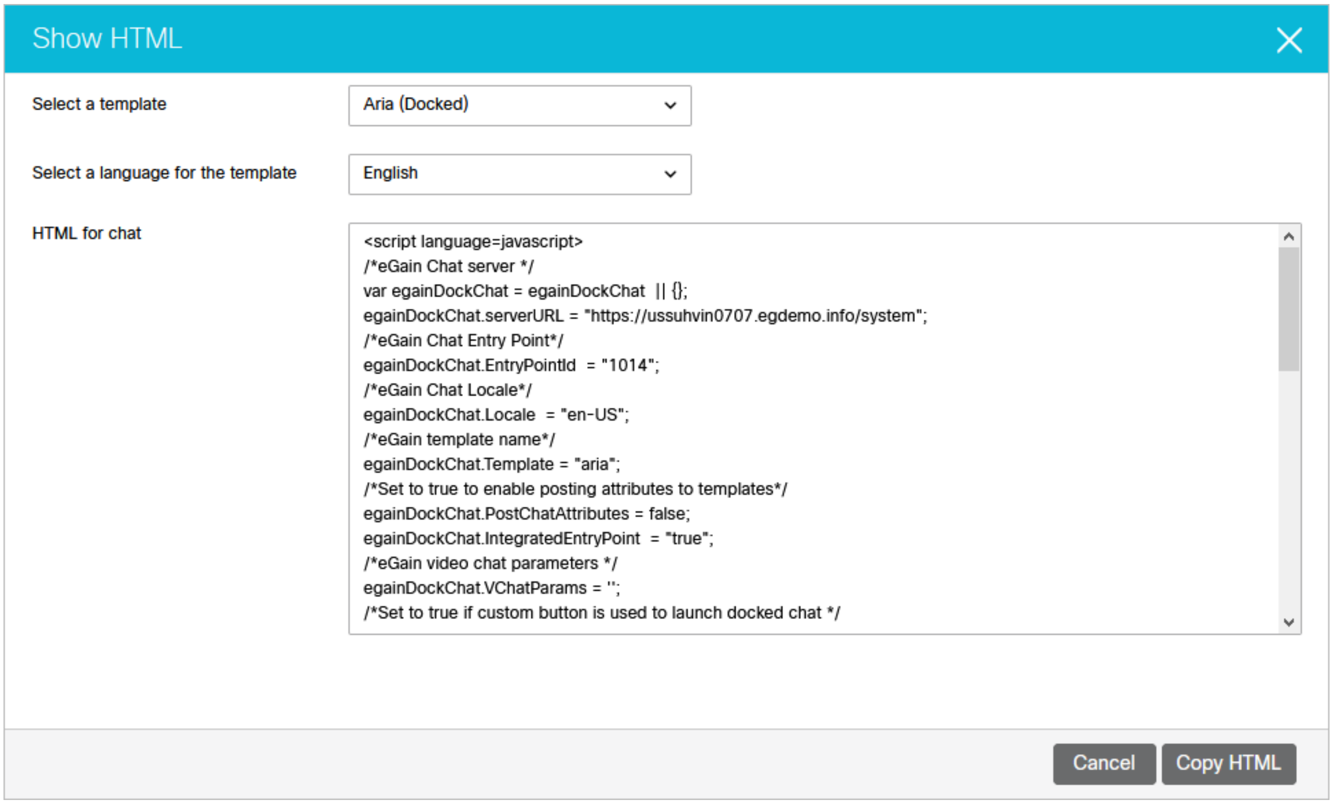
Docked chat allows website visitors to interact with customer service without creating additional browser windows. The docked chat frame always floats on top of the browser window, which lets the visitor continue to communicate with the agent without having to move between windows.

Docked chat utilizes a different set of templates than typical text chat. For more information about the docked chat templates, see About Chat Template Sets.

The UI does not display the custom chat templates or languages added for chat. Before you begin this task, get the list of template names and language files names that you have created on the web server: *Cisco_Home*`\eService\templates\chat.`

## To add docked chat to the website:

1. In the department-level Top Menu, click **Apps.**
2. In the Left menu, click **Chat & Messaging > Entry Points**.
3. Select the entry point you wish to add to the website.
4. Click the **Show HTML** button.
5. The Show HTML window opens. From the window, select the template for the entry point and select the Template Language from the dropdown list. Custom templates and languages are not displayed here. Click **OK**.




6. The HTML for the link appears. Here you can view and copy the HTML code of the entry point.

   You can only view or copy the HTML. It cannot be edited or deleted.

7. Before adding the code to your website, ensure that you replace the server name in the value of the `egainDockChat.ChatServerURL` property with the fully qualified domain name of the web server. If you are using custom templates and custom languages, change those values as well.

8. Locate `egainDockChat.EntryPointId` and check the value of the property to see if you need to replace the entry point ID.

9. Locate `egainDockChat.Locale` and check the value of the property to see if you need to replace the language-country codes with custom languages.

10. Locate `egainDockChat.Template` and check the value of the property to see if you need to replace the template name with custom templates.

11. Locate `egainDockChat.UseCustomButton` and `egainDockChatIsChatLaunched` and set the values to **true** and if you wish to allow the docked chat entry point to be initiated through a custom button or offer banner other than "Docked Chat".

12. Locate `egainDockChat.PostChatAttributes` and check the value of the property. By default, it is set to `false`. Set to `true` to enable posting attributes to the template in use and set any parameters you specified in application-chat-defaults. For example, if you wish to set customer parameters, you can set the parameters here:

    ```
    egainDockChat.PostChatAttributes = true;

    egainDockChat.SetCustomerParameters = function (){
    ```

```
egainDockChat.SetCustomerParameters("full_name","Test");

egainDockChat.SetCustomerParameters("email_address","test@test.com");

egainDockChat.SetCustomerParameters("subject","Test");
```

If you wish to enable secure chat for the entry point, you also need to set the `egainDockChat.PostChatAttributes` property to **true** and add the following:

```
egainDockChat.SetCustomerParameters("SAMLResponse","")
```

Secure chat requires this change to the entry point in addition to the chat template. For more details on configuring the template set for secure chat, see Enabling Customer Single Sign-On for Chat Templates.

13. Open the code view of the host web page and add the link code at the appropriate point. You may need to ask your webmaster to perform this task.

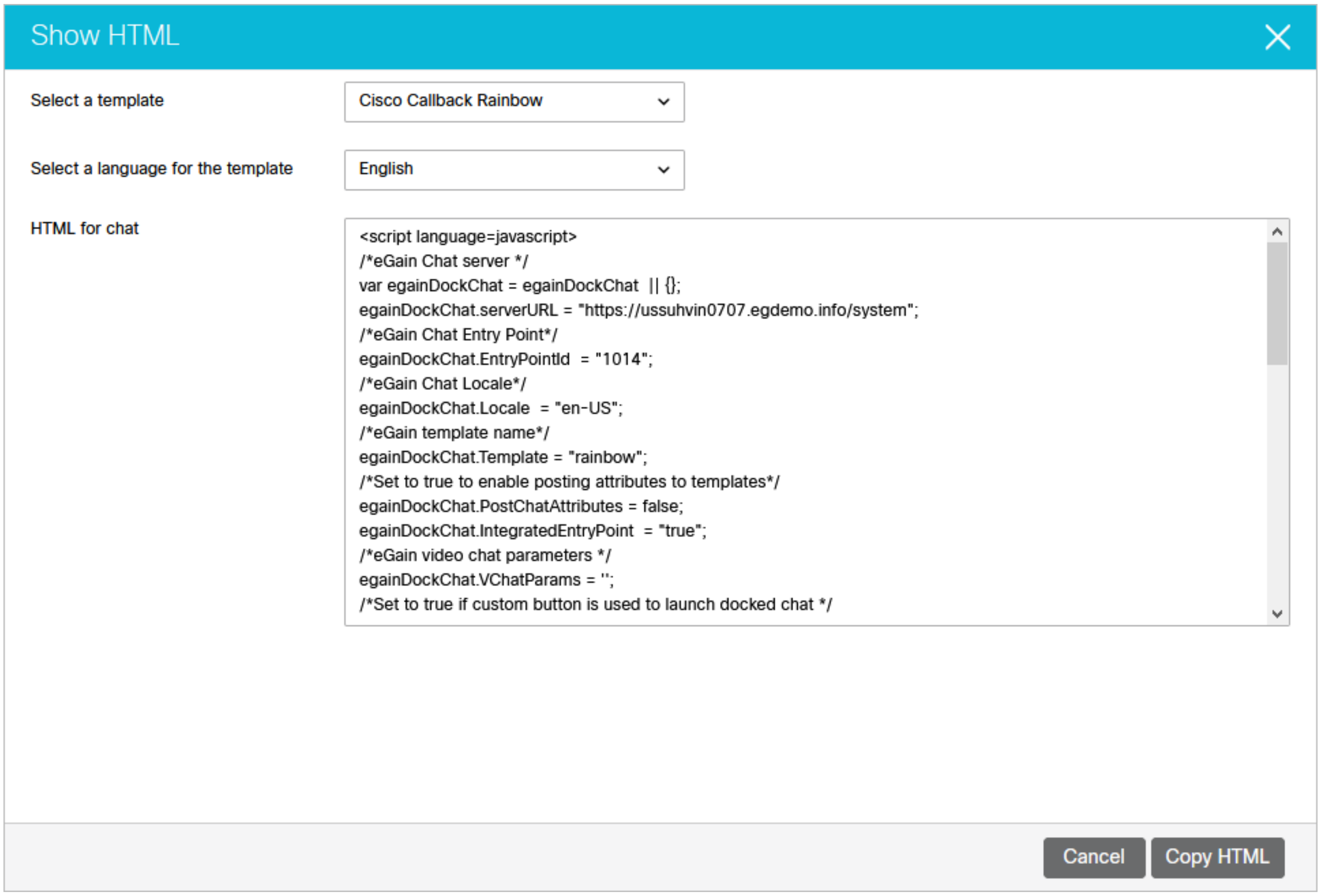
# Adding Callback Links to Websites

This section includes steps about how to enable auto-login for an entry point.

