



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

# Deployment Guide

Testing and Troubleshooting the Co-browse Solution

12/13/2025

# Testing and Troubleshooting the Co-browse Solution

## Contents

- 1 Testing and Troubleshooting the Co-browse Solution
  - 1.1 Testing Co-browse Without Chat
  - 1.2 Testing Co-browse With Chat
  - 1.3 Troubleshooting the Intranet Problem in IE10 and IE11
  - 1.4 Troubleshooting the "proxy instrumentation problem"
  - 1.5 Troubleshooting CSS Synchronization
  - 1.6 Troubleshooting Chat Widget Rendering Issues

Use the following procedures to test that a Genesys Co-browse solution is configured correctly.

### Tip

Also see [Restrictions and Known Limitations](#).

## Testing Co-browse Without Chat

### Required Components

The following components are the minimum required to test Co-browse without chat: The following components are the minimum required to test Co-browse without chat:

- Local Control Agent
- Configuration Server
- Solution Control Server
- Genesys Administrator
- Co-browse Server

See compatible versions in [Related Components](#).

### Preparing for Testing

#### Prerequisites

- Genesys Framework is running.
- Co-browse Server is installed and configured to work without chat. See [Installation Procedures](#).
- To allow Co-browse Server to work without chat, set the Co-browse Server option `useChat` in the [chat section](#) to false.

#### Start of Procedure

1. Start the required servers.
2. After each server starts, check its trace log for errors.

#### End of Procedure

### Testing the Co-browse Solution Without Proxy

Instrument your website with the Co-browse JavaScript snippet. See [Basic Instrumentation](#).

## Testing the Co-browse Solution With Proxy

To learn how to use the proxies included in the Co-browse installation package, see [Test with the Co-browse Proxy](#).

## Testing Co-browse Instrumentation

### Prerequisites

- The Co-browse JavaScript snippet is on your website. See [Basic Instrumentation](#).

### Start of Procedure

1. Open an instrumented page in a supported browser — this page is referred to as the Master page.

### End of Procedure

**Successful result:** The Live Chat and Co-browsing buttons are present on the page.

**Problem:** The buttons are absent.

### Possible causes of the problem

1. The page is incorrectly instrumented. To verify, open the page source and confirm the instrumentation script is present and correct. For details, see [Website Instrumentation](#). If you use the proxy for testing Co-browse, it might be a proxy problem. Refer to [Troubleshooting the "proxy instrumentation problem"](#) for details and a workaround.
2. Co-browse Server is not running or not working properly. Check the Co-browse Server trace log.
3. Localization settings for the Master page are incorrectly specified in the instrumentation script. For details about localization settings, see [Localization](#).
4. The network has a problem.

### Important

To further investigate a problem, enable the Master browser console log in your instrumentation script. For details, see [Enabling console logs](#).

## Testing Co-browse Session

### Opening the Slave Page

**Prerequisites:** You have successfully completed [Preparing for Testing](#).

### Start of Procedure

1. Open the Slave page using `http://<Co-browse_Server_host>:<port>/cobrowse/slave.html`.  
For example: `http://localhost:8700/cobrowse/slave.html`

### End of procedure

**Successful result:** The Slave page opens and has an edit box for the Session ID.

**Problem:** The edit box does not appear.

### Possible causes of problem:

1. The URL is incorrect.
2. The localization settings for the Slave are incorrectly specified in the `slave` section of the Co-browse Server application. For details about localization settings, see [Localizing the slave UI](#).
3. The network has a problem.
4. If it is not clear what the problem is, enable debug logging on the Slave browser, open the Developer Console, and reload the page. You will see debug logs in the Developer Console.

### Important

To enable the debug log in the Slave browser, insert `debug=1` into the Slave URL and reload the page. For details, see [Enabling Console Logs](#).

### Next Step

- [Starting a Co-browse Session](#)

### Starting a Co-browse Session

**Prerequisites:** You have successfully completed [Opening the Slave Page](#).

### Start of procedure

1. In the Master page, click Co-browsing.

### End of procedure

**Successful result:** The Co-browse session starts and the Session ID appears on the page.

**Problem:** The Co-browse session does not start.

### Possible causes of problem:

1. It could be an intranet problem if the Master page is viewed on Internet Explorer (IE) 10 or IE 11. For details, see [Troubleshooting the intranet problem in IE10 and IE11](#)
2. Co-browse Server is not responding or not working. Check the Co-browse Server debug log.
3. The network has a problem.

