



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

12/19/2025

# Customer Q&A

## Contents

- [1 Customer Q&A](#)

This page answers some of the most common questions we receive about Genesys Co-browse.

**Q: Which emerging technologies and industry standards related to the Co-browse product are supported and will be evolved?**

A: The key technologies targeted are HTML5 (including SVG), JavaScript and CSS.

**Q: What is the Co-browse solution's architecture in relation to hardware and software?**

A: See [Co-browse Architecture](#).

**Q: Can you describe any high availability and redundancy solutions?**

A: In the current release, common resources not pertaining to a certain live co-browse session are served by any node in the cluster. Each co-browse session is hosted on a single server in the cluster. Future releases will support server failover functionality for Co-browse sessions where live sessions will be almost transparently transferred to another server in the cluster. Web chat sessions are already transparently migrated to another Co-browse server. Co-browse historical data redundancy is achieved through the Cassandra cluster.

**Important**

Starting in 9.0.005.15, Cassandra support is deprecated in Genesys Co-browse and **Redis** is the default database for new customers. Support for Cassandra will be discontinued in a later release.

**Q: What is the highest amount of simultaneous users successfully handled?**

A: Thanks to horizontal solution scalability, the highest amount of simultaneous users is limited only by the server hardware involved.

**Q: Has the site traffic been verified by any third-party?**

A: Not applicable. HTTP communication with Co-browse server is tested with ZAP proxy, [https://www.owasp.org/index.php/OWASP\\_Zed\\_Attack\\_Proxy\\_Project](https://www.owasp.org/index.php/OWASP_Zed_Attack_Proxy_Project).

**Q: Can you describe Alert, Monitoring and control options and functionality (Administrative control and notifications for server and edge site errors)?**

A: Co-browse server monitoring is performed through standard Genesys platform tools (messages are displayed in Genesys Administrator and the Solution Control Interface). The same goes for alerts.

**Q: Can you describe the technologies that the application is written in?**

A: Java (Jetty 9.x as a container), JavaScript, HTML5 (including Mutation Observers and Web Sockets), CSS, and CometD.

**Q: Is the company's core technology developed by internal engineering staff, or is it outsourced to partner developers?**

A: Internal engineering develops the core technology.