The UI does not display the custom templates or languages added. Before you begin this task, get the list of template names and language files names that you have created on the web server: *Cisco_Home*`\eService\templates\callback`.

## To add a callback link to a website:

1. In the department-level Top Menu, click **Apps.**
2. In the Left menu, click **Chat & Messaging > Entry Points**.
3. Select the entry point you wish to add to the website.
4. Click the **Show HTML** button.
5. The Show HTML window opens. From the window, select the template for the entry point and select the Template Language from the dropdown list. Custom templates and languages are not displayed here. Click **OK**.




6. The HTML for the link appears. Here you can view and copy the HTML code of the entry point.

> You can only view or copy the HTML. It cannot be edited or deleted. If you are creating a new entry point and want to view its HTML, you have to first save it to enable the Show HTML button.

7. Open the code view of the host web page and add the link code at the appropriate point. You may need to ask your webmaster to perform this task.

# Configuring Auto-Pushback Settings for Chats

The chat auto-pushback feature allows you to pushback chat activities to the queue, if the agents do not click on the new chats assigned to them in the configured time (default value is 2 minutes). Use these setting to enable auto-pushback and to define if agents should be made unavailable after a chat is pushed back automatically from the agent's inbox. This feature is available for both integrated and standalone chats.

> These change takes effect for all departments in the installation.

### To configure auto-pushback settings for chats:

- From the Partition Level settings, configure the Chat Auto-Pushback settings.

# Configuring Your Website for Chat Customer SSO

Chat templates should also be configured for chat customer single sign-on. For more information on configuring chat templates for Secure Chat with templates, see Enabling Customer Single Sign-On for Chat Templates.

### To configure your website for chat customer single sign-on:

1. Generate the HTML code for chat entry point. For more information about chat entry points, see About Entry Points.
2. Edit the `egainDockChat.postChatAttributes` parameter in the HTML code. Set the value of this parameter to **True**.

   > If using an undocked template, edit `egainChat` properties in place of `egainDockChat` parameters.

3. Add the following code immediately after the `</script>` tag in the generated HTML code:

```
<script language=javascript>

    // Customer information passed to ECE as name-value pairs. Base64 encoded SAML token is passed with the name "SAMLResponse"

    egainDockChat.storeChatParameters('SAMLResponse', '<Base64 encoded SAML assertion>'); // You MUST pass the SAML assertion with the name 'SAMLResponse' in this function

</script>
```

   > You need to add an additional code to get the Base64 encoded SAML assertion for the user that is logged on to your website.

4. If you want to transfer any additional attributes outside the SAML assertion, you can pass them as name-value pairs using the `egainDockChat.storeChatParameters` function as follows:

```
<script language=javascript>

    // Customer information passed to ECE as name-value pairs. Base64 encoded SAML token is passed with the name "SAMLResponse"

    egainDockChat.storeChatParameters('SAMLResponse', '<Base64 encoded SAML assertion>');

    egainDockChat.storeChatParameters('fieldname_2', '977-213-4444'); // SSN of the customer

    egainDockChat.storeChatParameters('fieldname_3', '4AZZXX7895463'); // account number of the customer

    //fieldname_2 and fieldname_3 are the second and third field respectively in the loginParameters array configured in the eGainLiveConfig.js file of the chat template
```

```
</script>
```

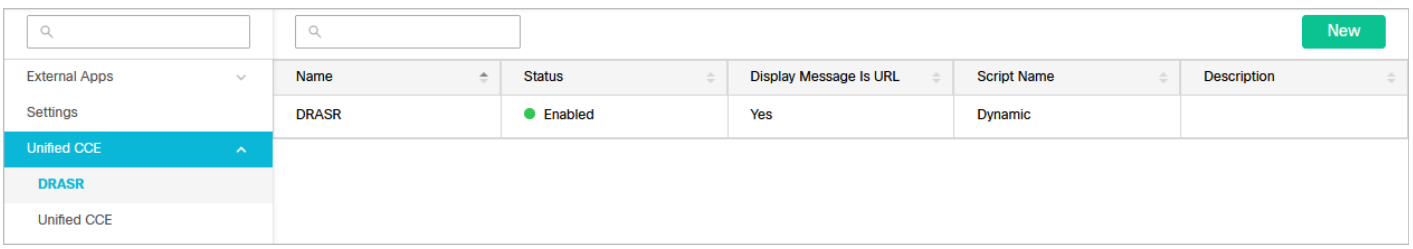
> The attributes transferred as name-value pairs outside the SAML assertion must not be the same as the attributes transferred in the SAML assertion. This will result in chat request being denied.

5. Save your changes.

# Configuring DRASR

Dynamic Run Application Script Request (DRASR) allows you to display messages with dynamic text (such as expected wait time) to customers while chat and call requests are being processed by the ECE and Unified CCE integrated systems. While ECE provides wait time messages for chat customers to be handled by ECE agents by default, DRASR should be used if:

- You do not want to use the default wait time string.
- You wish to display wait time value that is calculated in Unified CCE at the time of the Unified CCE script being run.




## Preparing to Create Dynamic Messages

You can use ECC variables and call variables to display the dynamic content. Dynamic messages can be displayed for chats and callback activities. Configuring the dynamic messages includes the following steps:

- From Unified CCE, configure the Network VRU scripts and use them in the Unified CCE scripts used for chat activities. You will need the name of the Network VRU script for configuring the dynamic messages. For details about doing these tasks, see the *Enterprise Chat and Email Deployment and Maintenance Guide*.
- From the ECE Administration Console, identify the integrated chat queues for which you want to display the dynamic messages.
- To display the dynamic content using ECC variable macros, prepare your macros by doing the following:
    - Refer to About Call Variables for the names of ECC variables that can be used.
    - Identify the ECC variables or call variables you want to use in the dynamic message and prepare the macros for the variables. For details about these objects and how they are used in Unified CCE, see the Scripting and Media Routing Guide for Cisco Unified ICM/Contact Center Enterprise & Hosted available here: https://www.cisco.com/en/US/products/sw/custcosw/ps1001/products_user_guide_list.html.

      The macro will be added in the format `%ECC` <*Variable_Name*>`%`. For example, `%ECCuser.ece.activity.id%`

While selecting ECC variables to be used in macros, make sure that the variables have valid values. If you use a variable that does not have a value, a run application script failure will occur and the customer will not be able to chat. The error template is displayed to the customer.

- To display the dynamic content using call variable macros, prepare your macros by doing the following: From the Administration Console, from the Call Variables tab of the queue properties, identify the call variables you want to use in the message and note down the number associated with the call variable. For example, in the following figure the number for `customer_phone_no` is 1 and for `activity_id`, the number is 2. The macro is added in the format `%CV`*Number*`%`. For example, `%CV1%` for `customer_phone_no`.