### Next Step

- [Joining the Slave to the Co-browse Session](#)

### Joining the Slave to the Co-browse Session

**Prerequisites:** There are no problems when [Opening the Slave Page](#) and [Starting a Co-browse Session](#).

#### Start of procedure

1. Copy the Session ID from the Master page and paste it into the edit box on the Slave page.
2. Join the session.

#### End of procedure

**Successful result:** The Slave user successfully joins the session.

**Problem:** The Slave user cannot join the session.

#### Possible causes of problem:

1. Co-browse Server is not responding or not working. Check the Co-browse Server debug log.
2. The network has a problem.

**Problem:** The Slave user cannot join the session and sees the message, Session ID is invalid or has expired.

#### Possible causes of problem:

1. The **slave.html** and **data-gcb-url** addresses (see [Instrumentation](#)) are different. Make sure the addresses are correct and correspond to each other.
2. You have a case where configuration options in the cluster section are not specified. There might be a difference between the **slave.html** address, **data-gcb-url** address and **<host>:<port>** in the Co-browse Server configuration. Make sure each is correct and corresponds to each other.
3. You have a case where configuration options from the cluster section are specified but incorrect. Make sure the options are correct and correspond to the **data-gcb-url** and **slave.html** addresses.

#### Next Step

- [Testing Co-browse With Chat](#)

## Testing Co-browse With Chat

### Required Components

The following components are the minimum required, in addition to the [components listed above](#), to test Co-browse with chat: The following components are the minimum required, in addition to the [components listed above](#), to test Co-browse with chat:

- Stat Server
- Universal Routing Server
- Interaction Server
- Contact Server
- Chat Server
- Interaction Workspace (IWS)/Workspace Desktop Edition (WDE)
- Co-browse Plug-in for IWS or WDE
- Chat strategy activated on necessary queue

See compatible versions in [Related Components](#).

You must also have at least one agent that can log in and go ready for Chat using Interaction Workspace or Workspace Desktop Edition.

## Preparing for Testing

### Prerequisites

- Genesys Framework is running.
- Both Universal Contact Server and Interaction Server have connections to Stat Server.
- Both Stat Server, Universal Routing Server, and Chat Server have connections to Interaction Server.
- Interaction Server has a connection to Chat Server through the ESP port.
- Co-browse Server is installed and fully configured for working with Chat. See [Installation Procedures](#).
- Interaction Workspace is configured to work with Co-browse Server. See [Installing the Plug-in for Interaction Workspace](#).
- You have installed Internet Explorer 9 or above.

### Start of Procedure

1. Start the required servers.
2. After each server starts, check its trace log for errors.
3. In Interaction Workspace, Log in as an agent and go ready for Chat.

### End of Procedure

## Testing Chat with Co-browse

### Initiating a Chat Session

**Prerequisites:** You have successfully completed all previous test procedures.

### Start of Procedure

1. Confirm that an agent is logged in and ready for Chat in Interaction Workspace.
2. On the Master page, click Live Chat.

### End of Procedure

**Successful result:** The Chat Widget opens and the agent receives the Chat interaction.

**Problem:** The agent does not receive the Chat interaction.

### Possible Causes of the Problem:

1. The Chat inbound strategy is not activated or activated on an improper Interaction Server or Universal Routing Server.
2. Stat Server is not working properly. See the Stat Server debug log.
3. Universal Routing Server is not working properly. Universal Routing Server may not be processing your Chat inbound strategy or your Chat inbound strategy is not working as expected. See the Universal Routing Server debug log.
4. The necessary eServices components do not work properly. See the Chat Server, Interaction Server, and Universal Contact Server debug logs.
5. The agent cannot initiate a Chat session. Verify that the proper Capacity Rule is assigned to the agent.
6. Interaction Workspace cannot handle Chat interactions. See the Interaction Workspace debug log.
7. The network has a problem.

