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

12/14/2025

Testing and Troubleshooting the Co-browse Solution

Contents

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

Testing Co-browse Without Chat

Required Components

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

- Local Control Agent
- Configuration 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.

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 Customer side page.

End of Procedure

Successful result: The `Co-browsing` button is 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 Customer side 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 Customer side browser console log in your instrumentation script. For details, see Enabling console logs.

Testing Co-browse Session

Opening the Agent Side Page

Prerequisites: You have successfully completed Preparing for Testing.

Start of Procedure

1. Open the Agent side 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 Agent side 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 Agent side are incorrectly specified in the `slave` section of the Co-

browse Server application. For details about localization settings, see Localizing the Agent side UI.

3. The network has a problem.
4. If it is not clear what the problem is, enable debug logging on the Agent side 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 Agent side browser, insert debug=1 into the Agent side 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 Agent Side Page.

Start of procedure

1. In the Customer side 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 Customer side page is viewed on Internet Explorer (IE) 11. For details, see Troubleshooting the intranet problem in IE 11
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 Agent side to the Co-browse Session

Joining the Agent side to the Co-browse Session

Prerequisites: There are no problems when Opening the Agent side Page and Starting a Co-browse Session.

Start of procedure

1. Copy the Session ID from the Customer side page and paste it into the edit box on the Agent side page.

2. Join the session.

End of procedure

Successful result: The Agent side user successfully joins the session.

Problem: The Agent side user cannot join the session.

Possible causes of problem:

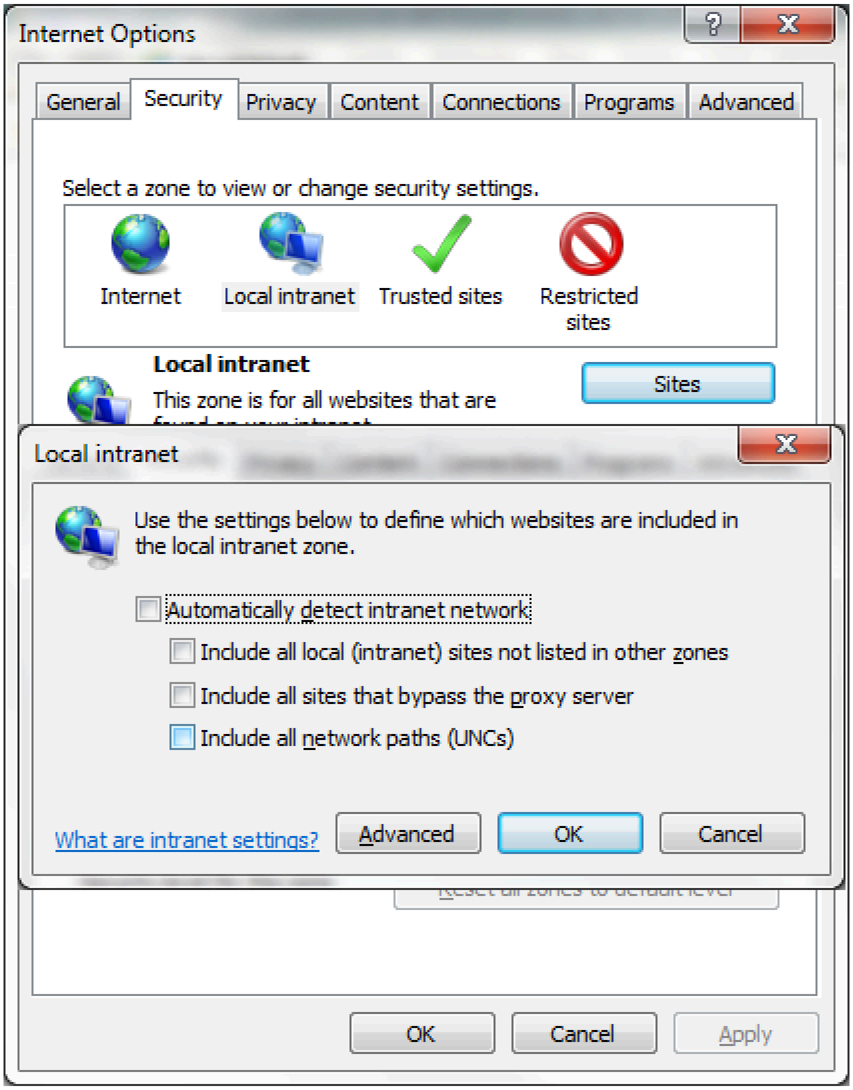
1. Co-browse Server is not responding or not working. Check the Co-browse Server debug log.
2. The network has a problem.

Troubleshooting the Intranet Problem in IE 11

> **Important**
>
> Starting from 9.0.014.xx, support for Internet Explorer 11 is deprecated and will be dropped in the future releases.

IE 11 does 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:



Select a zone to view or change security settings.

This zone is for all websites that are

Local intranet

Use the settings below to define which websites are included in the local intranet zone.

☐ Automatically detect intranet network

☐ Include all local (intranet) sites not listed in other zones

☐ Include all sites that bypass the proxy server

☐ Include all network paths (UNCs)

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(d, s, id, o) {
  var fs = d.getElementsByTagName(s)[0], e;
  if (d.getElementById(id)) return;
  e = d.createElement(s); e.id = id; e.src = o.src;
  e.setAttribute('data-gcb-url', o.cbUrl);
  fs.parentNode.insertBefore(e, fs);
})(document, 'script', 'genesys-js', {
  src: "http://192.0.2.10:8700/cobrowse/js/gcb.min.js",
  cbUrl: "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 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 Exporer).

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 Agent side, it is most likely a CSS synchronization problem.

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 Agent side browser. Make sure that requests from the Agent side 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 <LBRACE>. 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 Agent side 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 Agent side browser. The Agent side user may end up with something visually different than the Customer side 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 Agent side representation by providing additional CSS using the `cssPatchUrl` server configuration option.