

Browser Monitoring

release date. 2026. 09. 11.



Table Of Contents

• Browser Monitoring	4
◦ Getting Started.....	13
◦ Installation	
■ Supported Environments	
■ Browser Compatibility.....	15
■ Preliminary configuration	17
■ Application of the browser agent.....	19
◦ Configuration	
■ Setting the actual user ID.....	27
■ APM Integration Settings	29
■ Collecting custom events	33
■ Collecting custom logs	38
◦ Dashboard	
■ Browser Monitoring Dashboard.....	47
■ Browser Error.....	62
■ Page Load.....	69
■ Resources	81
■ AJAX.....	84
■ Setting a widget	88
■ Flex Board	97
■ Using Flex board	100
■ Managing the widgets	110
◦ Analysis	
■ User Action Hitmap.....	115

■ AJAX HITMAP.....	128
■ Error tracing.....	140
■ Searching the user session log.....	148
■ Session Replay	156
◦ Metrics.....	170
■ Browser metrics	172
■ Metrics Chart	184
■ Metrics Search	187
◦ Statistics	
■ Page load statistics	189
■ IP Load Statistics	202
■ Resource Load Statistics.....	213
■ Error Load Statistics.....	223
◦ Report.....	227
◦ Alert	232
■ Event Configuration	0
■ Metrics	238
■ Composite Metrics.....	260
■ Hitmap Patterns.....	267
■ Event Notification Settings.....	273
■ Event Notification Format.....	295
■ Event History and Monitoring	303
■ Real-time Notifications.....	311
■ System Events	314
◦ Laboratory	317
◦ Browser agent collection data	318

Browser Monitoring

Browser Monitoring collects and analyzes various performance metrics such as page load time, server response time, resource loading time, and browser errors for browser applications. Through the collected and analyzed data, bottlenecks or performance issues for browser applications can be identified and optimized. It also provides insight into performance differences across browsers, providing you with the data to have the best user experience.

Why do you need Browser Monitoring?

- **Optimizing the response time for browser applications in a complex server environment in the era of Cloud and MSA**

With the advent of Cloud and MSA, the server environment has become more complex. In this complex environment, it is difficult to determine the cause of a slow browser application response time based on the performance metrics of each individual server. Accordingly, the factors including the network status between the server and clients and the browser loading time must be taken into account. In addition, the related factors such as browser's request data size and request cycle that can affect the response time should be considered.

Browser Monitoring is required to optimize the response times and user experiences of real users.

- **Quick response to unpredictable errors that occur in real user environments**

Due to the introduction of CI/CD, where development, modification, and distribution are made frequently, it is difficult to prevent errors that occur to actual users by only pre-release testing. Various errors may occur depending on the user device, browser type, and network conditions while using web pages.

Browser Monitoring should monitor errors that occur to actual users and apply technical approaches to prevent or minimize them.

- **Provision of statistical data to provide optimized user experiences in various browsers and devices**

As browser types and devices become more diverse, it is required to monitor errors encountered by real users. For example, you can see the errors that occur only in a specific browser or device. This allows us to improve user experiences and provide better services.

- **Analyzing and optimizing the browser application performance in the SPA environment**

In a web application, Single Page Application (SPA), web pages have been implemented primarily in the browser. In the past, all elements of a web page were rendered on the server, but now the browser renders pages and fetches only the required data from the server for operation. With all this complexity of web applications, you cannot tell the application performance status only by managing the server. Therefore, **Browser Monitoring** should be used to find problems in browser applications and perform optimization.

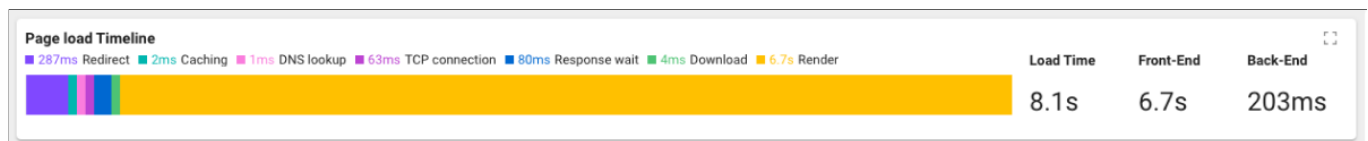
- **Improving the user experience performance through the comprehensive response time measurement and optimization of 1st party and 3rd party resources**

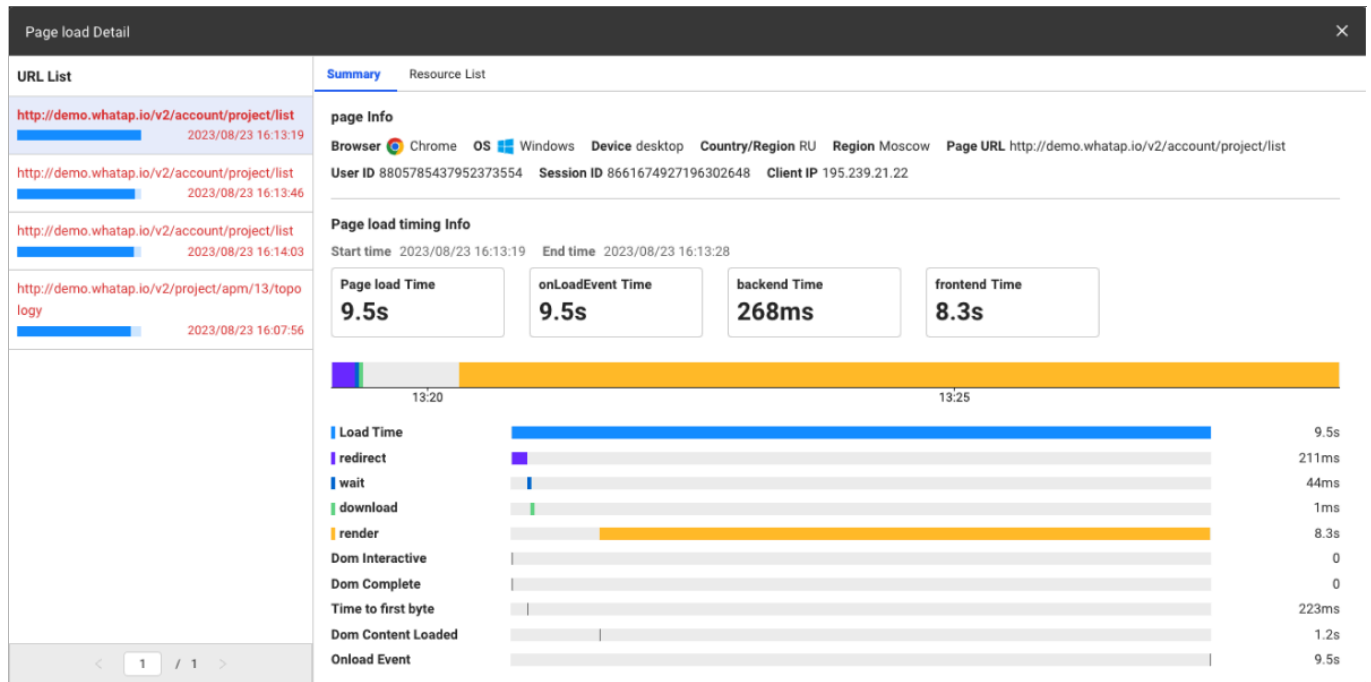
The performance experienced by actual users of any web service should be comprehensively measured with the response times of the 1st party resources operated in house and the 3rd party resources operated externally. The overall response time may be delayed only by the response time delay of the 3rd party resources.

Key features

Browser Monitoring monitors the website operation problems, end-user usability, and end-user environment from the end user's point of view. It provides the following key features.

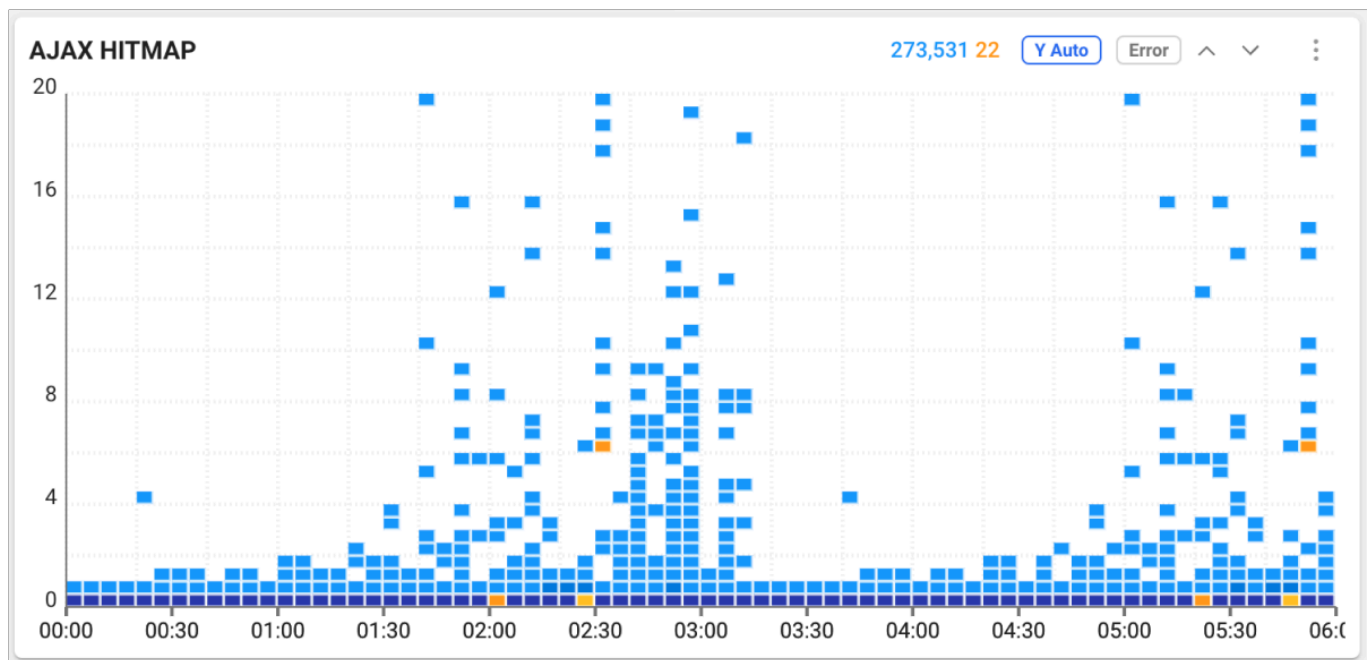
Provision of the page load performance data





