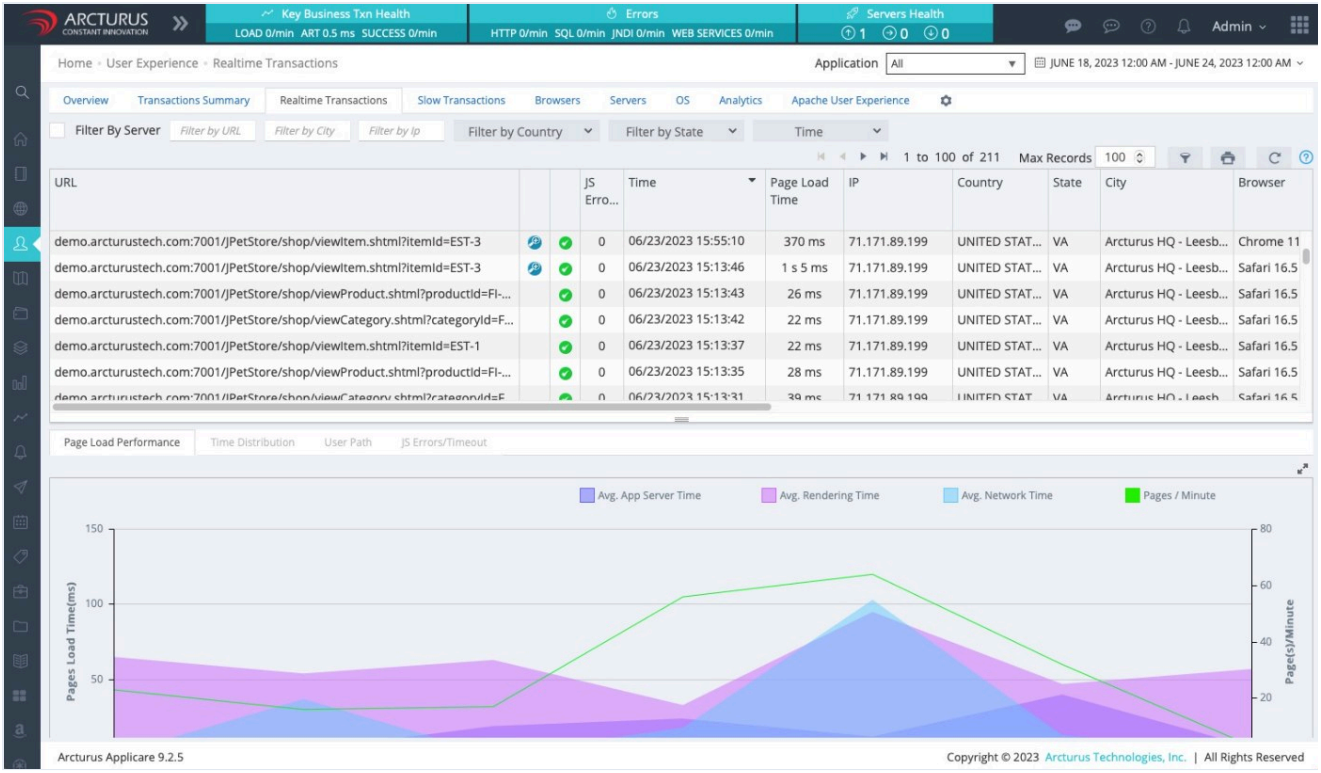
USING APPLICARE › USER EXPERIENCE MONITORING

EUM Real-Time Transaction Monitoring

Applicare 10.0 · Updated September 13, 2024 · **Pending technical review**

The real-time view shows the last n number of individual transaction recorded. Along with performance stats you can view location and browser information of users accessing your application. This view doesn't show any transaction older than a day.



If the Applicare Controller/Collector gathering EUM data is situated a proxy, firewall or load balancer, ensure that these intermediaries forward client requests with the "X-Forwarded-For" header parameter. If this parameter is not included in the HTTP request then Applicare may show the IP address of proxy, firewall or load balancing server in End User Experience Analyzer.

Below are the details on monitored data shown for each transaction

- **Resource Fetch Time:** The duration required for a web browser to retrieve a resource, such as an image, CSS file, or JavaScript file. This includes the time it takes to:
  - **DNS lookup:** The time required to resolve the domain name of the resource.
  - **TCP connection establishment:** The time required to establish a connection to the server hosting the resource.
  - **TLS handshake:** The time required to establish a secure connection with the server, if the resource is hosted over HTTPS.
  - **Request:** The time required to send the request for the resource to the server.

- **Response:** The time it takes to receive a response from the server, including duration required to download the resource is known as Resource Fetch Time. This metric is crucial for evaluating web page performance, as a shorter Resource Fetch Time leads to faster page loading and enhance user experience.

  Several factors can influence Resource Fetch Time, including:

  - **Resource Size:** Larger resources will take longer to fetch.
  - **Internet Connection Speed:** A slower internet connection will lead to a longer Resource Fetch Time.
  - **Browser Performance:** A faster browser can retrieve resources more quickly.
  - **Server Location:** The closer the server is to the user, the shorter the Resource Fetch Time will be.