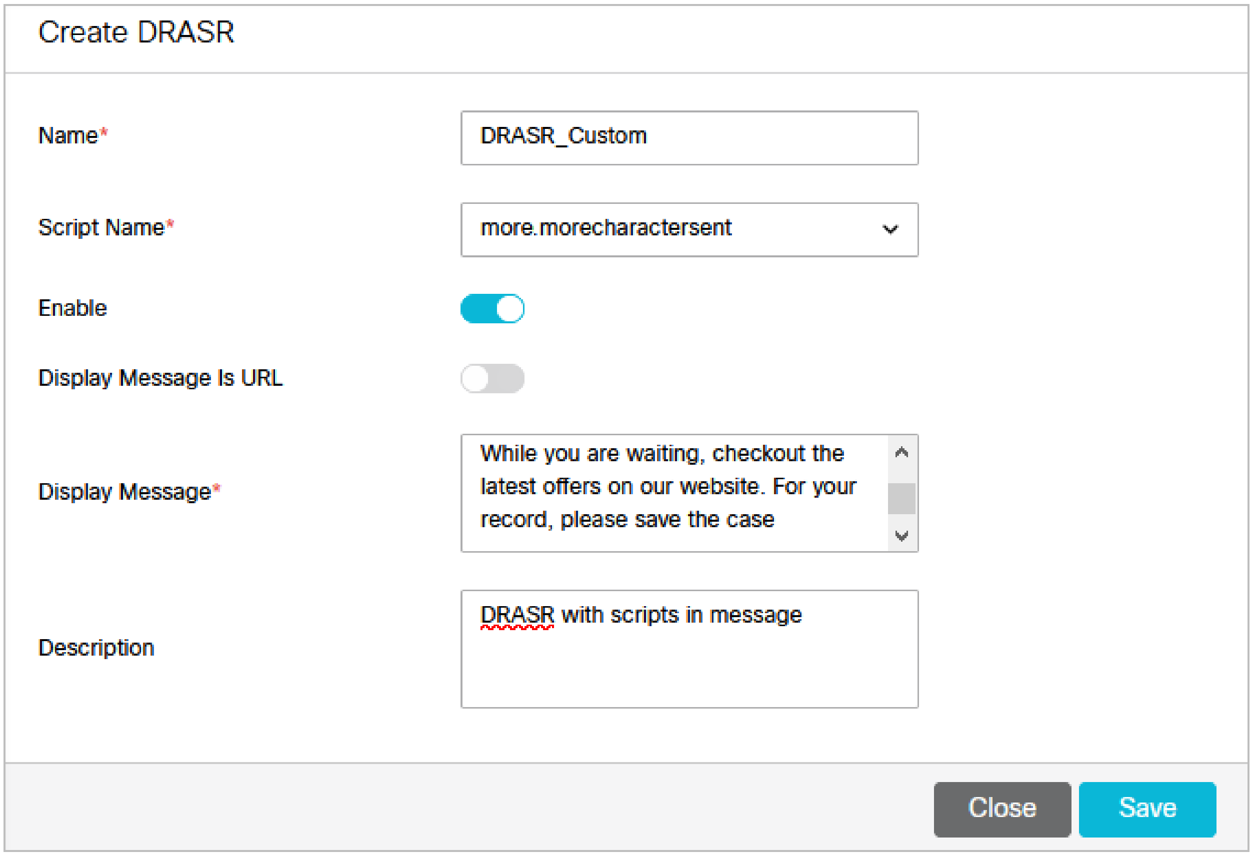
## Creating Dynamic Messages

### To create dynamic messages:

1. In the global-level Top menu, click the **Integration** option.
2. In the Left menu, navigate to **Unified CCE > DRASR**.
3. Click the **New** button.
4. In the Create DRASR space, set the following:
   - **Name:** The name of the script as it will be identified in the application.
   - **Script name:** From the dropdown, select the Network VRU script configured in Unified CCE. A script can be associated with only one message.
   - **Enabled:** Click the Toggle button to enable DRASR.
   - **Display message is URL:** If the message is a URL, set to **Yes**.

     If enabled, the message must be a valid URL or it will not display properly.

   - **Display message:** Provide the message that will be displayed to users upon accessing the entry point. If you have set the message to be a URL, provide a valid URL. In such cases, only provide the URL in the message and do not provide any text. A sample text message, with variable macros: `An agent is expected to be available in approximately %ECCuser.wait.time% minutes. While you are waiting, checkout the latest offers on our website. For your record, please save the case number %CVcase_ID%`
   - **Description:** The description of the script.




5. Click the **Save** button.

# Integration Using Chat Messaging APIs



## About Chat Messaging API Integration

Chat Messaging APIs can be used to build external integrations with custom messaging apps, messaging channels like Facebook Messenger, and building custom web templates. The chats created from these Chat Messaging APIs are routed to agents like regular web chats and have the web chat features available.

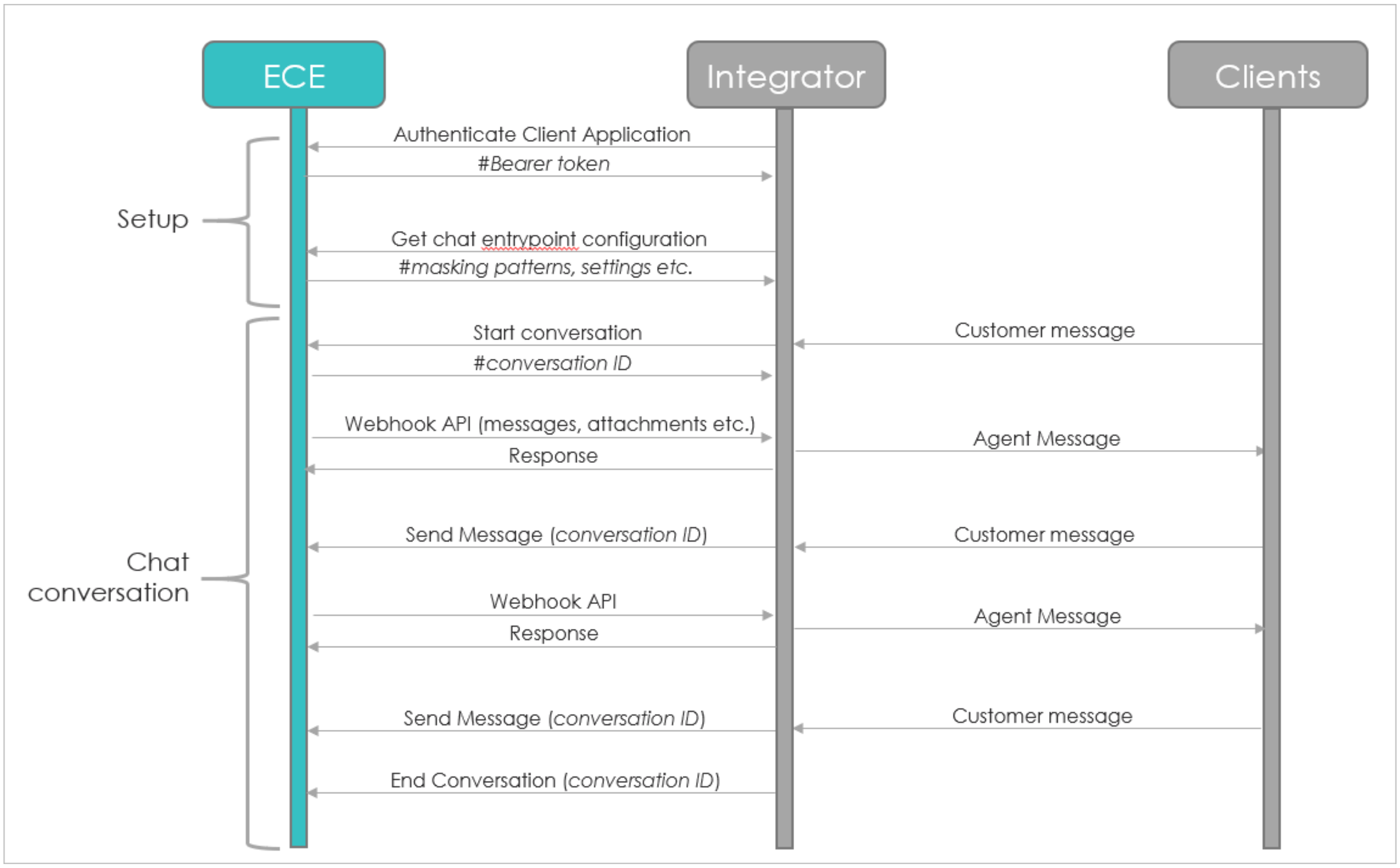
The client applications for the integration is setup from the Enterprise Chat and Email Administration console.

In order to use chat messaging APIs, the following must first be performed:

1. Register a client application. See Registering Applications for Chat API Integration.
2. Register a Webhooks Callback URL to receive the chat messages and events. See Registering Applications for Chat API Integration.
3. Associate the client application with a chat entry point. See Creating Messaging Adapters.
4. Create an end to end chat session using the REST APIs: Authenticate Client Application, Get Entry point Configuration, Start Conversation, Send Chat Message, End Conversation. For more information about these APIs, see *Enterprise Chat and Email Developer's Guide to Web Service APIs*.

### Post Integration Configuration

Once you have registered the client application, added the callback URL, and configured messaging adapter, you must use the Chat Messaging APIs to build the integration. For more information, see the Enterprise Chat and Email Knowledge Authoring and Interaction API Reference Guide. For more information about the postconfiguration process, see also the following diagram.



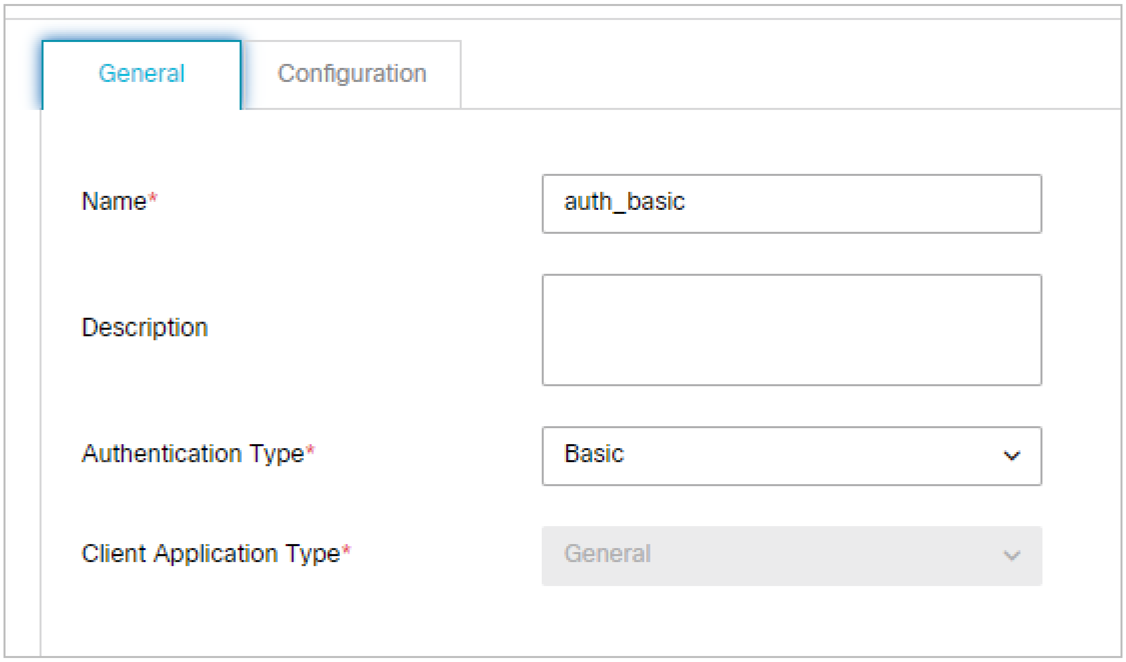

With the client application registered and the messaging adapter created, you can follow the integration flow