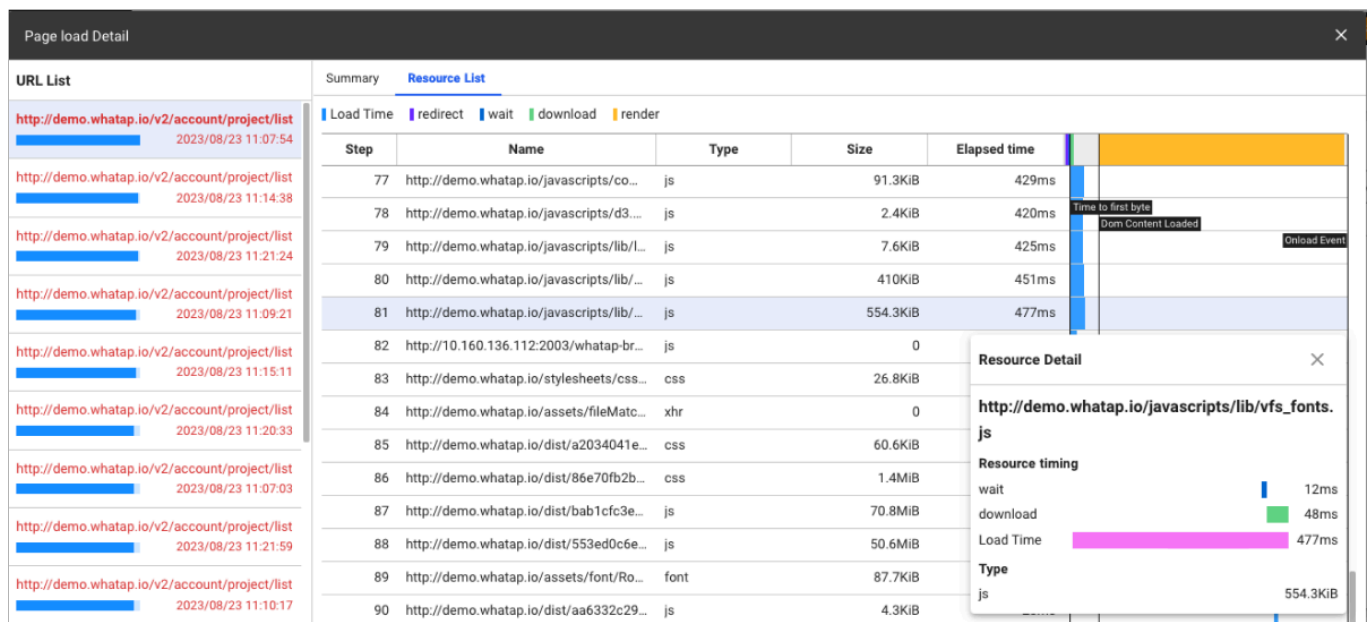
It monitors how quickly the browser loads web pages. Users prefer the web pages that are loaded quickly. Accordingly, the page load performance data allows developers to optimize the web page performance. The page load performance data provides information such as how long it took the browser to load pages, how long the page elements were loaded, and the server response time. This information can help you find the way to improve the performance of the web pages.

Provision of AJAX performance data



It monitors the performance data when exchanging data with the server via AJAX. AJAX is a technology that allows the web pages to exchange data asynchronously with the server, allowing browser applications to be updated without refreshing the pages. **Browser Monitoring** collects various data, including AJAX requests and response times, server response time analysis, and network latency. The collected data is useful to improve the performance of browser applications and improve the user experience.

Provision of the web page resource data



The resources downloaded from the server by the browser significantly affect the page loading performance. The resource list lets you quickly identify the resources that are slow to load or have large files. We can use this information to improve the performance of web pages and the user experience.

Provision of the browser error information

Timestamp	Log
2023-04-20 20:40:19.444	errorType onError pageLocation http://test-client.jtest.ai.s3-website.ap-northeast-2.amazonaws.com/ title All posts RUM Test url none browser Chrome browserVer 111.0.0.0 os macOS osVer 10.15.7 device Desktop host test-client.jtest.ai.s3-website.ap-northeast-2.amazonaws.com UserAgent Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/111.0.0.0 Safari/537.36 message Missing initializer in const declaration stack

Error: SyntaxError: Missing initializer in const declaration

at onClick

in webpack://gatsby-starter-blog/.src/components/layout.js

Line 154

:

Column 17

```

return (
  <div className="global-wrapper" data-is-root-path={isRootPath}>
    <div className="controls">
      <button
        id="script-error"
        type="button"
        onClick={() => {
          const badCode = "const s;"
          eval(badCode)

        }}
      >
        Generate script error
      </button>
      <img className="bad-img" />
    </div>
    <Helmet>
      <script src={withPrefix("script.js")} />
      <script src={withPrefix("script.js")} />
    </Helmet>
  </div>
)

```

at Object.je

in http://test-client.jtest.ai.s3-website.ap-northeast-2.amazonaws.com/framework-661afe2a1ce1c4656c92.js

Line 2

:

Column 17040

at Ke

in http://test-client.jtest.ai.s3-website.ap-northeast-2.amazonaws.com/framework-661afe2a1ce1c4656c92.js

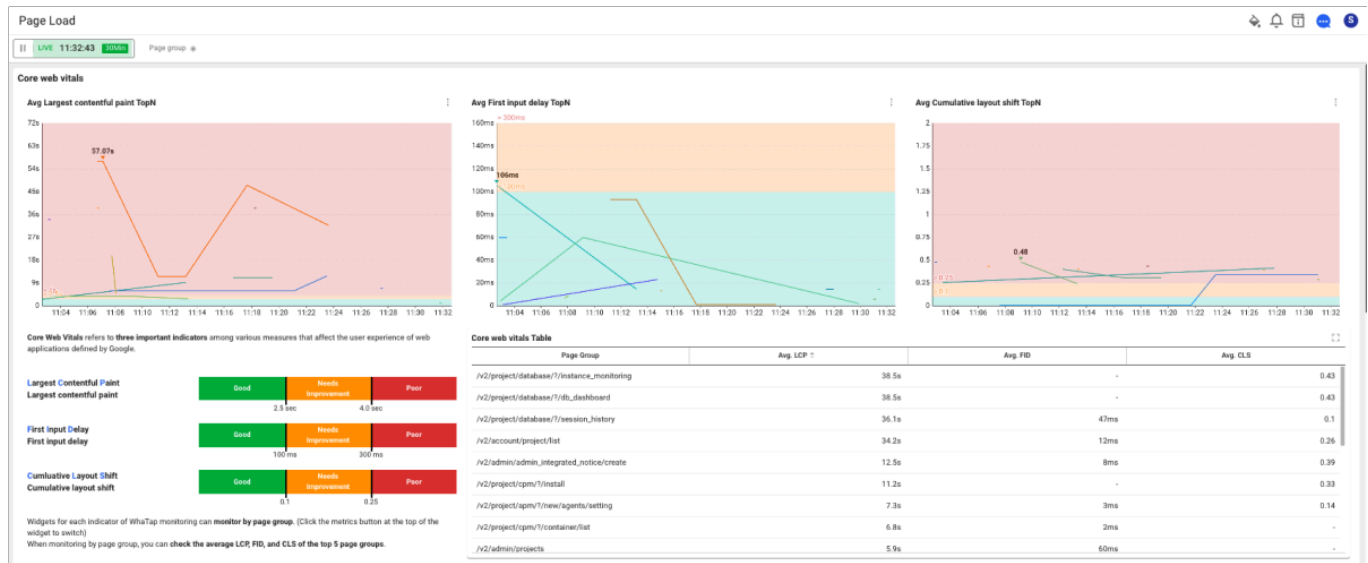
Line 2

:

Column 17194

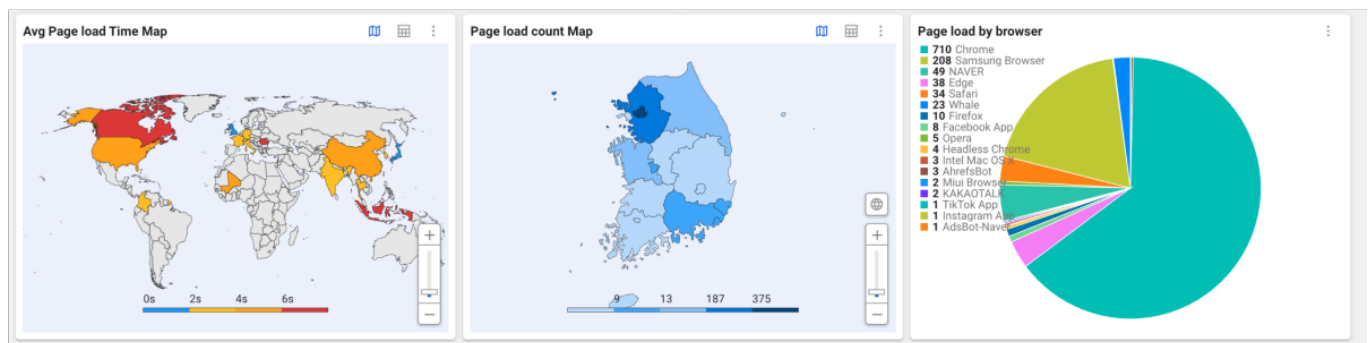
It collects browser error data that occurs while using the website. This information enables developers to identify and respond quickly to any problems while using the website. You can improve the user experience and increase the stability of the website.