### Next Step

- [Starting the Co-browse Session](#)

### Starting the Co-browse Session

**Prerequisites:** You have successfully completed [Initiating a Chat Session](#).

### Start of Procedure

1. As an agent, accept the Chat interaction.
2. Open the CO-BROWSE tab of the Chat interaction.
3. From the Master page, click Co-browsing.

### End of Procedure

### Successful Results:

- The Co-browse session starts.
- The Session ID appears on the Master page.
- The Co-browse page automatically opens in the CO-BROWSE tab for the agent.

**Problem 1:** The Co-browse session does not start. The Session ID does not appear on the Master page.



### Possible causes of problem 1:

1. It could be an intranet problem if the Master page is viewed in IE10 or IE11. For details, see [Troubleshooting the intranet problem in IE10 and IE11](#).
2. Co-browse Server is not responding. See the Co-browse server debug log.
3. The network has a problem.

**Problem 2:** The Co-browse session starts but the Co-browse page does not automatically open for the agent.

### Possible causes of problem 2:

1. Interaction Workspace is incorrectly configured to work with Co-browse Server. You must configure Interaction Workspace's relationship with Co-browse Server:
  - For a standalone Co-browse Server, verify that Interaction Workspace's Connection List contains the Co-browser Server application. See [Installing the Plug-in for Interactive Workspace](#).
  - For a cluster of Co-browse Servers, verify the options in the cobrowse section of the Interaction Workspace Application. See [Configure the Co-browse Plug-in for Interaction Workspace](#) for details. Also, see the Interaction Workspace debug log.
2. Co-browse Server is not responding. See the Co-browse Server debug log.
3. The network has a problem.
4. If it is not clear what the problem is, enable browser debug logging on the Interaction Workspace log, end the chat, and try to start a new chat with co-browsing again. In this case, the browser logs will be stored in the Interaction Workspace log, starting with the word BROWSER. Open the log and try to investigate the problem.