# App Authentication

## Configuring Basic Authentication

### To create a basic app authentication configuration:

1. In the global-level Top Menu, click the Integration option.
2. In the Left menu, navigate to **External Apps** > **Authentication**.
3. Click the **New** button.
4. In the Create Authentication workspace, on the General tab, provide the following:
   - **Name:** The name for the configuration.
   - **Description:** A brief description of the configuration.
   - **Authentication type:** Select **Basic**.




5. Click the Configuration tab. On the Configuration tab, provide the following details:
   - **User name:** The user name for the account the data adapter will use for authentication.
   - **Password:** The password for the account the data adapter will use for authentication.
6. Click the **Save** button. This configuration can now be used when registering apps for Callback. For more information, Registering Applications for Chat API Integration.

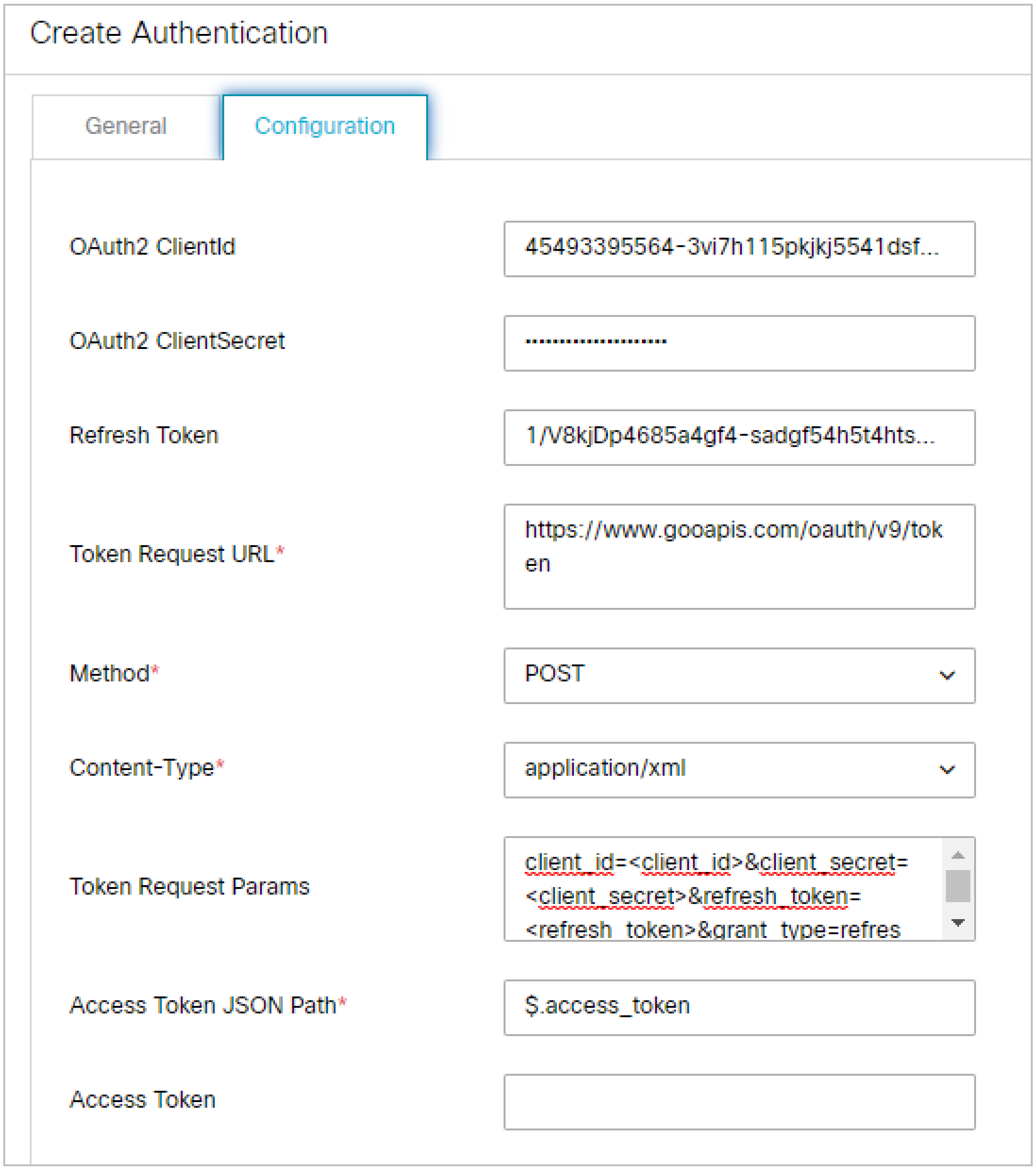
# Configuring OAuth 2.0 Authentication

## To create an OAuth 2.0 app authentication configuration:

1. In the global-level Top Menu, click the Integration option.
2. In the Left menu, navigate to **External Apps** > **Authentication**.
3. Click the **New** button.
4. In the Create Authentication workspace, on the General tab, provide the following:
   - **Name:** The name for the configuration.
   - **Description:** A brief description of the configuration.
   - **Authentication type:** Select **OAuth 2.0**.
5. Click the Configuration tab. On the Configuration tab, provide the following details:
   - **OAuth2 ClientId:** The unique string representing the registration information provided by the service provider.
   - **OAuth2 ClientSecret:** The unique string provided by the service provider that acts as a means of authorizing a client that is requesting an access token.

- **Refresh Token:** Credentials used to obtain new access tokens when the current access token expires.
- **Token Request URL:** Provide the access token request URL and click OK to close the window. This field is required.
- **Method:** Select the method in which the access token is retrieved. Select either **Post** or **Get**. The method selected depends on what the client authentication server supports.
- **Content-Type:** Select the format of the token. Options include: `application/json`, `application/xml`, and `application/x-www-form-urlencoded`.
- **Token Request Params:** Provide the parameters for the token request body. Format the parameters string property with variables and values to ensure that correct information is sent in the request.

  For example: `client_id=<client_id>&client_secret=<client_secret>&refresh_token=<refresh_token>&grant_type=refresh_token`
- **Access Token JSON Path:** The JSON query path of an attribute in the returned JSON response. For example, `$.access_token`.
- **Access Token:** If you wish to provide a current access token, enter the string in the field. Leave this field blank if you do not have an Access Token. New tokens are requested by the application using the provided details.
- **Headers:** Provide any necessary additional values to include in the response header by entering a Name and Value for the field and clicking the **Add** button.




6. Click the **Save** button. This configuration can now be used when registering apps for Callback. For more information, Registering Applications for Chat API Integration.

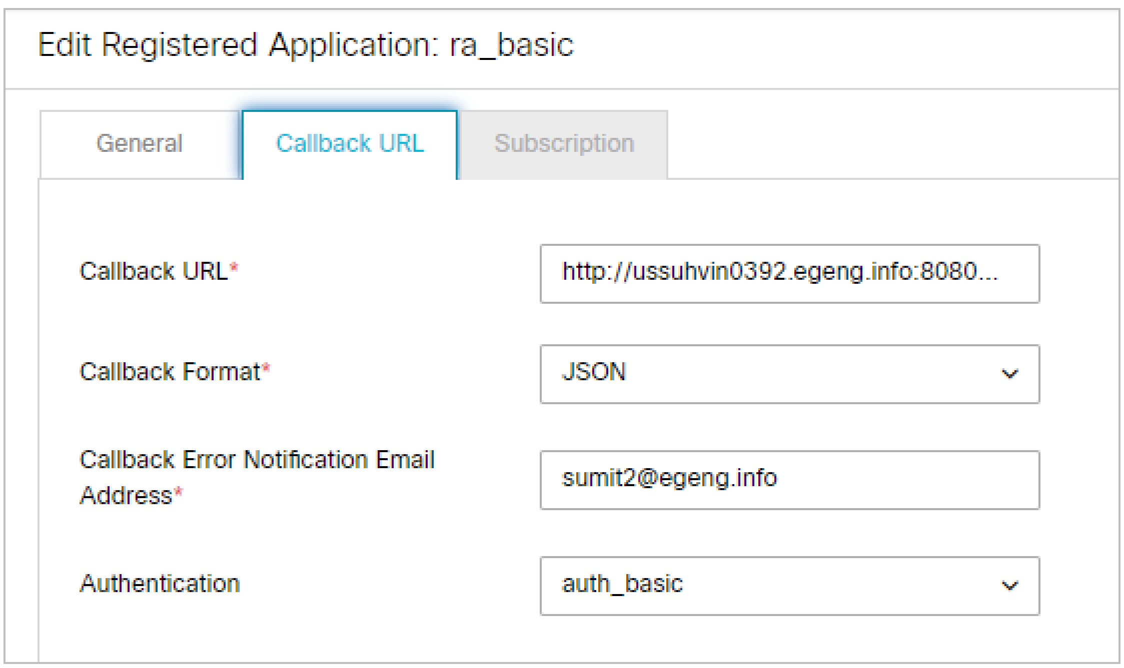
# Registering Applications for Chat API Integration

While registering a client application, you must also register a Webhooks Callback URL to receive the chat messages and events.