Provision of the core web vitals information



Core Web Vitals is an important metric for the web page performance, defined by Google, including page load time, response time, rendering time, and user interaction time. By collecting and analyzing this data, we can help the website achieve higher exposure.

User access environment analysis



It analyzes various user access environments such as user device information, browser information, operating system information, and regional information.

If a problem occurs on a specific device, you can resolve the problem by analyzing information from the device. In addition, if the web application runs slowly in a specific region, you can optimize it by identifying information

in the region. By identifying the browser or operating system, we can provide the data needed to resolve web application compatibility issues.

Key Concepts

Before using the browser monitoring dashboard, be familiar with the terms defined by WhaTap.

Page Load

This process loads the content and resources of the page when accessing the web application. The browser monitoring service defines the page load time as follows.

- When there is no DOM interaction, AJAX request, or resource response within 100 ms
- When the `window onload` event occurs

The combination of the above conditions determines the final load time.

Changing the router

If you configure your web application as a Single Page Application (SPA), the page load event does not occur whenever you navigate to any page. We use the browser's history API to collect the data related to the page switching.

The router change time is defined as the page [First definition to the load](#).

The event ends when there is no DOM interaction, AJAX request, or resource response within 100 ms after page switching.

User Session

In the browser monitoring service, the **user session** is a unit of behavior by which a user continues to interact with the web application.

It defines the session between user's page entry to exit as one **user session**. The session has a maximum duration of 4 hours and it is reset if there is no interaction with the web application for 15 minutes.

Page group

To have a meaningful statistical group in the browser monitoring service, the URL path is normalized and provided under the name of **page group**. Numbers and dates that are judged to be meaningless in the URL path unit analysis are replaced with question marks (?) for provision.

For example, the URL path such as `/v2/project/sms/12/dashboard/resource_board` is provided in the form of `/v2/project/sms/?/dashboard/resource_board`.

Browser Error

The browser monitoring service classifies errors that occur in the browser as follows:

- `console` : This developer-defined error appears on the browser console.
- `onError` : This error occurs when a specific resource cannot be loaded or there is a problem in the script.
- `fetch error` : This error occurs when there is an HTTP request through a fetch.
- `xhr error` : This error occurs when there is an HTTP request through XHR.

Getting Started

WhaTap monitoring starts with the following steps:

[Step 1. Sign up](#)

[Step 2. Select a product](#)

[Step 3. Create a project](#)

[Step 4. Check the access key](#)

After confirming the supported environment, proceed to the agent installation step.

Sign up

To use the WhaTap monitoring service, you must first [sign up](#). For more information about signing up, see the [Account Management > Sign Up](#) documentation.

Select a product

1. Log in to the [WhaTap Monitoring Service](#).
2. Select  **All Projects** > **+ Project**.
3. On the **Select product** screen, select the Go product to monitor and then click the Create button.

Create a project

After selecting the product to monitor, create a project before installing the agent.

1. On the **Create project** screen, enter or select the following items to create a project:
 - **Project name** (required): Enter the project name.
 - **Data server region** (required): Select the region where the data server is located. A region is a collection of data centers that provide cloud services. Your data is stored in the selected region.

- **Time zone** (required): Set the time zone that serves as the reference for generating alerts and reports.
- **Notification language setting** (required): Set the language for alert notification messages generated in the project (supports Korean and English).
- **Project groups**: You can group and manage multiple projects. Select a group if there is one to join.
- **Project description**: Enter an additional description for the project.

2. After completing all settings, click the **Creating a project** button.

ⓘ When adding a project with an organization selected, you must set the **Groups of organization** as required. For more information about groups, see [the following documentation](#).

Check the access key

The **access key** is a unique ID for activating the WhaTap service.

In the installation guide section, select the **Getting the access key** button. After automatically issuing the **access key**, proceed to the next step.

ⓘ After creating a project, you are automatically redirected to the **Agent installation** page. If you are not redirected to the **Agent installation** page, select **Management > Agent installation** from the left menu.

Browser Compatibility

To start the Browser monitoring, apply the browser agent to the web application to collect. Before applying the browser agent, check the browser compatibility.

Collected data items for each browser version

The data that WhaTap browser agent can collect may differ depending on the browser and its version.

Collected data	Chrome	Edge	Safari	Firefox	Opera	Chrome Android	Safari iOS	Samsung Internet
Page Load	> 6	> 12	> 8	> 7	> 15	107	> 9	> 4
Resources for page load	> 28	> 12	> 11	> 35	> 15	107	> 11	> 4
Resources	> 52	> 79	> 11	> 57	> 39	107	> 11	> 6.2
AJAX	> 4	> 12	> 3.1	> 2	> 10	107	> 3.2	> 4
Core Web Vitals	> 77	> 79	fid only	fid only	> 64	107	fid only	> 12
Switching to the SPA page	> 4	> 12	> 3.1	> 2	> 10	107	> 3.2	> 4
console error	> 4	> 12	> 3.1	> 4	> 11.5	107	> 3.2	> 4
Exception unhandled promise error	> 49	> 79	> 11	> 69	> 36	107	> 11.3	> 5
Exception unhandled Java script error	> 10	> 12	> 5.1	> 6	> 15	107	> 5	> 4
Session Replay	> 47	> 79	> 8 *	> 6	> 55	> 47	> 9 *	> 5

⚠ To avoid overheads during **session replay** data collection, data is collected in idle time of the main thread. For this purpose, use the `requestIdleCallback` API.

The `requestIdleCallback` API is designed to perform a callback when the browser is idle, making sure that it does not interfere with the main tasks while collecting the **session replay** data. For more information, see the [following link](#).

✅ Safari and Safari iOS do not support the `requestIdleCallback` API, so session replay data is not collected by default. Setting the `sessionReplayCollectAllBrowser` option to true enables session replay data collection on these browsers. However, in this case, data is collected without checking main thread idle time, which may impact performance. For more details on applying this option, refer to [this link](#).

Preliminary Configuration

Before applying the Browser Agent and collecting data, you must complete the required pre-configuration.

Content Security Policy Settings

If your web application already has a Content Security Policy (CSP) configured, you must add the following items:

- The CDN URL of the Browser Agent
- The URL of the collection server used to transmit collected data

Additionally, to enable efficient data transmission by the Browser Agent, you must add CSP options related to Web Workers.

Directive	Source	Description
script-src, script-src-elem	unsafe-inline	Install the browser agent by using an inline script.
script-src, script-src-elem	https://repo.whatap-browser-agent.io/rum/prod/	Browser agent CDN domain
connect-src	https://rum-ap-northeast-2.whatap-browser-agent.io	Domain for transmission of collected data
worker-src, child-src	blob:	Web worker-related CSP options for efficient data transfer by the browser agent (recommended)

Timing-Allow-Origin Setting

On servers where CORS is applied, most resource timing attributes are returned as 0. To allow the Browser Agent to collect resource attribute values, the server must return the `Timing-Allow-Origin` HTTP response header.

How to configure

If you have control over the server that provides the resources, add the header to each response and specify all domains that are allowed to access the attributes (the domain of the website where the Browser Agent is applied).

- Example 1. Resource timing data is provided from all sources.

```
Timing-Allow-Origin: *
```

- Example 2. Resource timing data is provided only in the domain of the web application to which the browser agent is applied.

```
Timing-Allow-Origin: www.site.com
```

Application Of The Browser Agent

To use the WhaTap monitoring service, after [Sign up](#), create a project and then apply the WhaTap browser agent to web applications.

See the following video guide.

Installing the WhaTap browser agent

Follow the instructions on the agent installation screen to apply the WhaTap browser agent code to your web application.

▼ 2. Install Whatap Browser Agent

1) Browser agents collect data based on user sessions. You can set the percentage of total sessions that you collect.

100 %

Async sync

Load the browser agent asynchronously to avoid affecting the page load performance.

However, data such as ajax and error that occurred before the browser agent is loaded may be missing when collecting.

```
<script>
(function (w, h, _a, t, a, b) {
  w = w[a] = w[a] || {
    config: {
      projectAccessKey: "x4vt222vl9j4k-xiu0ec4m206it-x1928rof7idesu",
      pcode: 32674,
      sampleRate: 100,
      proxyBaseUrl: "https://rum-ap-northeast-2.whatap-browser-agent.io/",
    },
  };
  a = h.createElement(_a);
  a.async = 1;
  a.src = t;
  t = h.getElementsByTagName(_a)[0];
  t.parentNode.insertBefore(a, t);
})(window, document, 'script', 'https://repo.whatap-browser-agent.io/rum/prod/v1/whatap-browser-agent.js', 'WhatapBrowserAgent', '');
</script>
```

Copy

Add the above script code to the top of the head tag in every HTML page you want to monitor.

1 Data collection sampling

The WhaTap browser agent collects data based on the user session. You can set the percentage of all sessions collected from 0 to 100.