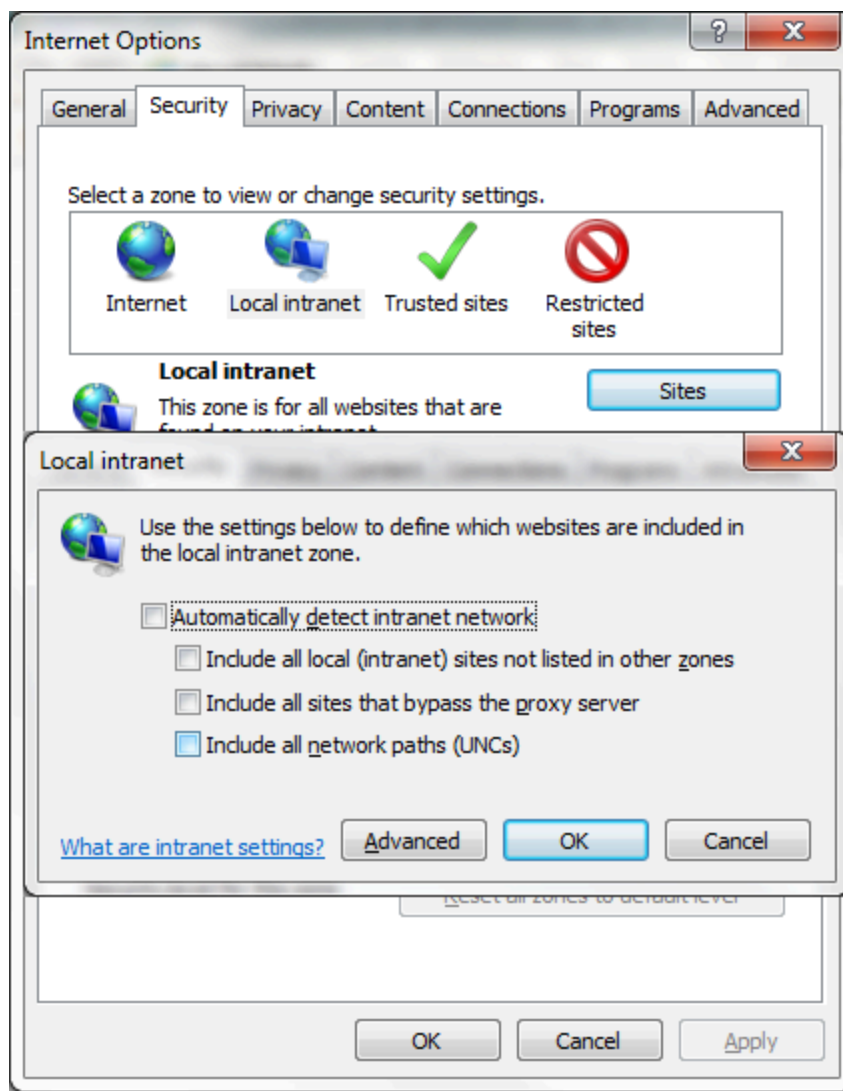
### Important

To enable the debug log in the Slave browser, set the `useBrowserLogging` option in the cobrowse section of the Interaction Workspace application to `true`.

## Troubleshooting the Intranet Problem in IE10 and IE11

IE10 and IE11 do not allow WebSockets on local domains. Internet Explorer uses a built-in algorithm to determine if the domain is local and falls under IE's Local intranet security zone rules. This algorithm is affected by several factors, one of which is the browser's proxy settings. If your domain is listed as "excluded" from proxying, then IE treats your domain as local and does not allow WebSockets to be opened.

To overcome this, you can disable IE's intranet network settings. Go to Tools > Internet Options > Security > Local intranet > Sites and deselect each checkbox:



Disable the intranet network settings.

If you reach Co-browse Server by an internal IP address (such as 192.168.XX.XX), you can overcome the problem by adding a "fake" domain in the hosts file. For example:

```
192.0.2.10    cobrowse.com
```

Next, modify your website instrumentation to use the "fake" domain (in this case, cobrowse.com) for the URL of your Co-browse application.

```
<script>(function(co, b, r, o, w, s, e) {  
  e = co.getElementsByTagName(b)[0];  
  if (co.getElementById(r)) return;  
  s = co.createElement(b); s.id = r; s.src = o;  
  s.setAttribute('data-gcb-url', w);  
  e.parentNode.insertBefore(s, e);  
})(document, 'script', 'gcb-js',  
'http://192.0.2.10:8700/cobrowse/js/masterApp-build.js',  
'http://cobrowse.com:8700/cobrowse')
```

```
);</script>
```

### Important

This problem is unlikely to happen in production environments because Co-browse Server is in Internet Explorer's Internet zone for end users. It may occur when testing the Co-browse solution if Co-browse is deployed in a local network and used exclusively within a company.

## Troubleshooting the "proxy instrumentation problem"

### Prerequisites

- Co-browse instrumentation is done via the Co-browse proxy (most likely to happen in a development or demo environments - it is impossible in production).

### Symptoms

- The website's JavaScript fails completely or partially. This might lead to different problems — the most likely is that some areas of the site are unresponsive or do not render at all (most likely the dynamic areas, such as tabs, accordions, submenus, and so on).
- Errors in the browser console (or error alerts in older versions of Internet Explorer).

### Troubleshooting

1. Open developer tools and examine console logs for errors that happen outside of `gcb.min.js`.
2. Remove instrumentation, reload the page with `Ctrl+F5` (to clean the cache), and see if the same errors are still there.
3. If errors are there with and without Co-browse instrumentation, it is not a proxy issue.
4. If errors are there only with Co-browse instrumentation, it is probably a proxy issue.

### Root cause

The Co-browse proxy works by examining all requests made to the website and replacing a certain sequence of characters with Co-browse instrumentation *AND* this sequence of characters. For example, the following:

```
....  
</head>  
<body>  
....
```

becomes:

```
....  
<COBROWSE_INSTRUMENTATION>  
</head>  
<body>  
....
```

after the `</head>` sequence of characters is replaced by the proxy.

However, if any of the site's JavaScript files contains this sequence, it is *ALSO* "instrumented" and will most likely be broken.

### Treatment

1. Find the sequence of characters that appears **ONLY** in the website's HTML code and does not appear in any of its JS files.
2. Modify the proxy's `map.xml` file to use the new character sequence. For example it may be:

```
<map replace="%s </body>" domains=...
```

or

```
<map replace="%s <meta charset" domains=...
```

#### Important

All special characters in the `replace` attribute should be converted to **HTML entities**.