### To register an application for chat API integration:

1. In the global-level Top Menu, click the Integration option.
2. In the Left menu, navigate to **External Apps** > **Register Applications**.

3. Click the **New** button.
4. In the Create Registered Application workspace, on the General tab, provide the following:
   - **Name:** Type the name of the application. This is required information. For example, “Messaging App Integration”.
   - **Application URL:** This field is disabled and cannot be changed if **Chat** is the selected Application Type.
   - **Application Type:** This is required information. Select **Chat** from the dropdown to integrate Chat APIs for custom messaging channels, mobile messaging, and so on.
   - **Description:** Type a brief description.
5. Click the Callback URL tab. On the Callback URL tab, provide the following:
   - **Callback URL:** The Webhooks Callback URL to receive the chat messages and events. The same callback URL can be configured for multiple client applications.
   - **Callback Format:** Select the format of the messages or events that are posted to the callback URL. Options available include: **XML** and **JSON**.
   - **Callback Error Notification Email Address:** Provide the address to which email notifications are sent if the call to the callback URL fail.
   - **Authentication:** Select an authentication configuration to use for callback. For more information about creating authentication configurations, see App Authentication.




6. Click the **Save** button. This generates the client key and client secret for the server configuration in the General tab. If you need to regenerate the client secret and key, click the **Regenerate Client Secret** button.

# Creating Messaging Adapters

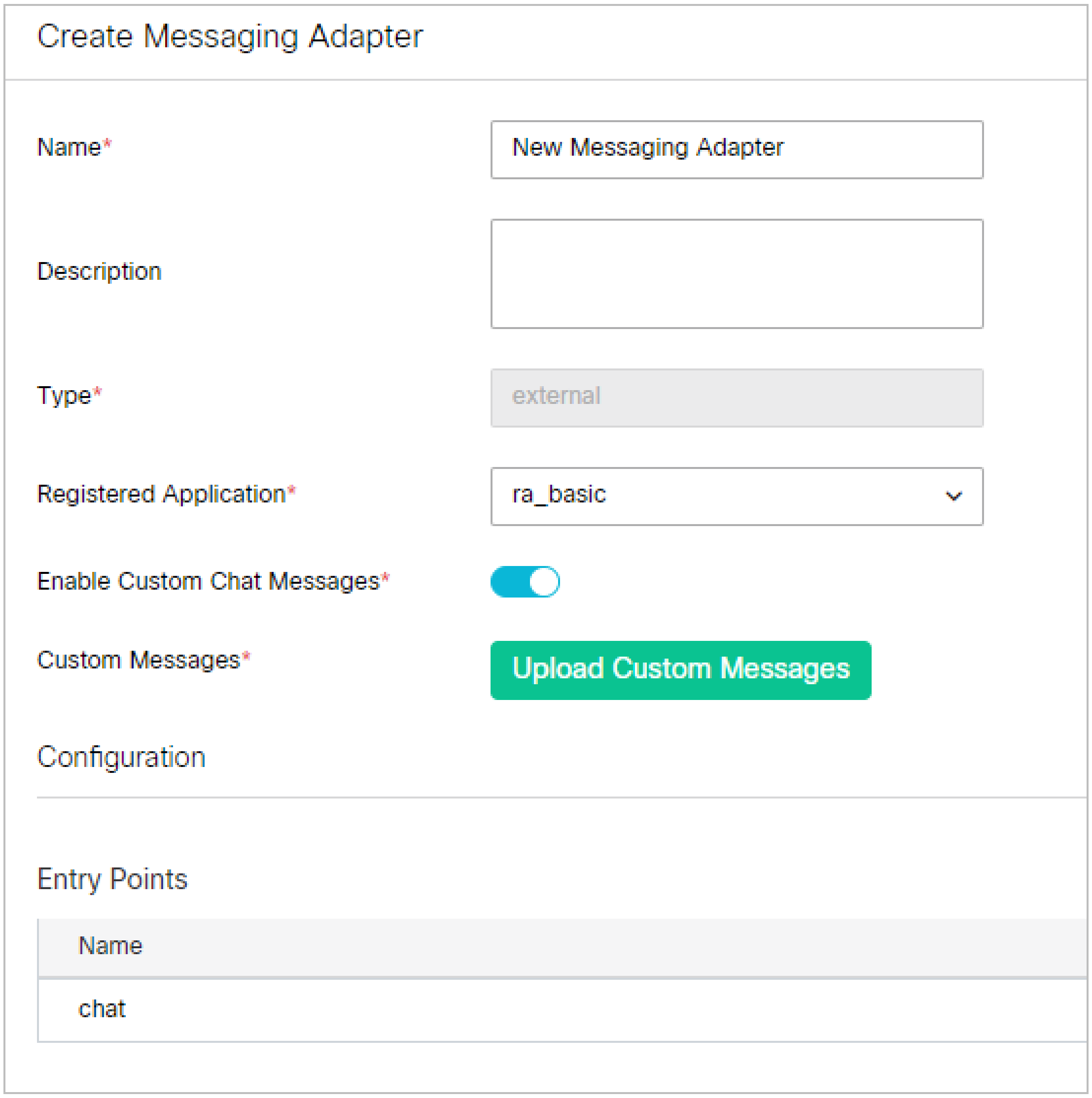
Once the client application has been registered, along with the Webhooks Callback URL, the messaging adapters must be created. By creating messaging adapters, you are mapping client applications to a chat entry point.

For more information about registering client applications, see Registering Applications for Chat API Integration.

### To create messaging adapters:

1. In the department-level Top Menu, click **Apps.**
2. In the Left menu, click **Chat & Messaging > Messaging Adapters**.
3. Click the **New** button.
4. The workspace refreshes to show theCreate Messaging Adapter page. Provide the following details:
   - **Name:** Name of the messaging adapter.
   - **Description:** Description of the messaging adapter.
   - **Type:** This is set to 'External' and cannot be changed.
   - **Registered Application:** Select a registered application from the dropdown. For more information about registering applications, see Registering Applications for Chat API Integration.
   - **Enable Custom Chat Messages:** Click the **Toggle** button to enable custom chat messages.

     Click the **Download Default Messages** button to download a zip file containing the set of default chat messages for the default languages of the application. If you wish to use custom chat messages, edit the downloaded default messages and save them to a local directory to upload in the Custom Messages field.
   - Custom Messages: This field is enabled if the **Enable Custom Chat Messages** toggle button is enabled. Click the **Upload Custom Messages** button to browse to and select the file to upload your custom messages.




5. Click the **Save** button.

# Appendix A: Aqua Chat Templates

# About Aqua Chat Template Sets

> The Aqua template set for chat has been deprecated. Upgraded systems can continue to use the templates, but no new features will be available on these templates, going forward.

Template sets consist of CSS (cascading style sheets) and JS (JavaScript) files. The CSS files control the look and feel of the Chat Console. The JS files contain the business logic used to render data in the Chat Console. Templates are also used to determine what information is requested to identify the customer (e.g. name, email address, phone number). You can also compose messages that the customer will see under certain circumstances (e.g. if they request a chat session out of hours).

An entry point can be mapped to multiple template sets and multiple entry points can use the same template set.

> The system supplied templates should not be deleted.

The template sets are managed from the web server and not from the Administration Console.

## Do I Need to Create New Templates or Just Use the Default One?

There are templates shipped with the application, but you may still want to create a new one that is customized to look like your website.

You can choose to use one template set for all your chat entry points for a consistent look and feel. Alternatively, you can use different templates, so that the look and feel of each one is targeted towards a specific audience.

## How Easy are the Templates to Modify?

Templates are easy to modify. All the properties of the templates are modified using the files available on the web server. It is possible to make a number of layout and wording changes as well as branding changes to match them with the look and feel of your website.

Some important things to note:

- You must have a good knowledge of the HTML and CSS languages to make changes in the HTML and CSS files.
- Always create a backup copy of the files before making any changes.

