Browser Real User Monitoring

On this page:

Related pages:

- End User Monitoring

Search the Browser RUM topics:

Browser Real-User Monitoring (Browser RUM) allows you to see how your web application is performing from the point of view of a real or synthetic end user. You can answer questions like:

- Which 1st or 3rd-party Ajax or iframe calls are slowing down page load time?
- How does server performance impact end-user experience in aggregate or in individual cases?

You can drill into the data to explore how users experience your application in their web browsers.

Monitor Your Application

Browser RUM offers multiple ways to look at your data in real time. You can:

- **Understand and improve your web page's performance**
  - Know how your pages, Ajax requests (XHR, Fetch API calls), and iframes are performing over time. See The Pages & Ajax Requests View.
  - Gain insight into individual requests, with detailed charts on how your pages, Ajax requests, and iframes load and build in your end user's browsers, with links, if enabled, to reports on server-side performance. See Browser Snapshots.
  - Find your worst performing pages by multiple common metrics. See Top Pages.
- **Reduce errors**
  - Learn which pages are loading with JavaScript errors, and the script file and line number that are creating the problem. See Browser Snapshots.
- **Learn about your users**
  - See how your web users are connecting to your application by device/platform and browser. See Browser App Dashboard.
  - Find out where in the world your web users are and how your application is performing across countries and regions. See Browser App Dashboard.
  - Understand how users are navigating through your website and what actions they are taking. See Browser RUM Sessions.

For more information on using Browser RUM, see Monitor Your Applications with Browser RUM.

How Browser RUM Works

Browser RUM works in the following way:

1. An end user requests the first page from your web application.
2. Your web application executes whatever business logic that the particular page requires.
3. Your web application creates the response page to return to the end user. The response page includes:
    a. application specific information
    b. a copy of a small JavaScript script that knows how to collect relevant performance information about that page. This script is called the JavaScript Agent.
4. The page, with the JavaScript Agent included, is returned to the end user.
5. As the page is being constructed in the browser, the script collects relevant information about the page's performance.
6. At approximately the same time as the `onload` event for the page fires, a copy of a somewhat larger JavaScript file, the JavaScript Agent extension, is downloaded asynchronously by the injected agent.
7. This second script packages the collected performance information and sends it via a web beacon to the EUM Server collector for processing.
8. Working together the two scripts continue to collect and send performance information as the end user navigates through the instrumented pages of your application.

Set Up and Configure Browser RUM

Browser RUM is easy to set up. It is also highly configurable. You can:

- Set up and enable Browser RUM. See Set Up and Access Browser RUM.
- Instrument your application to work with Browser RUM. For more information, see Configure the JavaScript Agent.
- Set up how your information appears in the Controller UI. For more information, see Configure the Controller UI for Browser RUM.

License and Enable Browser Monitoring

Browser Real-User Monitoring requires a separate license and must be enabled before it is available for use.

For information about licensing, including a description of the types of licenses, Lite and Pro, see Browser RUM Licenses.

For information on enabling or disabling Browser RUM, see Enable and Disable Browser RUM.