2 WhaTap browser agent script

The WhaTap browser agent is provided in the form of an inline script. Add the script code provided by the installation guide to the top in the `<head>` tag of any HTML page to monitor.

Select one of the following two methods to apply the agent.

- **Async** (Asynchronous Load): Loads the WhaTap browser agent asynchronously in your web application.
 - It does not affect the load performance of your web application.
 - There may be missing data such as AJAX or errors that occurred before loading the browser agent.
- **Sync** (Synchronous Load): Loads the WhaTap browser agent synchronously in your web application.
 - It is recommended if you want to collect all data when loading your web application.
 - It may affect the load performance of your web application.

Setting the WhaTap browser agent's options

It sets the options to apply to the WhaTap browser agent. The options can be set in the Config object of the installation script. You can set the project access keys, percentage of all user sessions, resource domains to exclude from collection, and such.

config example

```
1 config: {
2   projectAccessKey: {projec_access_key},
3   pcode: {pcode},
4   sampleRate: 100,
5   ignoreOrigins: [ 'https://ignore-site.com/', /^((https?:\/\/)([^/]*)(ignore-site.io)(/)(.*)/i ],
6 }
```


Option parameters

projectAccessKey String required

It is a project access key. You can see in the project installation guide (**Management > Agent Installation**).

```
1 config: {
2   projectAccessKey: {projec_access_key},
3   pcode: {pcode},
4   sampleRate: 100,
5   proxyBaseUrl: "https://repo.whatap-browser-agent.io/rum/prod/v1/whatap-browser-agent.js"
6 }
```

pcode Number required

It is a project access key. You can see in the project installation guide (**Management > Agent Installation**).

```
1 config: {
2   projectAccessKey: {projec_access_key},
3   pcode: {pcode},
4   sampleRate: 100,
5   proxyBaseUrl: "https://repo.whatap-browser-agent.io/rum/prod/v1/whatap-browser-agent.js"
6 }
```

sampleRate Number required

You can set the ratio of user sessions to be collected. You can set it from 0 to 100.

```
1 config: {
2   projectAccessKey: {projec_access_key},
3   pcode: {pcode},
4   sampleRate: 100,
5   proxyBaseUrl: "https://repo.whatap-browser-agent.io/rum/prod/v1/whatap-browser-agent.js"
6 }
```

proxyBaseUrl String **required**

It is the agent collection data transmission URL.

```
1 config: {
2   projectAccessKey: {projec_access_key},
3   pcode: {pcode},
4   sampleRate: 100,
5   proxyBaseUrl: "https://repo.whatap-browser-agent.io/rum/prod/v1/whatap-browser-agent.js"
6 }
```

ignorePageUrls Array<string | RegExp>

optional

It is a list of page URLs to exclude from collection. String matching is performed using the `startsWith` method.

```
1 config: {
2   projectAccessKey: {projec_access_key},
3   pcode: {pcode},
4   sampleRate: 100,
5   proxyBaseUrl: "https://repo.whatap-browser-agent.io/rum/prod/v1/whatap-browser-agent.js"
6   ignorePageUrls: ["https://test.webpage.com/", "http://localhost:2003/page1/", /^.localhost.$/i]
7 }
```

ignoreResources Array<string | RegExp>

optional

It is a list of resource URLs to exclude from collection. String matching is performed using the `startsWith` method.

```
1 config: {
2   projectAccessKey: {projec_access_key},
3   pcode: {pcode},
4   sampleRate: 100,
5   proxyBaseUrl: "https://repo.whatap-browser-agent.io/rum/prod/v1/whatap-browser-agent.js"
```

```

6   ignoreResources: ["https://test.web.com/yard/api/flush", "http://localhost:2003/whatap-browser-agent.js",
  /\^\.Vpath1Vapi.$/i]
7 }

```

ignoreErrors Array<string | RegExp>

optional

It is a list of browser errors to exclude from collection. String matching is performed using the `includes` method.

```

1  config: {
2    projectAccessKey: {projec_access_key},
3    pcode: {pcode},
4    sampleRate: 100,
5    proxyBaseUrl: "https://repo.whatap-browser-agent.io/rum/prod/v1/whatap-browser-agent.js"
6    ignoreErrors: ["cannot read", "cors", "basic"]
7  }

```

collectUserClick Boolean

optional

Default `false`

It can collect user click events. For the method to find the collected data, see [the following](#).

```

1  config: {
2    projectAccessKey: {projec_access_key},
3    pcode: {pcode},
4    sampleRate: 100,
5    proxyBaseUrl: "https://repo.whatap-browser-agent.io/rum/prod/v1/whatap-browser-agent.js"
6    collectUserClick: true
7  }

```

sessionReplaySampleRate Number

optional

Default `0`

It is the ratio of sessions to collect the session replay data. You can set from 0 to 100 for user sessions to collect.

For example, if `sampleRate` is set to **50** and `sessionReplaySampleRate` to **20**, 50% of all sessions are collected, and for only 20% of those sessions, the session replay data is collected.

```
1 config: {
2   projectAccessKey: {projec_access_key},
3   pcode: {pcode},
4   sampleRate: 100,
5   proxyBaseUrl: "https://repo.whatap-browser-agent.io/rum/prod/v1/whatap-browser-agent.js"
6   sessionReplaySampleRate: 50
7 }
```

sessionReplayMaskAllTexts Boolean

optional

Default `true`

If you set the value to `false`, all text data is collected without masking.

```
1 config: {
2   projectAccessKey: {projec_access_key},
3   pcode: {pcode},
4   sampleRate: 100,
5   proxyBaseUrl: "https://repo.whatap-browser-agent.io/rum/prod/v1/whatap-browser-agent.js"
6   sessionReplaySampleRate: 50,
7   sessionReplayMaskAllTexts: false
8 }
```

sessionReplayMaskAllInputs Boolean

optional

Default `true`

If you set the value to `false`, the data in all input fields is collected without masking.

```
1 config: {
2   projectAccessKey: {projec_access_key},
3   pcode: {pcode},
4   sampleRate: 100,
5   proxyBaseUrl: "https://repo.whatap-browser-agent.io/rum/prod/v1/whatap-browser-agent.js"
6   sessionReplaySampleRate: 50,
7   sessionReplayMaskAllInputs: false
8 }
```

sessionReplayCollectAllBrowser Boolean

optional

Default `false`

The session replay data is collected even in the browsers that do not support `requestIdleCallback()` . e.g. Safari, Safari on iOS

```
1 config: {
2   projectAccessKey: {projec_access_key},
3   pcode: {pcode},
4   sampleRate: 100,
5   proxyBaseUrl: "https://repo.whatap-browser-agent.io/rum/prod/v1/whatap-browser-agent.js"
6   sessionReplaySampleRate: 50,
7   sessionReplayCollectAllBrowser: true
8 }
```

ignoreStatusZero Boolean optional

Default `false`

This option excludes data from collection when the status code of the AJAX request is `0`.

```
1 config: {
```

```

2  projectAccessKey: {project_access_key},
3  pcode: {pcode},
4  sampleRate: 100,
5  proxyBaseUrl: "https://repo.whatap-browser-agent.io/rum/prod/v1/whatap-browser-agent.js",
6  ignoreStatusZero: true
7  }

```

- ! • For more information about the session replay, see [the following](#).
- For more information about the browsers that support session replay collection, see [the following](#).

Next steps

- **Collecting custom events**

To identify the web service issues and improve user experience through browser monitoring, it provides an interface that allows developers and operators to additionally collect desired events that occur on web pages. For more information about custom events, see [the following](#).

- **Setting the actual user ID**

In the browser monitoring, you can collect data by setting the user ID via the actual user's login ID or email address. You can check the user session performance and event information based on the actual login ID, and then check the browser's errors to identify problems. For more details, see [the following](#).

- **Configuring the session replay**

Session replay is a feature that allows to record and replay all events performed by users on the website. This feature allows you to reproduce user behaviors such as clicking, scrolling, typing, and page switching. This helps you understand exactly how users actually interact with the website. For more details, see [the following](#).

- **Starting the monitoring**

Go to [WhaTap Monitoring Service](#) and then start the browser monitoring. Select a created project and then go to **Dashboard > Browser Monitoring Dashboard**. You can see the monitoring status. For more information about **Browser Monitoring Dashboard**, see [the following](#).

Setting The Actual User ID

Browser monitoring is basically managed by issuing user IDs with random numbers. However, it provides an additional function to collect data by setting the user ID via the actual user's login ID or email address. By setting the actual user ID, you can see the user session performance and event information and additionally see the browser errors based on the actual login ID or email address.

By identifying users through the browser monitoring, it can be used for custom analysis, accurate user experience analysis, marketing strategy planning, and security enhancement.

Setting the actual user ID

Interfaces

The interface provided by the browser agent is contained in the `WhatapBrowserAgent` object of `window`. See the interface of `setUserID` provided by `WhatapBrowserAgent`.

Typescript

```
setUserID: (userID: string) => void;
```

Application example

You can apply the interface provided by the browser agent as an example code.

Typescript

```
const loginComplete = (data) => {
  if (data) {
    /*
     * @description Setting the user ID of the WhaTap browser agent
     */
    window?.WhatapBrowserAgent?.setUserID?.(data?.email);
  }
};
```

When the `loginComplete` function runs, the actual user's email address is set as the user ID to be used for browser monitoring.

APM Integration Settings

You can link HTTP requests generated in the browser to backend transactions.

When linked, you can view frontend indicators together with backend and infrastructure performance information, which helps you quickly identify and understand the causes of delays and errors that real users experience.