The following table describes the files that can be changed for the **Aqua** template. All the files are available on the web server at *Cisco_Home*`\eService\templates\chat\Aqua`.

| Use | To |
|---|---|
| `\l10n\`*Language-Locale.json* | ▪ Update the text displayed on the chat pages. |
| `\application-chat-defaults.js` | ▪ Enable anonymous chats. |

| Use | To |
|---|---|
| | ▪ Enable auto login.<br>▪ Enable surveys.<br>▪ Configure the maximum length allowed for customer messages.<br>▪ Configure sound alert for chat messages. |
| `\css\appication.0.0.0.css`, `iframe-style.css` | ▪ Configure the look and feel of the chat pop-up and the chat typing area. |
| `\index.html`<br>`\transcript\transcript.html`<br>`\components` folders<br>▪ `aqua\components\alternate-contact-options`<br>▪ `aqua\components\article-content`<br>▪ `aqua\components\article-toolbar`<br>▪ `aqua\components\chat-unavailable-message`<br>▪ `aqua\components\error-message`<br>▪ `aqua\components\escalation-search-results-list`<br>▪ `aqua\components\footer-small`<br>▪ `aqua\components\header-small`<br>▪ `aqua\components\launch-button`<br>▪ `aqua\components\message-input-horizontal`<br>▪ `aqua\components\post-chat-survey` | ▪ Configure the look and feel of the chat pages, the chat typing area, and the text editor toolbar.<br>▪ Change the images used.<br>▪ **Note**: The size of the custom images should match the size of the default images. However, if you are changing the size of the chat frame, you can adjust the images size to match the new frame. |

| Use | To |
|---|---|
| ▪ `aqua\components\pre-chat-params-list`<br>▪ `aqua\components\status-bar`<br>▪ `aqua\components\thanks-message`<br>▪ `aqua\components\transcript`<br>▪ `aqua\components\wait-screen` | |

# Creating Aqua Template Sets

You need access to the web server to perform this task.

> Anytime you update or create new templates, you must copy the templates files on all web servers in your installation.

Create a separate template set for the following types of activities using the **Aqua** template:

- Docked chat activities

### To create a new Aqua template set:

1. On the Web Server, browse to *Cisco_Home*`\eService\templates\chat.`
2. Create a copy of the **Aqua** folder and rename the folder.
3. The new template set is created. You can now edit the files in the template folder to configure your template settings and to customize the look of your templates. For a list of Aqua template files that can be changed, see About Chat Template Sets.

After creating a template set, some of the things that you can do are:

- Configure the login page parameters
- Enable anonymous chats
- Set the maximum length of chat messages
- Configure the knowledge base link
- Configure the survey questions
- Change the off hours message

# Deleting Aqua Template Sets

Repeat these tasks on all web servers in your installation.

## To delete a template set:

1. On the web server, browse to *Cisco_Home*`\eService\templates\chat`.
2. Delete the template folder that you do not want to use. If you are using these templates for any entry point, make sure to update the HTML code on the web pages with the new template sets you want to use.

# Configuring Login Parameters

Build the web form which customers must fill in to initiate a chat. The system uses the customer's contact details (for example, email address, name, and so on) to identify existing customers and add the chat session to their customer records automatically. It also creates customer record for new customers.

You can:

- Change the text of the options
- Remove fields
- Change the order of fields
- Change error messages

## To configure the login page parameters:

1. Open the *Cisco_Home*`\eService\templates\chat\`*Template_Folder*`\application\application-chat-defaults.js` file in a text editor.
2. Locate the `LoginParameters` property. From here, you can do the following:
    a. To remove an existing field, select the code for the field and delete it from the file. For example, to remove the name from the login page, delete the following lines:

    ```
    {
    "eGainAttributeName": "full_name",
    "eGainParentObject": "casemgmt",
    "eGainChildObject": "individual_customer_data",
    "primaryKey": false,
    "controlType": "text",
    "label": "APP.EG_FULLNAME",
    ```

```
"required": true,

"minLength": "1",

"maxLength": "120",

"validationPattern": "[^<>]+$"

},
```

> Do not remove a field that is set as the primary key.

b. To change the required fields, locate the attribute which you want to change and do the following:

   a. For the attribute, locate the property `required` and change the value to **true** to make it a required field, or change the value to **false** to make it an optional field.

   b. If you are setting a field as required, you must set the minimum length allowed for the field.

   c. In the attribute section, locate the property `minLength` and set a minimum length for the field. You can also set the maximum length by configuring the `maxLength` setting.

c. By default, the email address is set as the primary key for the login page. Also, note that only one field can be set as the primary key. The attribute that is set as the primary key needs to be marked as a required field.

d. To enable the use of selections in dropdowns for autologin, the values of the `egainChat.SetCustomerParameters(fieldname,value)` parameter need to be passed as such:

   a. **Single-select:** `fieldname` is the string “fieldname_index of login parameter” and value is the internal value of selected option. For example, `egainChat.SetCustomerParameters('fieldname_1','savings')`

   b. **Multi-select:** `value` is an array of selected values. For example, `egainChat.SetCustomerParameters('fieldname_1',["abc","xyz"])`

3. Open the *Cisco_Home*`\eService\templates\chat\`*Template_Folder*`\l10n\`*Language-Locale*`.json` file in a text editor. Make the following changes:

   - You can change the text of the fields displayed on the login page. Locate the following properties and provide new labels for the fields.
     - `EG_FULLNAME`**:** Change the label of the name field.
     - `EG_EMAIL_ADDRESS`**:** Change the label of the email field.
     - `EG_PHONE_NUMBER`**:** Change the label of the phone number field.
     - `EG_QUESTION`**:** Change the label of the text field.
     - `EG_CHARACTERS_REMAINING`**:** Change the text of the message displayed to indicate the character count for the text box on the chat interaction page. Make sure you do not delete `{{characterCountRemaining}}` from the message.

- `EG_MESSAGE_LENGTH_ERROR`**:** Change the text of the warning message to be displayed when the message of the customer exceeds the allowed length. As part of the message, if you want to display the allowed length and the number of characters the user has typed, make sure you retain `(MESSAGE_LENGTH)` and `(MAX_LENGTH)` in the new message.

4. Save the changes.

5. Clear the browser cache and access the chat link to test the changes.

# Adding New Fields to the Login Page

Along with full name, email address, and phone number, you can use custom attributes on the login page.

Custom attributes created for the following objects can be added to the login screen.

- Customer Data
- Contact Point Data
- Activity Data

The following table lists the fields that can be added to the login page. It also lists the attribute names and object names for each field. You will need this information while adding new fields to the login screen.

| For | Parent Object | Child Object | Attribute Name |
|---|---|---|---|
| Full Name | casemgmt | individual_customer_data | full_name |
| Email Address | casemgmt | email_address_contact_point_data | email_address |
| Phone Number | casemgmt | phone_number_data | phone_number |
| Subject Custom attributes added to activity data | casemgmt | activity_data | subject<br>If the custom attributes added to the object has two words, replace the space with underscore (_). For example, if the attribute created in the application is `More details`, the attribute name will be `more_details.` |
| Custom attributes added to customer data | casemgmt | customer_data | If the custom attributes added to the object has two words, replace the space with underscore (_). For example, if the attribute created in |

| For | Parent Object | Child Object | Attribute Name |
|---|---|---|---|
| | | | the application is `Family name`, the attribute name will be `family_name`. |
| Custom attributes added to contact point data | casemgmt | contact_point_data | If the custom attributes added to the object has two words, replace the space with underscore (_). For example, if the attribute created in the application is `Mobile number`, the attribute name will be `mobile_number`. |

## To add new fields:

1. Open the *Cisco_Home*`\eService\templates\chat\`*Template_Folder*`\application\application-chat-defaults.js` file in a text editor and do the following:
   a. Locate the `loginParameters` property, and copy a section for one of the existing parameters and paste it where you want to add the custom field. Edit the following properties.
      - `eGainAttributeName`**:** Provide the attribute name. See the table for details about the attribute name you need to provide.
      - `eGainParentObject`**:** Provide the parent object name. See the table for details about the object name you need to provide.
      - `eGainChildObject`**:** Provide the child object name. See the table for details about the object name you need to provide.
      - `primaryKey`**:** Set the value as **false.**
      - `controlType`**:** Set the format for the field. For a single line text field, set the value to **text**. For a multi-line text field, set the value to **textarea**. For a single-select dropdown, set the value to **singleselect**. For a multi-select option, set the value to **multiselect**.
      - `label`**:** Provide the label  from the l10n language file for the field. For example, for the phone number field, use `APP.EG_PHONE_NUMBER`.
      - `required`**:** If you want to set the field, as optional set the value as **false.** To make the field mandatory, set the value as **true.**
      - `minLength`**:** Set the minimum required characters for the field.
      - `maxLength`**:** Set the maximum allowed characters for the field. The value should not be set more than the value set in the application while creating the custom attribute.
      - `validationPattern`**:** Provide the expression against which the information provided by the customer is to be checked. For example, for phone number you can give an expression like, `^\s*\(?\s*\d{3}\s*\)?\s*[-.]?\s*\d{3}\s*[-.]?\s*\d{4}\s*$`. And, for email you can use the expression like, `^[0-9a-zA-Z\.\-\_]+\@[0-9a-zA-Z\.\-]+$` - this expression refers to

the format X@Y. Z (john@mycompany.com). For fields that require an integer value, you need to at least set the validation string as `\\d+$`.

   b. Save the changes.

2. Open the *Cisco_Home*`\eService\templates\chat\`*Template_Folder*`\l10n\`*Language-Locale*`.json` file in a text editor and do the following:

   a. Add the following line in the `APP:{ }` list in the file: `"EG_`*Name*`" : "`*Display name for the attribute*`",`

      `Where: EG_`*Name* should match the value set in **label** in step 1.

   b. Save the changes.