3. Restart the proxy.

## Troubleshooting CSS Synchronization

If some or all of the content of your website is not properly rendered on the slave (agent) side, it is most likely a CSS synchronization problem.

#### Warning

If your website is using conditional comments for Internet Explorer, also see **Limitations on IE Conditional Comments**.

First, try different **CSS synchronization** settings. The possible settings are server (default setting), browser, or both.

The most reliable CSS synchronization mode is server but there may be edge cases where browser mode or both modes together produce better results.

If the problem persists, try the following:

1. Server mode works by proxying your site's CSS. Co-browse Server makes an HTTP request to get your site's CSS. Make sure requests from Co-browse Server are not blocked.
2. When using browser mode and sometimes with server mode, requests for your site's CSS are made from the slave browser. Make sure that requests from the slave browser are not blocked.

3. Server mode sometimes stalls on invalid CSS. When it does, a message appears in the Co-browse server logs.

Example:

```
19:16:34,454 [ WARN] LoggingCSSParseErrorHandler - [1:30]-[1:43] Encountered text ' {'  
corresponding to token <LBACE>. Skipped until token }. Was expecting one of: <S>, ";
```

Depending on the kind of error, the parser will do one of the following:

- Skip the stylesheet completely and let the slave browser load the stylesheet as is. CSS :hover effects will not be synchronized for this stylesheet.
- Supply a corrupt version of the stylesheet to the slave browser. The slave user may end up with something visually different than the master user.

To avoid CSS synchronization issues, you should **validate your CSS using the freely available CSS Lint Tool**. Use the tool to avoid CSS errors. CSS with warnings from the tool is usable.

In cases where CSS synchronization issues continue, you can manually fix slave representation by providing additional CSS using the `cssPatchUrl` **server configuration option**.

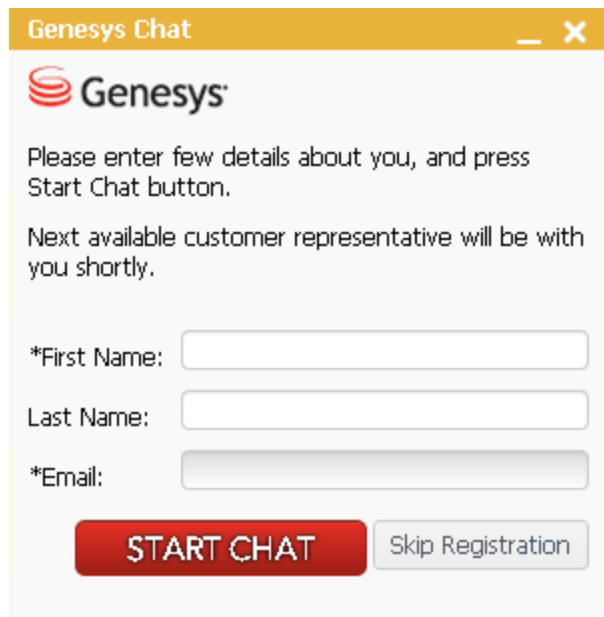
## Troubleshooting Chat Widget Rendering Issues

### Prerequisites:

- Built in chat widget uses **embedded** mode.

### Symptoms:

Chat widget renders incorrectly or has unexpected styling. For example:



The screenshot shows a Genesys Chat widget window. At the top is a yellow header bar with the text "Genesys Chat" and a close button. Below the header is the Genesys logo. The main content area contains the following text: "Please enter few details about you, and press Start Chat button." and "Next available customer representative will be with you shortly." Below this text are three input fields: "\*First Name:", "Last Name:", and "\*Email:". At the bottom of the form are two buttons: a red "START CHAT" button and a grey "Skip Registration" button.

In the figure above, the Start Chat button and Email field do not render as expected.

#### **Possible Causes of the Problem:**

In embedded mode, chat HTML and CSS is added directly to your site's page. Your site's CSS will affect Chat Widget styling. Generally, all the chat widget's styling is sandboxed, but some of your site's CSS may leak into the widget and create unwanted styling effects.

#### **Treatment:**

Find out which CSS rules create unwanted style effects in the chat widget and create a CSS patch that overrides these rules. You can use your browser's developer tools to find the CSS rules. For more information on CSS customization of the chat widget, see the **CSS-based** tab in the [Customizing the User Interface](#) section.