Requirements

- An application with Whatap APM installed is required.
- You must configure the AJAX request URLs to be linked in the browser monitoring agent.
- The server providing the HTTP API may need to **configure and allow Access-Control-Allow-Headers** for the headers below(Important).

Table. Access-Control-Allow-Headers configuration and settings

Info	Description
x-wtap-po	Project information used for MTID tracking
x-wtap-mst	Transaction information used for MTID tracking
x-wtap-sp1	Page URL-related information configuration

Browser Agent Settings

Specify the HTTP request URLs to be traced.

Configure them in the `config` section of the browser monitoring agent installation script.

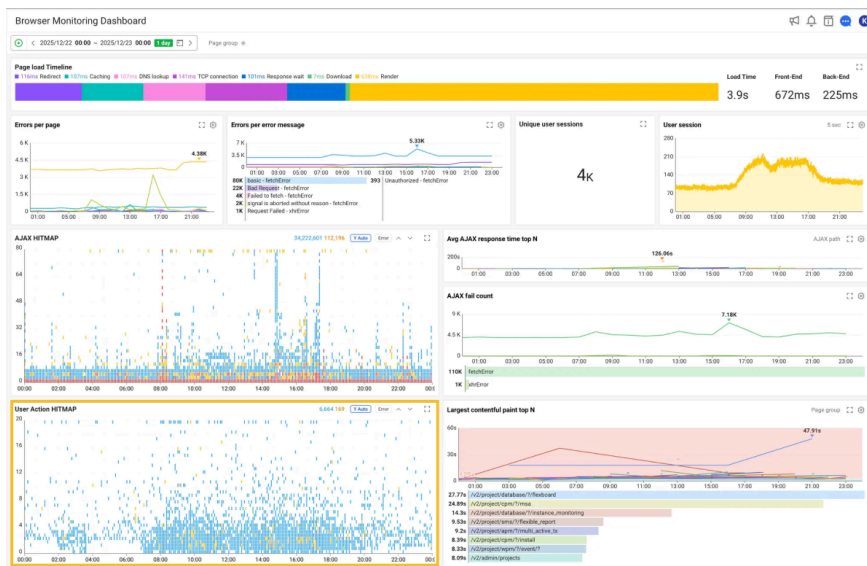
- Register target AJAX URLs as a string array in the installation script option `tracingAjaxUrls` .
- You can configure the ratio of multi-transaction generation using `traceSampleRate` .

```

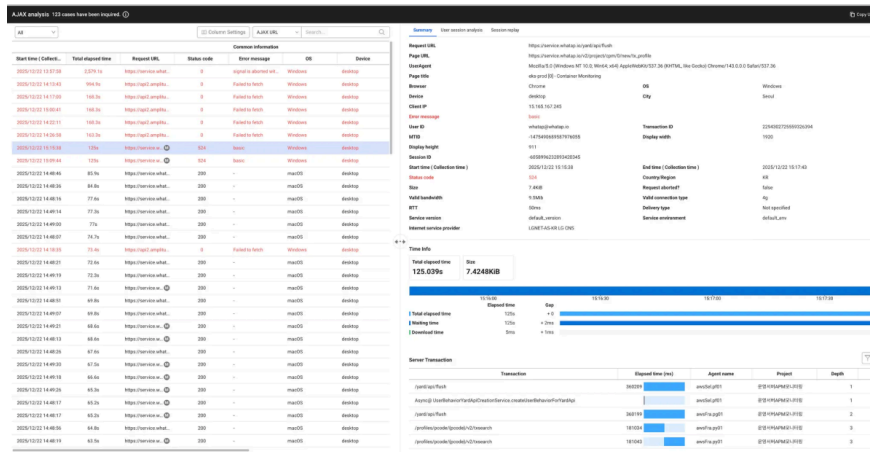
1 <script>
2 (function (w, h, _a, t, a, b) {
3   w = w[a] = w[a] || {}
4   config: {
5     projectAccessKey: "xxxxxxxxxxxx-xxxxxxxxxxxx-xxxxxxxxxxxx",
6     pcode: 1,
7     sampleRate: 100,
8     proxyBaseUrl: "https://remote.whatap-browser-agent.io/",
9     traceSampleRate: 50,
10    tracingAjaxUrls: ['http://dev.whatap.io/yard/api/flush', 'http://devote.whatap.io/',
11    /^https?:\\V\\localhost(?:\\d+)?(\\V.*?)?$/]
12  },
13  };
14  a = h.createElement(_a);
15  a.async = 1;
16  a.src = t;
17  t = h.getElementsByTagName(_a)[0];
18  t.parentNode.insertBefore(a, t);
19  })(window, document, 'script', '${front_rum_url}', 'WhatapBrowserAgent', '');
20 </script>

```

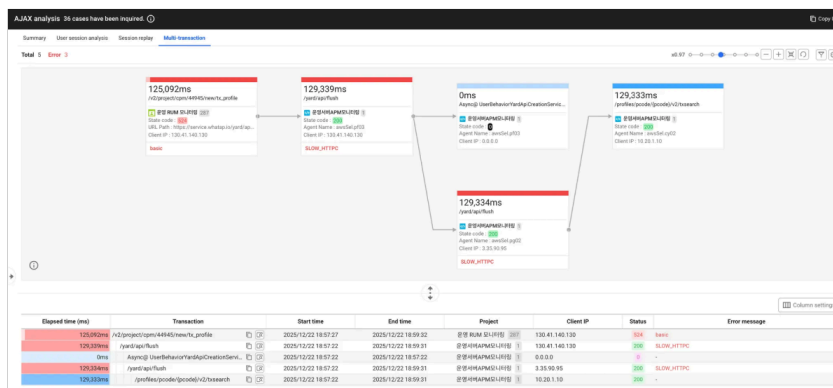
1. In the AJAX Hitmap where real-time and historical data can be viewed, check the HTTP requests occurring at the client level.



2. Drag the hitmap to select the analysis range and review the table.



- You can check AJAX requests generated on the client side by user ID, page URL, AJAX URL, and more, and you can link them to the backend transactions processed in APM.



- When clicking a node, you can view detailed information about the APM transaction on that screen.

Applications

No.	Name	Type	Group
1	Applications	Applications	Applications
2	Applications	Applications	Applications
3	Applications	Applications	Applications
4	Applications	Applications	Applications
5	Applications	Applications	Applications
6	Applications	Applications	Applications
7	Applications	Applications	Applications
8	Applications	Applications	Applications
9	Applications	Applications	Applications
10	Applications	Applications	Applications
11	Applications	Applications	Applications
12	Applications	Applications	Applications
13	Applications	Applications	Applications
14	Applications	Applications	Applications
15	Applications	Applications	Applications
16	Applications	Applications	Applications
17	Applications	Applications	Applications
18	Applications	Applications	Applications
19	Applications	Applications	Applications
20	Applications	Applications	Applications

Collecting Custom Events

It may be difficult to identify problems with web services and improve user experience only with basic information collected through the browser monitoring. To handle those difficulties, it provides an interface that allows developers and operators to additionally collect desired events that occur on the website.

The following event types can be additionally collected through the interface provided by the browser agent:

- Information on specific function calls for debugging
- Logic execution information related to the sales
- Button click information that directly affects the sales

You can use custom events as follows:

- **Understanding and improving user experience**

By tracing the behaviors or visited paths of specific users, we can better understand how users interact with the website and improve user experience. For example, measuring the clicks of specific buttons within the website or the transition speed between pages can help you discover and fix user experience issues.

- **Identifying and resolving performance issues**

Custom events can help you identify performance issues for specific features or components. You can trace the times to perform specific tasks and collect performance data to find bottlenecks. If you can address these performance issues, the overall performance of your website can be improved.

- **Provision of business insight**

By understanding how users interact with web applications and website, valuable information for business decisions can be obtained. By analyzing information such as most frequently used functions and chosen paths in web services, you can use the information for developing desired functions or marketing strategies.

- **Generating custom notifications and alerts**

By configuring notifications for specific event conditions, you can quickly detect and respond to system failures or unexpected user behaviors.

Collection data format

- `name` : Name that represents the event. It is recommended not to include unique IDs that cannot be grouped, such as random numbers or product numbers.
- `duration` : Time between a page load or page transition and an event occurrence.
- `custom_duration` : Execution period of a custom event. You can set the time at which a specific event is performed by directly specifying the start time and end time.
- `contents` : Processed results of custom events.

How to apply and collect codes

You can send custom events to WhaTap via the `addCustomEvent` method among the interfaces provided by the browser agent.

Interfaces

The interface provided by the browser agent is contained in the `WhatapBrowserAgent` object of `window`. See the interface of `addCustomEvent` provided by `WhatapBrowserAgent`.

Typescript

```
addCustomEvent: (
  eventName: string,
  option: {
    contents: string | undefined;
    customDuration: number | undefined;
  } | undefined,
) => void;
```

Application example

You can apply the interface provided by the browser agent as an example code.

Typescript

```
function PurchaseProduct(payload) {
  const startTimestamp = new Date();

  ...

  const endTimestamp = new Date();

  const duration = endTimestamp - startTimestamp;

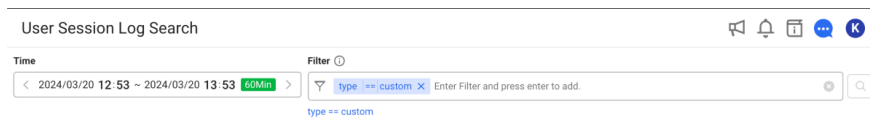
  window.WhatapBrowserAgent?.addCustomEvent('purchase product', {
    contents: JSON.stringify(payload),
    customDuration: duration,
  });
}
```

When the `PurchaseProduct` function is run, custom events are sent with the specified `name` value.