**Q: How is quality control ensured?**

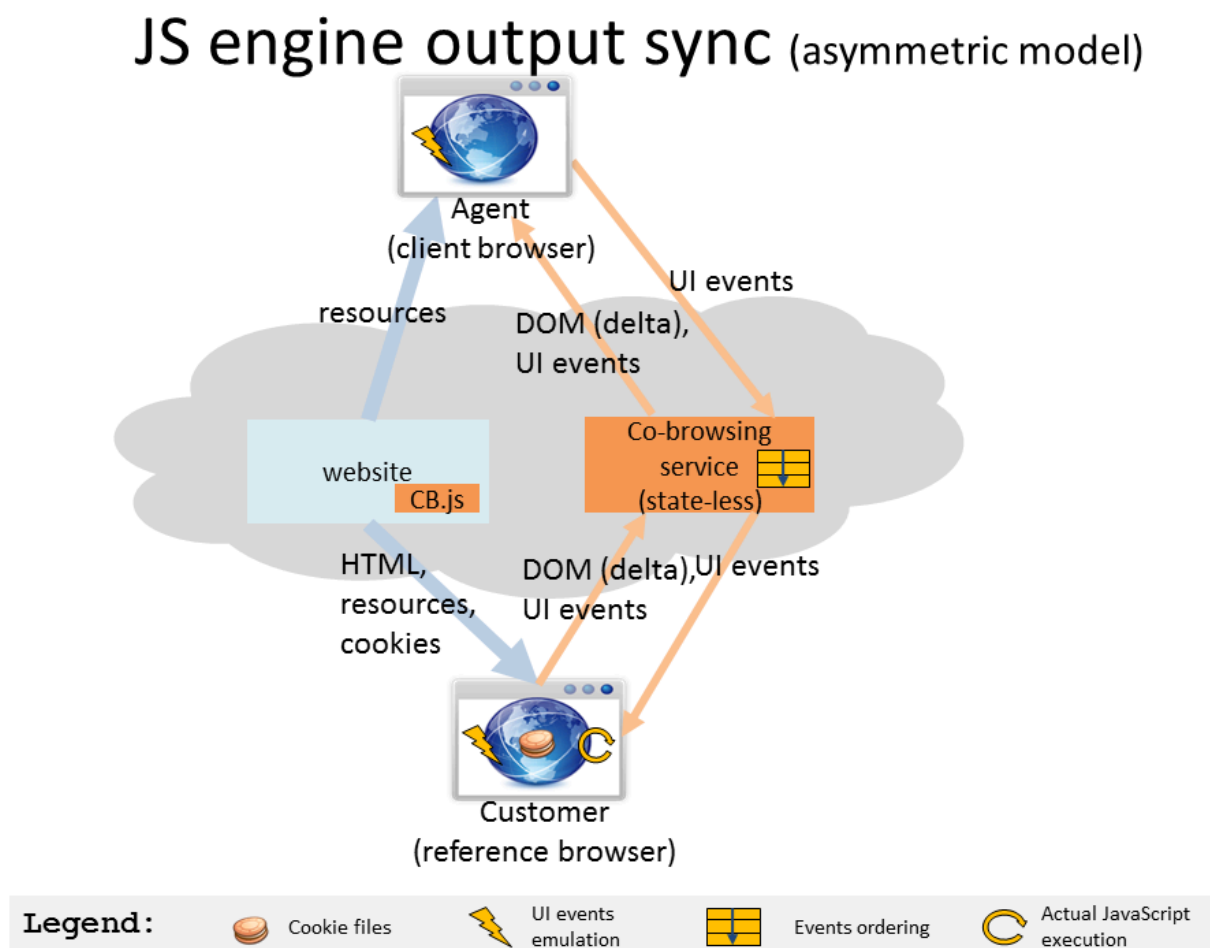
A: Daily builds are verified by Quality Assurance. The main use cases are automated.

**Q: Has the technology been recognized by third-party endorsements?**

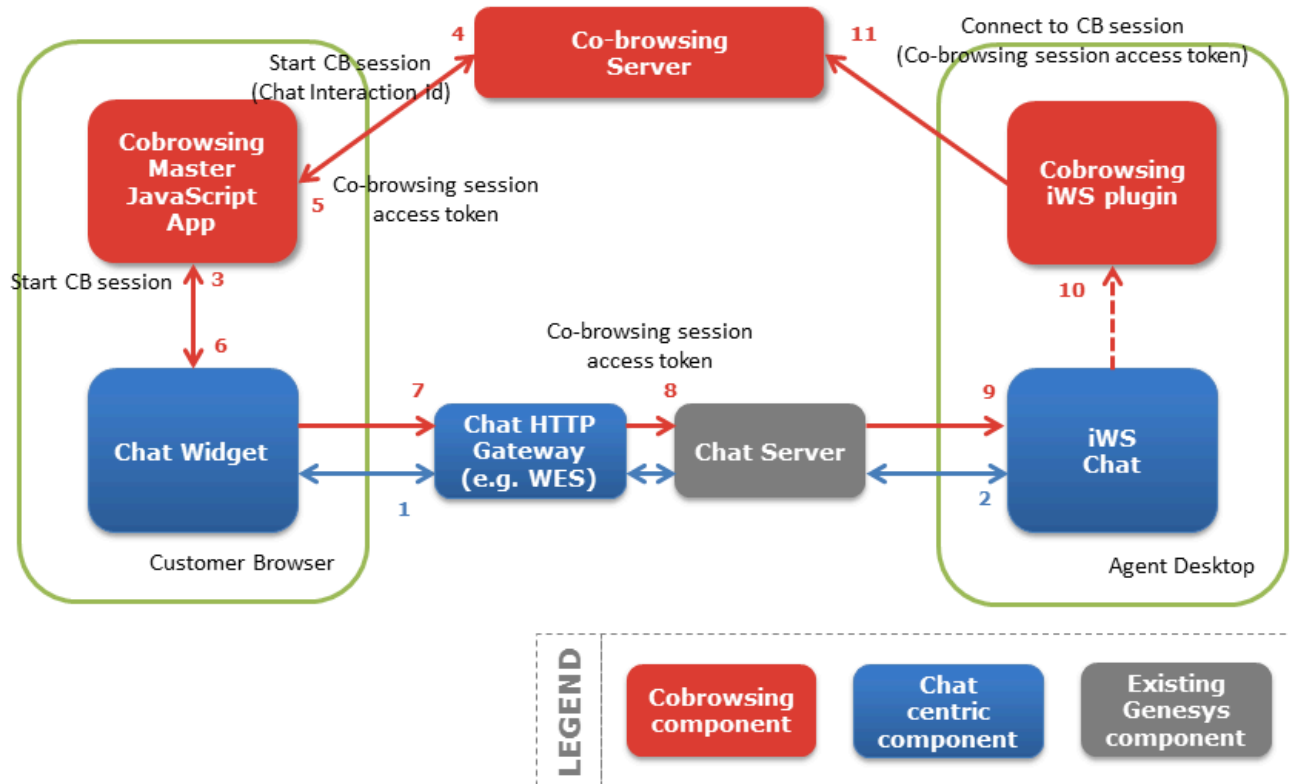
A: Similar principles are implemented by competitors.

