



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

CSS Synchronization

Contents

- 1 CSS Synchronization
 - 1.1 Why does Co-browse need to synchronize CSS?
 - 1.2 CSS Synchronization Architecture
 - 1.3 Configuring CSS Synchronization
 - 1.4 Synchronizing CSS Through a Secure Zone
 - 1.5 Troubleshooting CSS Synchronization

CSS Synchronization

This article gives an overview of how Genesys Co-browse synchronizes CSS between the customer and agent browsers.

Why does Co-browse need to synchronize CSS?

When a customer and agent are in a Co-browsing session, Co-browse tracks DOM-based and CSS-based changes in each browser and replicates changes from one browser to the other. All DOM-based changes pass from one browser to the other through the Co-browse server. Examples of DOM-based changes include creating new elements in the web page and adding or removing attributes from an element. To replicate CSS-based changes in the browser, Co-browse must make sure both browsers use the same CSS rules even when the customer and agent use different browsers. CSS-based changes can include drop-down menus and other hover events.

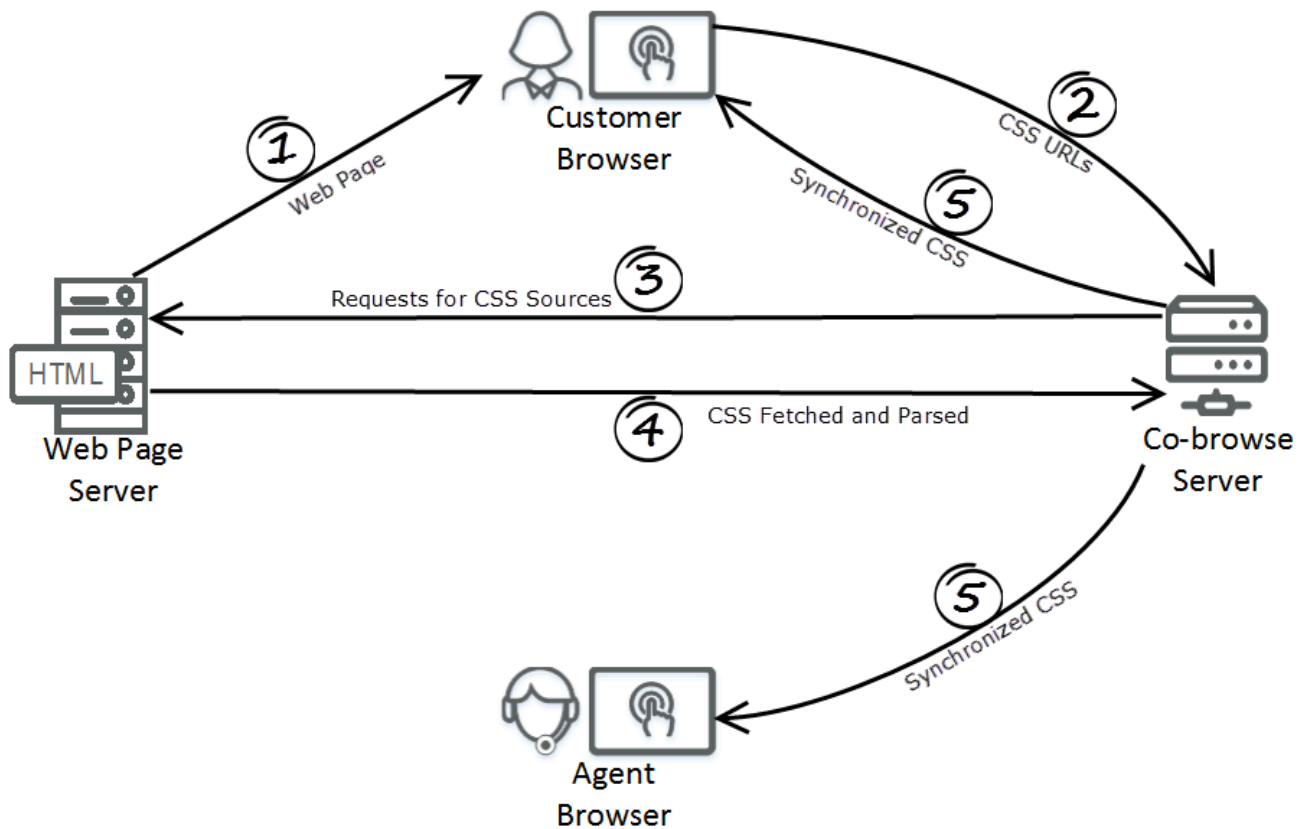
Synchronizing Browser Events

Some CSS-based changes depend on browser events that the server can not push from one browser to the other. By synchronizing CSS between the customer and agent, Co-browse server can replicate browser events it can not push. For example, the customer and agent browsers each have their own mouseover event that fires when the mouse pointer hovers over a web page element. Without CSS synchronization, CSS-based changes that fire based on the mouse pointer will show in one browser but not the other. To synchronize hover events, Co-browse server parses the web page CSS and adds a DOM-based pseudo-hover that fires the hover event on both browsers.

CSS Synchronization Architecture

The following diagram describes how Co-browse synchronizes CSS between the customer browser and the agent browser:

1. The customer opens a website and starts a Co-browsing session with an agent.
2. Co-browse server reads the CSS sources from the customer's view of the web page.
3. Co-browse server fetches all the required CSS stylesheets.
4. Co-browse parses the CSS stylesheets and adds additional Co-browse specific CSS.
5. Co-browse sends the synchronized CSS stylesheet to the agent and customer browsers.



Configuring CSS Synchronization

The **CSS** option of the **JavaScript Configuration API** manages CSS synchronization. Genesys recommends using the server strategy and the **CSS** option is set to server by default. In some edge cases, changing the **CSS** option may produce better CSS synchronization results. For more about improving CSS synchronization, see the **CSS synchronization section of the troubleshooting page**.

Synchronizing CSS Through a Secure Zone

If you deploy Co-browse into a secure zone like a DMZ or local intranet, you must make sure the Co-browse server can still access your public web page by configuring a **Forward Proxy**. Otherwise, Co-browse will not be able to synchronize CSS and the agent side may not properly render.

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. See the [CSS synchronization section of the troubleshooting page](#).