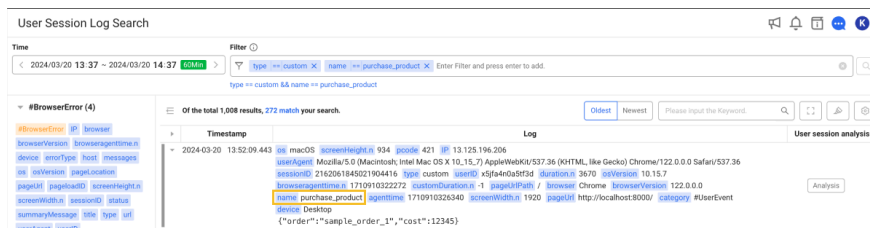
Viewing the collected data

Using the user session log search menu

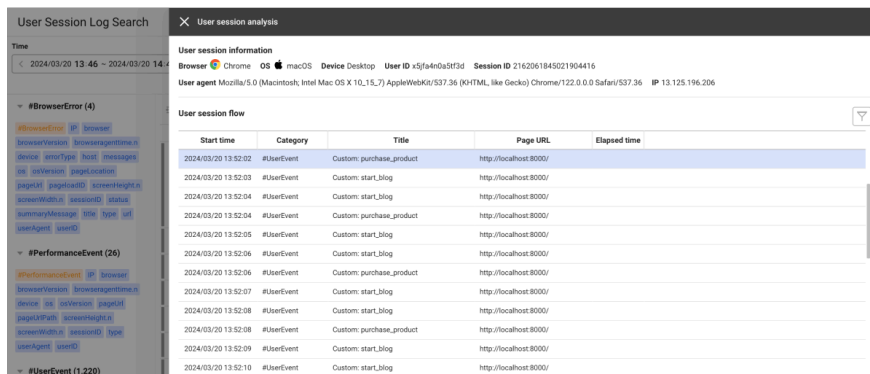
1. Select a browser monitoring project and then go to **Analysis > User Session Log Search**.
2. In the **Filter** option, select the `type` tag and then enter "custom."



3. Select .
4. You can additionally select the `name` tag from the searched results and then filter out specific values.



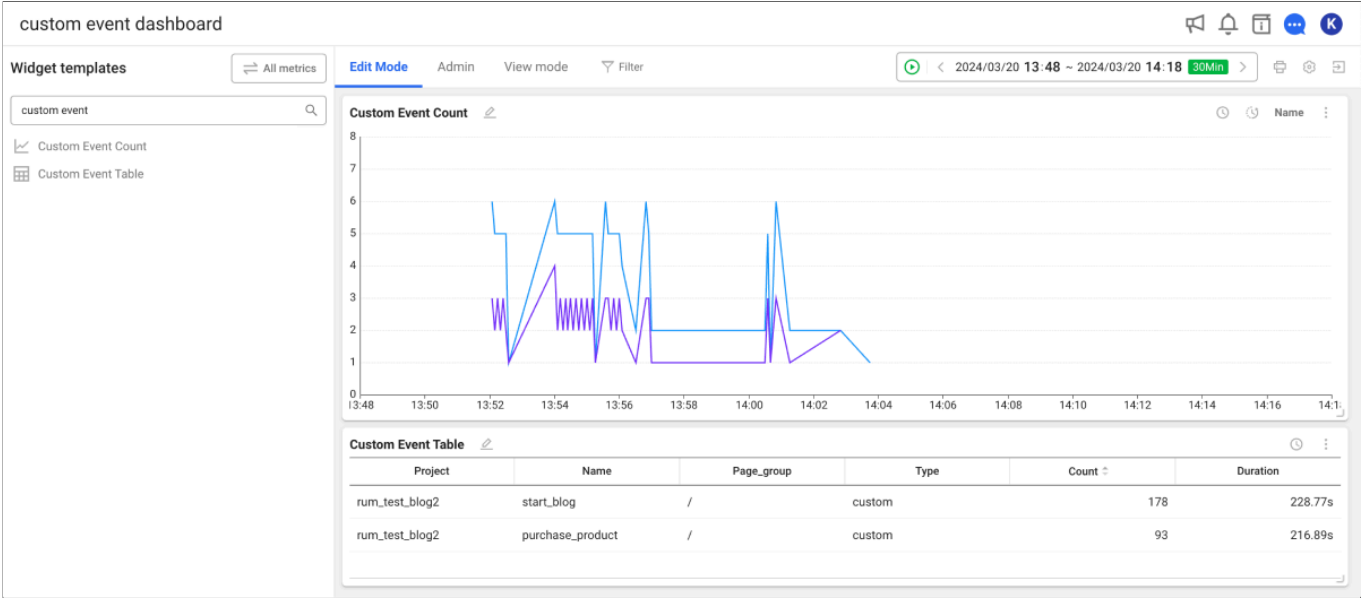
- To see user behaviors until an event occurs, in the **User session analysis** column from the searched result list, select **Analysis**.



! For more information about the user session analysis, see the following.

Checking with widgets on the Flex board

You can view custom event data by creating a widget in **Flex Board**.



- Custom event count: You can see the trend in collected custom events.
- Custom event table: You can see the list of the collected custom events.

Collecting Custom Logs

The browser agent exposes a `logger` interface so your web application can collect arbitrary log messages directly. Attach a level, context, and error to each message, then search and filter them in the WhaTap console. [Collecting custom events](#) captures the duration and result of an event; custom logs record level-based messages emitted while the application runs.

! Custom logs work only with browser agent **3.0 or later**. If you are on an earlier version, upgrade the agent before using this feature.

! Log monitoring is a billable feature. A 15-day free trial begins on the activation date, after which charges apply based on the collected logs.

Prerequisites

Obtain the following values from the WhaTap console.

Item	Description
Access key	Project access key (<code>projectAccessKey</code>)
Project code	Numeric project code (<code>pcode</code>)
Collector URL	The collector URL that receives data (<code>proxyBaseUrl</code>)

! If `proxyBaseUrl` is empty, all logs are silently discarded. Be sure to set it.

Enabling

To use custom logs, configure the following two settings.

1. **Log settings:** Turn on the **Activate log monitoring** toggle on the **Log Configuration** screen. Logs are collected only while log monitoring is on; turning the toggle off stops storing logs.
2. **Custom log option:** The browser agent loads asynchronously by default. Use the standard installation snippet from the WhaTap console, and add `enableCustomLog: true` to the `config` object.

```
<script>
(function (w, h, _a, t, a, b) {
  w = w[a] = w[a] || {
    config: {
      projectAccessKey: 'YOUR_ACCESS_KEY',
      pcode: 12345,
      sampleRate: 100,
      proxyBaseUrl: 'https://your-collector.example.com/',
      enableCustomLog: true, // Enable custom logs
      env: 'prod',
      version: '1.0.0'
    }
  };
  a = h.createElement(_a);
  a.async = 1;
  a.src = t;
  t = h.getElementsByTagName(_a)[0];
  t.parentNode.insertBefore(a, t);
})(window, document, 'script',
  'https://repo.whatap-browser-agent.io/rum/v3/whatap-browser-agent.js',
  'WhatapBrowserAgent', "");
</script>
```

Configuration options

Option	Type	Required	Description
<code>projectAccessKey</code>	<code>string</code>	Required	Access key issued by the WhaTap console
<code>pcode</code>	<code>number</code>	Required	Project code
<code>proxyBaseUrl</code>	<code>string</code>	Required	Collector base URL. Logs are not sent if unset
<code>enableCustomLog</code>	<code>boolean</code>	Required	Must be set to <code>true</code> to enable (default <code>false</code>)
<code>env</code>	<code>string</code>	Optional	Environment label (e.g., <code>prod</code> , <code>staging</code> , <code>dev</code>). Default <code>default_env</code>
<code>version</code>	<code>string</code>	Optional	Application version. Used to distinguish logs per deployment. Default <code>default_version</code>

ⓘ The `logger` interface is exposed only after the agent script has been downloaded and executed. Calls made very early in page load may be ignored. For a safe calling approach, see [Asynchronous load and safe calls](#).

⚠ Custom logs follow the session's sampling decision. If `sampleRate` excludes the current session from collection, the custom logs of that session are also not sent. To collect logs from all sessions, keep `sampleRate: 100` (the default).

Interface

The `logger` interface is included in the `WhatapBrowserAgent` object on `window`.

Typescript

```
window.WhatapBrowserAgent.logger.debug(message, context, error);
window.WhatapBrowserAgent.logger.info (message, context, error);
```

```

window.WhatapBrowserAgent.logger.warn(message, context, error);
window.WhatapBrowserAgent.logger.error(message, context, error);
window.WhatapBrowserAgent.logger.log (message, context, status, error);

```

Argument	Type	Required	Description
message	string	Required	Log body. Up to 8KB
context	object	Optional	Additional metadata. Used for search and filtering
error	Error or similar object	Optional	Used when sending with a stack trace

- All methods are synchronous and return nothing.
- Each call is buffered immediately and flushed together after about 1 second.
- A bad argument never throws, so it cannot break your application.
- `logger.log()` takes `status` (`'debug'` | `'info'` | `'warn'` | `'error'`) as its third argument; an invalid value falls back to `'info'` .