3. Clear the browser cache and access the chat link to test the changes. The login page should show the new fields.

# Enabling Anonymous Chats

Anonymous chats allow customers to chat with agents without providing any personal details, like name, email address, phone number, and so on. When the customer clicks the chat link or button, he is not displayed the login page. Anonymous chats can be enabled for the following:

- Chat entry points

### To enable anonymous chats for chat entry points:

1. Open the *Cisco_Home*`\eService\templates\chat\`*Template_Folder*`\application\application-chat-defaults.js` file in a text editor.

2. Locate the `loginParameters` property and delete all the parameters between the square brackets. The property will look like: `loginParameters : []`

3. Save the changes.

4. Clear the browser cache and access the chat link to test the changes. The chat should start without showing the login page.

# Enabling Auto-Login

When the chat link is provided in an authenticated section of the website, the auto-login feature can be used to automatically login the customer for chat. When the customer clicks the chat link or button, the login page is not displayed.

> A template where auto-login is enabled should not be used for regular chats.

### To enable auto-login for chats:

1. Open the *Cisco_Home*`\eService\templates\chat\`*Template_Folder*`\application\application-chat-defaults.js` file in a text editor.
2. Locate the `EnableAutoLogin` property and set the value to `true` to enable auto-login.
3. Save the changes.

## Configuring Sub-Domain Navigation Options

The Aqua template, by default, allows for website visitors to chat with agents while browsing pages hosted on a single domain. To allow for visitors to chat with agents while navigating to other web pages hosted on other sub-domains, further configuration is required. The template must be configured to store chat session data in cookies, thus allowing visitors to freely navigate so other sub-domains without ending the chat session.

> It is not recommended to use multiple chat templates on a domain if only one template has been configured for sub-domain navigation.

### To configure templates for sub-domain navigation:

1. Open the *Cisco_Home*`\eService\templates\chat\`*Template_Folder*`\application\applicationchat-defaults.js` file in a text editor.
2. Locate the `MultiSubDomainSupport` object and adjust the following properties:
   - `IsRequired`: By default, the values is set to **false**. Set the value to **true** to enable storage of chat session data in cookies.
   - `CookieParentDomain`: Provide the parent domain in which the cookies are stored. For example, `help.com`. This is required if `IsRequired` is set to **true**.
   - `CookiePath`: Provide the common URL path of all the sub-domains that should be supported for continuing the chat, for example: `/chat`. This prevents the cookie data from being sent over to the server with every request. This is required if `IsRequired` is set to **true**.
   - `IsCookieSecured`: Determine whether the cookies should only be used for HTTPS domains. Set to **true** to allow cookies only to be used in HTTPS connections, or set to **false** to allow cookies to be used in HTTP and HTTPS connections.
3. Save the changes.
4. Clear the cookies and browser cache before accessing the chat link to test the changes.

# Setting Maximum Length for Chat Messages

### To set the maximum length for chat messages:

1. Open the *Cisco_Home*`\eService\templates\chat\`*Template_Folder*`\application\application-chat-defaults.js` file in a text editor.
2. Locate the `MaxMessageSize` property and change the value to the number of characters you want to allow for each chat message. By default the value is set to **800**. This setting is also used to control the message length in the **Additional Comments** field in the survey form. The maximum value can be set to **2000**.
3. Save the changes.
4. Clear the browser cache and access the chat link to test the changes.

# Configuring Aqua Templates for Customer SSO

Chat entry points can be configured to transfer customer context information from the company website to the application through SAML. To learn more about Secure Chat for customers, see Chat Customer Single Sign-On.

### To configure chat templates for customer single sign-on:

1. On the web server, navigate to *Cisco_home*`\web\templates\chat\`*Template_Name***** and open `application-chat-default.js`.
2. Enable auto-login by setting the value of the `EnableAutologin` parameter to **true**.
3. Configure the attributes that will be transferred to the application along with the chat creation request. You can do this by configuring the `loginParameters` property for each of the attributes as described in Configuring Login Page Parameters.
4. If any of the attributes will be transferred to the application in the SAML assertion, specify the following additional properties for the attribute in `loginParameters`:
   - `providerAttributeName`: Name of the attribute in the SAML assertion.
   - `secureAttribute`: Set the value to **true** to indicate that this attribute will be transferred in the SAML assertion. For example:

     ```
     {
     "providerAttributeName”: ”my_name”,
     "secureAttribute": true,
     "eGainAttributeName": "full_name",
     "eGainParentObject": "casemgmt",
     "eGainChildObject": "individual_customer_data",
     ```

```
"primaryKey": false,

"controlType": "text",

"label": "APP.EG_FULLNAME",

"required": true,

"minLength": "1",

"maxLength": "120",

"validationPattern": ""
}
```

If `secureAttribute` is missing or is set to **false**, the application does not look for and extract information for that attribute from SAML assertion. Similarly, if `secureAttribute` is set to **true** for an attribute, but either `AttributeName` is not provided, or an attribute with the given name is missing from SAML assertion, chat creation requests are denied.

# Configuring Alternative Engagement Options

This feature is available only for standalone chats.

Out-of-the-box, templates come with three alternative engagement options for the chat window. These options are displayed to customers while they are waiting in queue (new chats or transferred chats) to be serviced by an agent. Once an agent joins the chat, the options are removed from the chat window. You can display these options as soon as the customer starts the chat, or after a delay. To enable these options, see Creating Queues.

- Give us a call
- Send us an email
- Visit our FAQ page

These options have placeholder links. You must update these links before using this feature. You can also:

- Update the text.
- Remove options.
- Add new options.

### To update links on the page:

1. Open the *Cisco_Home*`\eService\templates\chat\`*Template_Folder*`\application\application-chat-defaults.js` file in a text editor.

2. In the file, search for `AlternateEngmtParams` and in this section set the following properties:
   a. Search for `text :'APP.EG_ALT_ENGMT_PHONE'` and for the URL parameter provide the link for the callback page.
   b. Search for `text :'APP.EG_ALT_ENGMT_EMAIL'` and for the URL parameter provide the email address which customers can use to send emails. Make sure you do not delete the `mailto:` text.
   c. Search for `text :'APP.EG_ALT_ENGMT_FAQ'` and for the URL parameter provide the link to the FAQ page on your website.
3. Clear the browser cache and access the chat link to test the changes.

## To update the text on the page:

1. Open the *Cisco_Home*`\eService\templates\chat\`*Template_Folder*`\l10n\`*Language-Locale*`.json` file in a text editor and do the following:
   a. Locate the `EG_ALT_ENGMT` properties and change the value of these properties to the new message you want to display for the alternative engagement options.
   b. Similarly, locate the `EG_ALT_ENGMT_PHONE, EG_ALT_ENGMT_EMAIL`, and `EG_ALT_ENGMT_FAQ` properties and update the text for each of the options, such as: "Give us a call", "Send us an email", and "Visit our FAQ page".
   c. Save the changes.
2. Clear the browser cache and access the chat link to test the changes.

## To remove an option:

1. Open the *Cisco_Home*`\eService\templates\chat\`*Template_Folder*`\application\application-chat-defaults.js` file in a text editor.
2. In the file, search for `AlternateEngmtParams` and from this section delete the code lines for the help option you do not want to display. For example, to remove the **Send us an Email** option, delete the following lines of code:

```
{
  label :'APP.EG_ALT_ENGMT_EMAIL',
  image:'components/alternate-engagement-options/icon_email.png',
  hoverImage:'components/alternate-engagement-options/mail_hover.png',
  url:'mailto:support@company.com'
},
```

3. Clear the browser cache and access the chat link to test the changes.

### To add new options:

1. Open the *Cisco_Home*\eService\templates\chat*Template_Folder*\application\application-chat-defaults.js file in a text editor.
2. In the file locate the AlternateEngmtParams property, and copy a section for one of the existing help options and paste it where you want the new section. Edit the following properties.
   - label**:** Create a new parameter using the name as EG_ALT_ENGMT_*Name* from l10n*Language-Locale*.json.
   - image**:** Set the value as 'components/alternate-engagement-options/*icon_email.png*'
   - hoverImage**:** Set the value as 'components/alternate-engagement-options/*mail_hover.png*'
   - url**:** Provide the URL for the option.
3. Save the changes.
4. Place the images to be used for the new section at the following location: *Cisco_Home*\eService\templates\chat*Template_Folder*\components\alternate-engagement-options
5. Open the *Cisco_Home*\eService\templates\chat*Template_Folder*\l10n*Language-Locale*.json file in a text editor and do the following:
   a. Add the following line in the property file: EG_ALT_ENGMT_*Name*="*Text for the section*" Where EG_ALT_ENGMT_*Name* should match the value set in text in step 2.
   b. Save the changes.
6. Clear the browser cache and access the chat link to test the changes.

## Changing General Settings

### To change the general properties for templates:

1. Open the *Cisco_Home*\eService\templates\chat*Template_Folder*\l10n*Language-Locale*.json file in a text editor.
2. Edit the values for the following properties:
   - TITLE**:** Change the title of the chat window. The default value is Live Collaboration by Cisco.
   - EG_CONFIRM_CLOSE_CHAT**:** Change the message displayed to the agent when the customer closes the chat session.
   - EG_START_CHAT**:** Change the text of the button for starting chat. The default value is **START CHAT**.
   - EG_SEND**:** Change the text of the button for sending the first chat message. The default value is **Send**.
   - EG_SUBMIT**:** Change the text of the button for sending chat messages. The default value is **Submit**.