  Here are some tips for improving Resource Fetch Time:

  - **Use Minified and Compressed Resources:** Minified and compressed resources are smaller, resulting in fast fetch time.
  - **Use a CDN:** A CDN (Content Delivery Network) can deliver resources from servers that are closer to the user, which can improve Resource Fetch Time.
  - **Implement Caching:** A caching mechanism can store resources that have already been retrieved, reducing need to fetch them again.

- **Document Download Time:** The duration required to download a document from a web server to a user's device. This includes the time it takes to:

  - **DNS Lookup:** The time needed to resolve the domain name of the web server.
  - **TCP Connection Establishment:** The time required to establish a connection to the web server.
  - **TLS Handshake:** The time required to establish a secure connection with the web server, if the request is made over HTTPS.
  - **Request:** The time required to send the request for the document to the web server.
  - **Response:** The time needed to receive the response from the web server, including the time required to download the document.

- **Document Processing Time:** The time it takes for the browser to finish parsing the HTML code of a web page and construct the Document Object Model (DOM). The DOM is a hierarchical representation of the page's elements, which can be manipulated by JavaScript.

  Document processing time can be affected by a number of factors, including:

  - **Size of the HTML Code:** Larger HTML files will take longer to parse.
  - **Complexity of the HTML Code:** More complex HTML files will take longer to parse.
  - **Browser Performance:** Faster browsers will be able to parse HTML code more efficiently.
  - **The presence of JavaScript:** JavaScript files can increase the document processing time.

  Here are some tips for improving document processing time:

  - **Use Minified and Compressed HTML Code:** Minified and compressed HTML code is smaller, which means it will take less time to parse.

  - **Use a CDN:** A CDN (Content Delivery Network) can deliver HTML code from servers that are closer to the user, which can improve document processing time.
  - **Implement Caching:** A caching mechanism can store HTML code that has already been parsed, which can reduce the need to parse it again.
  - **Optimize JavaScript files:** Optimizing JavaScript files can improve their performance and reduce the document processing time.

- **Time to First Byte (TTFB):** A metric that measures the time it takes for a web browser to receive the first byte of a response from a web server. This includes the time it takes to:
  - **DNS lookup:** The time it takes to resolve the domain name of the web server.
  - **TCP connection establishment:** The time it takes to establish a connection to the web server.
  - **TLS handshake:** The time it takes to establish a secure connection to the web server, if the request is made over HTTPS.
  - **Request:** The time it takes to send the request for the resource to the web server.
- **DOM Ready Time:** The time it takes for a web browser to finish parsing the HTML code of a web page and construct the Document Object Model (DOM). The DOM is a hierarchical representation of the page's elements, which can be manipulated by Javascript.

  The DOM Ready Time event is fired when the DOM is ready, but the page is not necessarily fully loaded yet. This means that images, CSS, and other resources may still be loading. However, the DOM Ready Time event is a good indication of how long it takes for a web page to become interactive.

  There are a number of factors that can affect DOM Ready Time, including the size and complexity of the HTML code, the speed of the user's internet connection, and the performance of the browser.

  DOM Ready Time is an important metric for measuring the performance of a web page. A short DOM Ready Time means that the page will be interactive more quickly, which can improve the user experience.

  Here are some tips for improving DOM Ready Time:
  - Use minified and compressed HTML, CSS, and JavaScript files.
  - Avoid using too many images or other large resources.
  - Use a CDN to deliver static resources.
  - Use a caching mechanism to store frequently accessed resources.
- **Network Time:** The time it takes for a web browser to fetch all of the resources that are needed to load a web page. This includes the time it takes to:
  - **DNS lookup:** The time it takes to resolve the domain name of the resources.
  - **TCP Connection Establishment:** The time it takes to establish a connection to the servers that host the resources.
  - **TLS Handshake:** The time required to establish a secure connection to the servers, if the resources are hosted over HTTPS.
  - **Request:** The time it takes to send the requests for the resources to the servers.
  - **Response:** The time it takes to receive the responses from the servers, including the time it takes to download the resources.

- **Rendering Time:** The time it takes for a web browser to render a web page. This includes the time it takes to:
  - **Parsing the HTML Code:** The time it takes to parse the HTML code of the web page and construct the DOM.
  - **Loading CSS and JavaScript Files:** The time it takes to load the CSS and JavaScript files that are needed to style and interact with the web page.
  - **Rendering Images and Other Resources:** The time required to render the images and other resources that are used on the web page.

Several factors can contribute to high DOM ready time despite a low document processing time.

- **Large and Complex HTML Code:** A large and complex HTML code can take longer to parse, even if it does not contain any JavaScript. This is because the browser has to process more information to create the DOM.
- **Heavy Use of Third-Party Resources:** Third-party resources, such as images, CSS, and JavaScript files, can also add to the DOM ready time. This is because the browser has to wait for these resources to load before it can create the DOM.
- **Slow Web Server:** If the web server is slow, it can take longer for the browser to download the HTML code and third-party resources. This can also increase the DOM ready time.

If you notice a high DOM ready but a low document processing time, consider optimizing your HTML code to reduce its size and complexity, making it easier and quick to parse.

Monitoring with Applicare