Usage

Basic call

```

window.WhatapBrowserAgent.logger.info('checkout button clicked');

```

Sending with context

Pass an object as the second argument to make the log searchable and filterable in the WhaTap console.

```

window.WhatapBrowserAgent.logger.info('checkout step', {
  step: 'shipping',
  cartItems: 3,
  isPremiumUser: true
});

```

Only strings, numbers, and booleans in `context` are sent as is. Objects, arrays, `null`, `undefined`, and functions are automatically excluded, so flatten nested values or serialize them to strings.

```
// Ignored: user is an object
window.WhatapBrowserAgent.logger.info('login', {
  user: { id: 42, name: 'jihoon' }
});

// Flattened
window.WhatapBrowserAgent.logger.info('login', {
  userId: 42,
  userName: 'jihoon'
});

// Serialized
window.WhatapBrowserAgent.logger.info('login', {
  userJson: JSON.stringify({ id: 42, name: 'jihoon' })
});
```

Sending with errors

Pass an `Error` object or a caught value directly as the third argument, and `name`, `message`, and `stack` are sent together.

```
try {
  riskyOperation();
} catch (e) {
  window.WhatapBrowserAgent.logger.error(
    'riskyOperation failed',
    { feature: 'checkout' },
    e
  );
}
```

- `Error` instance: `name`, `message`, and `stack` are extracted as is.
- Plain object: the `name` / `message` / `stack` properties are used if present.
- Otherwise (string, number, etc.): converted to a string and placed in `message`.

Log levels

Level	Recommended use
debug	Detailed tracing during development. Avoid in production
info	Normal flow events (page entry, business milestones)
warn	Recovered abnormal situations (retry success, fallback used)
error	Failures that affect user experience (payment failure, data loss)

The WhaTap console color-codes each level.

Usage patterns

Tracking the checkout flow

```
const log = window.WhatapBrowserAgent.logger;

log.info('checkout:start', { cartId, items: cartItems.length, total });

try {
  const result = await pay(cartId);
  log.info('checkout:success', { cartId, paymentId: result.id });
} catch (e) {
  log.error('checkout:fail', { cartId, gateway: 'PG_X' }, e);
}
```

Augmenting global errors

Attach extra business context on top of the agent's automatic error collection.

```

window.addEventListener('error', (event) => {
  window.WhatapBrowserAgent.logger.error(
    'uncaught: ' + event.message,
    { filename: event.filename, line: event.lineno },
    event.error
  );
});

```

Send timing

- Logs are flushed together when no new call arrives for about 1 second.
- They are flushed at the latest 5 seconds after the first call.
- They are flushed immediately once 500 entries or about 512KB accumulate in a short time.
- A single message is sent up to 8KB; anything beyond that is truncated.

The defaults are enough in most cases and need no tuning.

Cautions

Caution	Reason
Do not send personal data directly	Mask values such as resident registration numbers, card numbers, and phone numbers in your code before sending
Do not call unconditionally inside loops	Calling too frequently can add network overhead
Messages over 8KB	Automatically truncated. Split large payloads into the <code>context</code> field
Do not pass objects or arrays directly in <code>context</code>	They are ignored. Flatten them or serialize them to strings

Asynchronous load and safe calls

If you need to call `logger.*` the moment the page opens, check that `logger` exists first.

```
function safeLog(level, message, context, error) {
  const agent = window.WhatapBrowserAgent;
  if (agent && agent.logger && typeof agent.logger[level] === 'function') {
    agent.logger[level](message, context, error);
  }
  // Ignored if before load.
}

safeLog('info', 'app boot', { route: location.pathname });
```

- ✓ Calls triggered by user events (clicks, form submissions, etc.) usually run after the agent has loaded, so they need no guard. Use the pattern above only in page boot code.

Troubleshooting

Symptom	Cause / action
<code>window.WhatapBrowserAgent.logger</code> is <code>undefined</code>	(1) Agent is below version 3.0 (2) Asynchronous load is not finished yet (3) Script blocked by CSP or ad blocker
Custom log call was dropped warning in the console	<code>enableCustomLog</code> is <code>false</code> . Add <code>enableCustomLog: true</code> to the config. This warning is printed only once
No network request even after calling	Missing <code>enableCustomLog: true</code> , unset <code>proxyBaseUrl</code> , or the call ran before the configuration
401 / 403 response	Access key or <code>pcode</code> error
All logs of a specific session are	<code>sampleRate</code> excluded that session from collection. Set <code>sampleRate: 100</code>

Symptom	Cause / action
missing	to collect all sessions
Only some logs are missing	Entries beyond 500 or about 512KB in a short time are sent immediately (normal behavior)
Message arrives truncated	Single message limit of 8KB. Excess is truncated automatically

Sent logs can be checked in the WhaTap console under **Analysis > User Session Log Search**. Search and filter by `status` , `message` , `context` keys, `error.message` , and more.

Browser Monitoring Dashboard

Home > Select Project > [Dashboard](#) > [Browser Monitoring Dashboard](#)

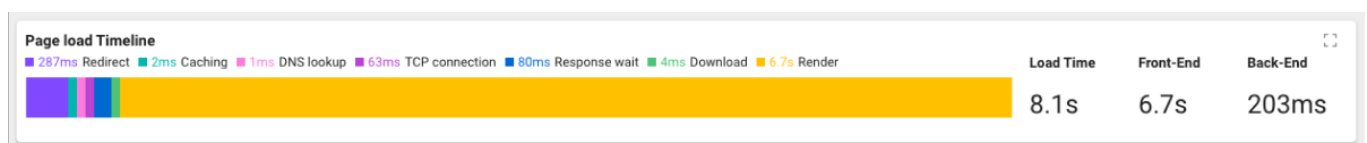


This is a dashboard where you can search real-time and historical data. Monitor the user experiences and performances of the browser applications. It provides data from various aspects, such as performance metrics, browser, and region, so that you can quickly identify health anomalies. It also provides the performance data for page load events and AJAX events.

This dashboard organizes real-time statistical data into chart-type widgets. Statistical data based on the page load and AJAX data generated by the web browser can be checked in real time.

- On the upper right of each widget, select . Then you can see the widget data in full screen.
- If you select  on the upper right of the widget, the chart configuration screen appears. You can set the options to display on the chart.
- Depending on the screen size of the web browser, the icon shape on the upper right of a widget may differ. If you select , the additional icons appear.
- For more information about the widget configuration, see [the following](#).

Page Load Timeline



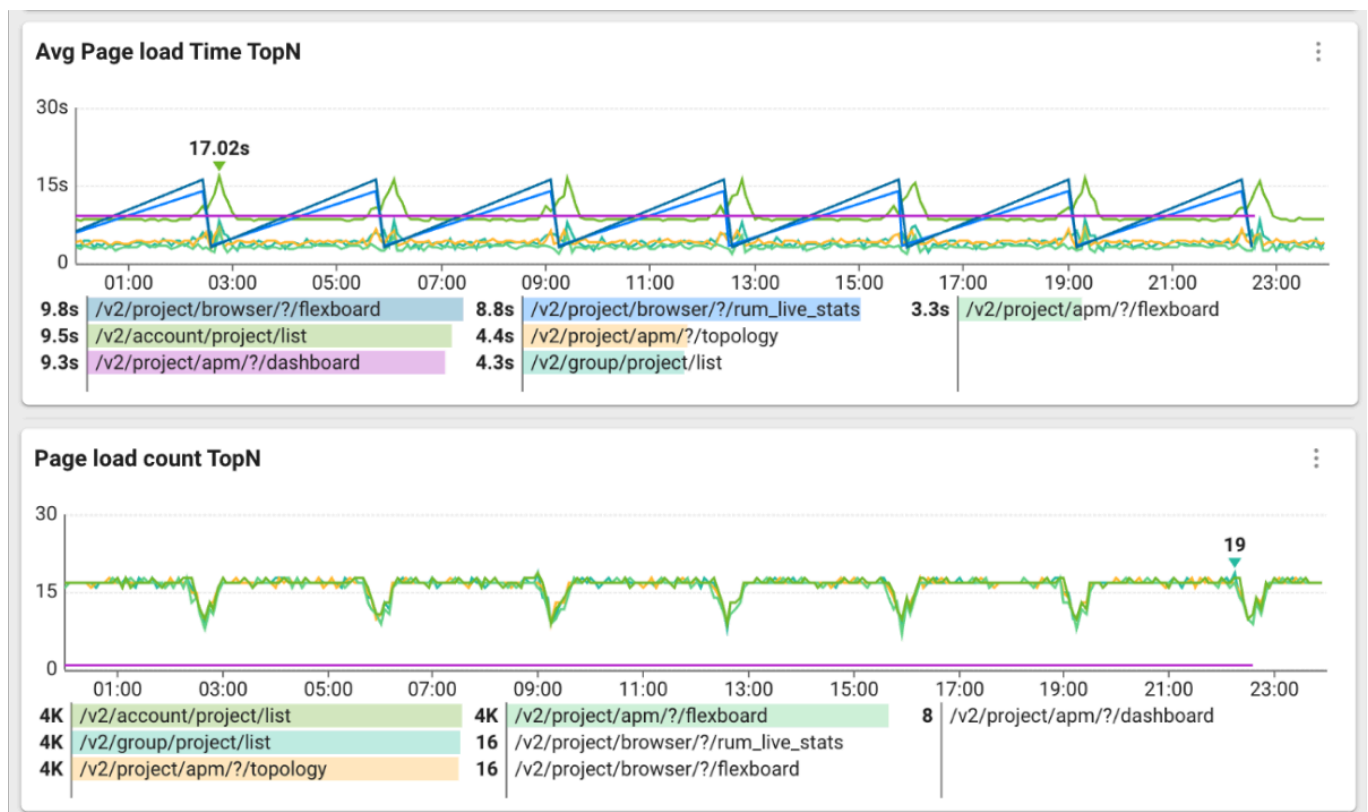
It provides the overall page loading performances for the browser applications. It displays the page load times and the average times for each load step. You can see the overall performance of your browser applications through this.