3. Save the changes.
4. Clear the browser cache and access the chat link to test the changes.

### To change the header and footer colors:

This section talks about changing the header and footer of all pages in the Chat Customer Console, that is, the Login page, the Chat page, and the Survey page.

1. Open the *Cisco_Home*`\eService\templates\chat\`*Template_Folder*`\css\application.0.0.0.css` file in a text editor.
2. To change the header color, locate the `#egain-chat-header-small` class and change the value of the `background` property.
3. To change the footer color, locate the `#egain-chat-footer-small` class and change the value of the `background` property.
4. Save the changes.
5. Clear the browser cache and access the chat link to test the changes.

### To change button colors:

This section talks about changing button colors for **Start Chat, Send,** and **Close** buttons.

1. Open the *Cisco_Home*`\eService\templates\chat\`*Template_Folder*`\css\application.0.0.0.css` file in a text editor.
2. Locate the `.action-footer .btn` class and change the value of the `Background` property.
3. Save the changes.
4. Clear the browser cache and access the chat link to test the changes.

## Changing the Size and Placement of the Chat Frame

This section talks about defining the size and placement of the chat frame.

### To change the size and placement of the chat frame:

1. Open the *Cisco_Home*`\eService\templates\chat\`*Template_Folder*`\css\iframe-style.css` file in a text editor.

2. Edit the values for the following properties:
   - `#egain-chat-wrapper`: Set the size and position of the chat section in a standard sized frame.
   - `#egain-chat-wrapper.small`: Set the size and position of the chat sections in a smaller frame.
   - `#egain-chat-wrapper.minimized`: Set the size and position of the chat sections in a minimized frame. By default, this area is left blank so the content follows the frame.
   - `#egain-chat-iframe`: Set the size and position of the chat frame.
3. Save the changes.
4. Clear the browser cache and access the chat link to test the changes.

# Configuring Off the Record Options

When data masking is enabled, masking rules prevent agents and customers from sending sensitive information like credit card numbers, social security numbers, and so on during the chat session. The Off the record feature allows agents and customers to exchange sensitive information as data masking rules do not apply to these messages. Any information exchanged while off-record is not stored in the system. Off the record conversation can be started and stopped only by customers. This feature is enabled while configuring data masking for the chat channel. For details, see Applying Patterns to Chat Channel.

### To change the text for off-the-record messages:

1. Open the *Cisco_Home*`\eService\templates\chat\`*Template_Folder*`\l10n\`*Language-Locale*`.json` file in a text editor.
2. To change the text that appears when a customer goes on-record and off-record, locate the following properties and change the values:
   - `EG_OFF_RECORD`: The default value is *Your messages are now off-record and will not be saved in the system.*
   - `EG_ON_RECORD`: The default value is *Your messages are now on record.*
3. To change the text that appears when data is masked, change the value of the following property::
   - `EG_SENSITIVE_DATA_NOTIFICATION`: The default value is *Don't worry. Your sensitive data has been masked.*
4. *T*o change the text that appears in the chat transcript when messages have been removed due to masking, change the value of the following property:
   - `EG_OFF_RECORD_MESSAGE_REMOVED`: The default value is *Message has been removed.*
5. Save the changes.
6. Clear the browser cache and access the chat link to test the changes.

# Changing the Off Hours Message

An off hours message can be displayed to customers when agents are not available for handling chats. Alternatively, you can display the More Help Options page as the off hours page.

### To change the off hours message:

1. Open the *Cisco_Home*`\eService\templates\chat\`*Template_Folder*`\l10n\`*Language-Locale*`.json` file in a text editor.
2. Locate the property `EG_CHAT_UNAVAILABLE` and change the message that is displayed when no agents are available to handle chats.
3. Save the changes.
4. Clear the browser cache and access the chat link to test the changes.

# Changing Sound Alert

A sound alert is played on the customer side when a new chat message comes. You can change this sound alert.

### To change the sound alert:

1. Open the *Cisco_Home*`\eService\templates\chat\`*Template_Folder*`\application\application-chat-defaults.js` file in a text editor.
2. Locate the `CustomerAudioAlert` property and change the value of the property to the path of the new chat sound file. Only `.wav` files are supported.
3. Save the changes.
4. Clear the browser cache and access the chat link to test the changes.

# Configuring Survey Questions

At the end of the chat session a survey form is presented to customers to measure the usefulness of the chat. The survey form has three questions and a text box for providing additional comments. You can change the text of these questions, however, you cannot add or remove questions from the survey form.

If a survey is not necessary, it can be turned off as well. For details, see Turning Off Surveys.

> If the installation is integrated with Cisco WXM, this post chat survey is not displayed when the chat ends and is instead replaced with the WXM survey form. Therefore, this setting does not apply to WXM survey

> forms if the installation is integrated with Cisco WXM. For more information about integrating with Cisco WXM, see About Certificate Management.

### To change the text of survey questions:

1. Open the *Cisco_Home*`\eService\templates\chat\`*Template_Folder*`\l10n\`*Language-Locale*`.json` file in a text editor.
2. Locate the section for survey questions by searching for `EG_SURVEY`. In this section, change the following properties:
   - `EG_SURVEY_HEADING`**:** Change the text for presenting the survey form.
   - `EG_SURVEY_QUESTION_1`, `EG_SURVEY_QUESTION_2`, **and** `EG_SURVEY_QUESTION_3`**:** Change the text of the survey questions.
   - `EG_SURVEY_QUESTION_4`**:** Change the title of the textbox.
   - `EG_MESSAGE_LENGTH_ERROR`**:** Provide the message to be displayed when the text in the comments box exceeds the allowed size. Keep in mind this is a general message that is used in the chat session and on the survey page when a user exceeds the character limit for the message.
   - `EG_THANKS_SURVEY_MESSAGE`**:** Change the message displayed to users after they have completed the survey.
   - `EG_SEND`**:** Change the text of the button used for submitting the survey. Keep in mind this is a general button which may be used in chat pages other than just the survey section.
3. Save the changes.
4. Clear the browser cache and access the chat link to test the survey form changes.

## Turning off Surveys

You can turn off the survey if you do not wish to display a survey form at the end of chat sessions.

> If the installation is integrated with Cisco WXM, this post chat survey is not displayed when the chat ends and is instead replaced with the WXM survey form. Therefore, this setting does not apply to WXM survey forms if the installation is integrated with Cisco WXM. For more information about integrating with Cisco WXM, see About Certificate Management.

### To turn off surveys:

1. Open the *Cisco_Home*`\eService\templates\chat\`*Template_Folder*`\application\application-chat-defaults.js` file in a text editor.
2. In the file locate the `SurveyPageOn` property and change its value to **false**.
3. Save the changes.

4. Open the *Cisco_Home*`\eService\templates\chat\`*Template_Folder*`\l10n\`*Language-Locale*`.json` file in a text editor.
5. In the file locate the `EG_THANKS_CHAT_MESSAGE` property and change the text to be displayed to customers when they end the chat session. The default value is `"Thank you for chatting with us."`
6. Save the changes.
7. Clear the browser cache and access the chat link to test the changes.

# Configuring Aqua Templates for WXM Surveys

If the installation is integrated with Cisco WXM, the WXM survey form is displayed at the end of the chat session to allow the customer to provide feedback.

For more information about integrating with Cisco WXM, see About Certificate Management.

### To configure Aqua templates for WXM surveys:

1. Open the *Cisco_Home*`\eService\templates\chat\`*Template_Folder*`\application\application-chat-defaults.js` file in a text editor.
2. Locate the `SurveyPageOn` property and change the value of the property to `true` to enable WXM surveys at the end of chat interactions that use Aqua templates.
3. Save the changes.

# Changing Chat Transcript Options

During a chat session and at the end of a chat session, customers have access to the Save Transcript button. Customers can use this to save the chat transcript for future reference.

You can change the tool tips and you can add a header, footer, and title for the transcript window. You cannot remove buttons from the chat window.

> These settings do not apply to the  transcripts that are emailed to customers. The settings for the transcript emails are configured through the entry points properties.

### To change the chat transcript options:

1. Open the *Cisco_Home*`\eService\templates\chat\`*Template_Folder*`\l10n\`*Language-Locale*`.json` file in a text editor.
2. Edit the values for the following properties:
   - `EG_TOOLBAR_SAVE_TRANSCRIPT`**:** Change the tool tip of the **Save Transcript** button.
   - `EG_TRANSCRIPT_HEADER`**:** Provide the header for the transcript page.

- EG_TRANSCRIPT_START_TIME**:** Change how you want the start time of the transcript to be displayed. Do not change the value of the {{chatStartTime}} property.
- EG_CHAT_TRANSCRIPT_HEADER**:** Change how you want the customers to see which agent they're talking to. Do not change the value of the {{AgentName}} property.

3. Save the changes.
4. Open the *Cisco_Home*\eService\templates\chat*Template_Folder*\l10n\messaging_*Language_Locale*.properties file in a text editor.
5. Edit the value for the following property:
   - MAIL_HEADER_LIVE_CHAT**:** Change the header of the transcript emailed by the system.
6. Save the changes.
7. Clear the browser cache and access the chat link to test the changes.