**Q: What are the basic principles of the system's inner workings?**

A: Conceptual diagram:



The diagram below shows chat and Co-browse integration. Co-browse server incorporates Web Chat Gateway function as well.



**Q: Can the web page be easily branded with company colors and logos?**

A: Yes.

**Q: Explain the process of embedding links on web pages as well as any other user interfaces. How much integration is required?**

A: Each cobrowsable website page must include a script. For more information, see [Website Instrumentation](#). The following is a basic example of the required script:

```
<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: "<COBROWSE_SERVER_URL>/gcb.min.js",
  cbUrl: "<COBROWSE_SERVER_URL>/cobrowse"
});</script>
```

Default functionality can be customized through JavaScript based configuration or the Co-browse JavaScript API. Co-browse website functionality can be roughly split into two parts:

- Co-browse session initiation UI
- The Co-browse session itself

The UI for each can be replaced or customized using CSS, the JavaScript API, or Localization. For more information, see [Customize the Genesys Co-browse UI](#)

**Q: Is the application flexible? Is it easy to add, customize and track new fields?**

A: Yes.

**Q: What are the desktop requirements for the customer and servicing agents using Genesys Co-browse?**

A: Genesys Co-browse requires Workspace Desktop Edition starting with release [8.5.000.30](#).

**Q: What is the minimum bandwidth required for a co-browse session?**

A: The bandwidth will depend on a co-browsed web site's content and the techniques used to present it. Unlike screen sharing solutions, Genesys Co-browse syncs initial HTML page/resources, DOM deltas, and actions so it does not require high bandwidth. With Web Socket (which is supported by modern mobile devices and where ever bandwidth is normally concerned) support, there is not even over-head associated with HTTP requests or responses to Co-browse server. Web Sockets must be supported by the reverse proxy/load balancer infrastructure.

**Q: Do you support secure WebSockets (wss://)?**

A: Genesys Co-browse recommends using WebSockets and supports secure WebSockets. Co-browse uses secure WebSockets (wss://) when the [website instrumentation cbUrl](#) uses https://. If the **cbUrl** uses http:// then Co-browse uses ws:// WebSockets. When using https:// in the **cbUrl**, you do not need any additional configuration to use secure WebSockets.

**Q: Can you co-browse within iframes?**

A: Co-browse supports co-browsing in iframes from the same domain or from the subdomain. Agents can view but not co-browse iframes pointing to a different domain.

The iframe will be visible for agents if the [allowedThirdPartyDomains](#) option is set correctly, however any mouse movements, scrolling, clicking, or data input on the customer side will not be synchronized to the agent side.

For more information, see [co-browsing in iframes](#).