For detailed descriptions of each step, see the following.

- **Redirect:** The average amount of redirection time spent by the browser when loading web pages.
- **Caching:** The average time spent by a browser to search cached resources while loading web pages.
- **DNS Lookup:** The average time spent by a browser to look up website domains while loading web pages.
- **TCP Connection:** The average TCP handshake time spent by a browser when loading web pages.
- **Response Wait:** The average time spent from the browser sending network requests while fetching web pages to the first byte receptions from the server.

- **Download:** The average time spent for a browser to download resources from the server while loading web pages.
- **Rendering:** Average time spent rendering resources downloaded from the server to the screen and completing page load events.
- **Load Time:** The average time spent for a browser to fully load web pages.
- **Front-End:** Average time taken to initially render web pages.
- **Back-End:** Average time taken to download resources by page load requests.

Page loading time and load count



When end users access the web pages via the browser application, they can see in real time how long and how often each page takes to load. With this information, the developers can identify high-traffic and slow-loading pages and prioritize which pages need speed improvements.

For example, if a page accessed by many users takes long time to load, the developers can focus on improving

the performance of the page. On the contrary, if a page with slow loading speed has a low traffic, it can be more efficient to invest more development resources in other pages rather than improving the page.

! To check the overall status, select [Page Group](#) in the upper right corner. In the metrics list, select [All](#).

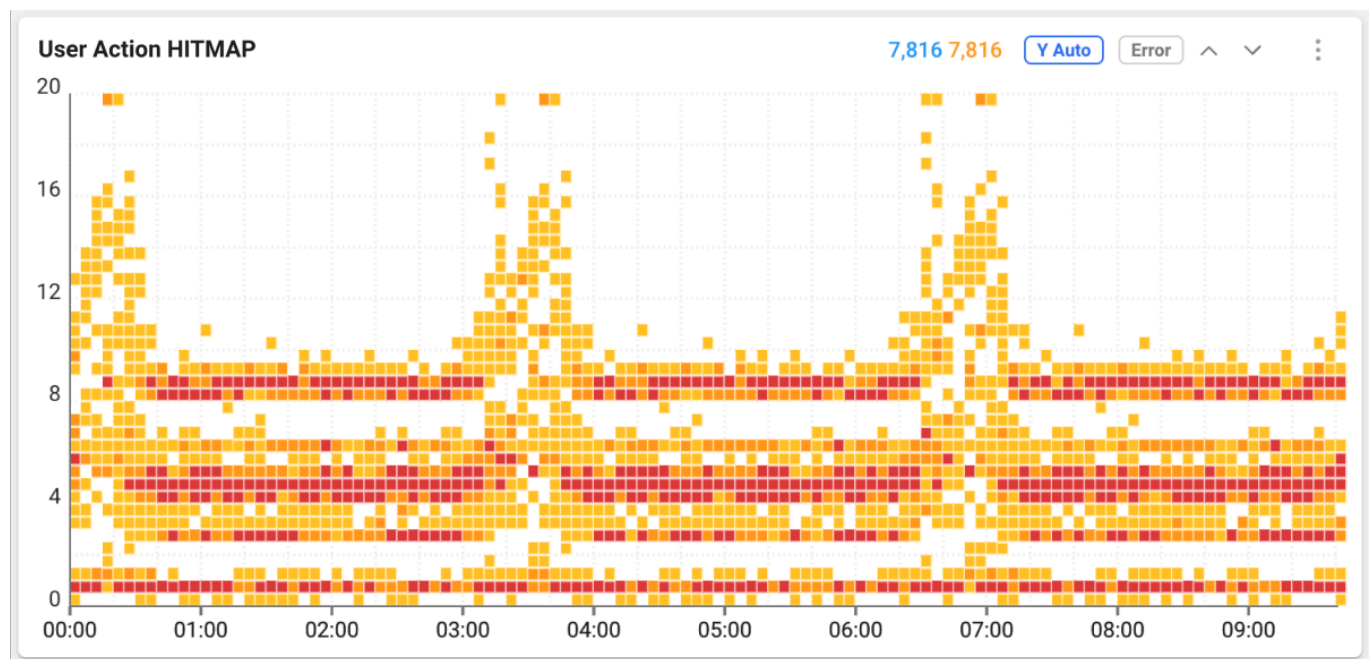
- [Average Page Load Time TopN](#)

It displays the average page load times by page group. The horizontal bar chart displays the average load times for each page group during the lookup period. Only the top 8 are displayed.

- [Page Load Count TopN](#)

Page load counts are displayed by the page group. The horizontal bar chart displays the sum of loads for each page group during the lookup period. Only the top 8 are displayed.

Page Load Hitmap



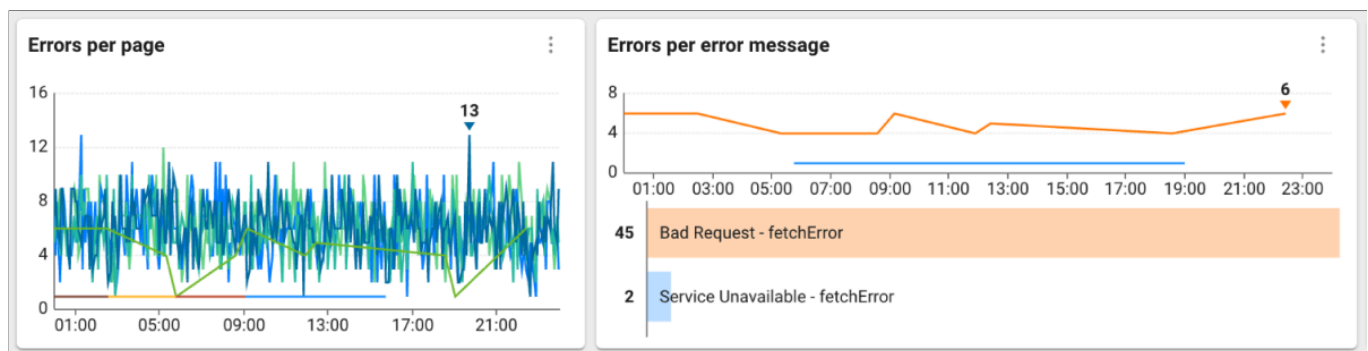
[Page Load Hitmap](#) is a distribution chart of response times for page loads over time.

- The horizontal axis is the end time of the page load event, and the vertical axis is the elapsed time of the page

load event.

- Normal page load events are displayed in light blue → blue → Indigo in order.
- Erroneous page load events are displayed in yellow → orange → red in order. The erroneous page load event occurs when a browser error is encountered while loading pages.
- Drag a specific area of the chart to see the detailed information. The [Page Load Detail](#) window appears. You can see the detailed information of the load event in the dragged area. For more information, see [the following](#).

Error counts per page and message



It displays the browser errors encountered by end users while using the browser applications. The errors are displayed for each page and error message. Through this, you can quickly identify the pages where errors usually occur and the content of each error.

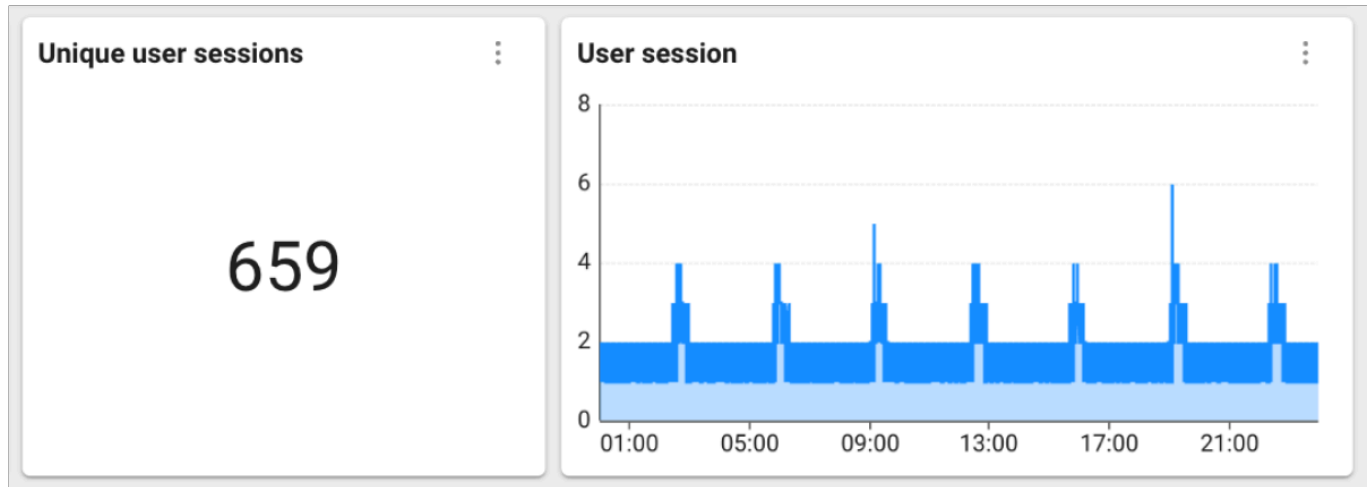
- [Page Error Count](#)

It displays the numbers of all errors on each page. If you hover your mouse over the chart, you can see the number of errors in the page group.

- [Error Counts per Error Message](#)

It displays the messages for source and AJAX errors, excluding console errors. The horizontal bar chart displays the sum of errors for each error message during the lookup period. Only the top 8 are displayed.

User session count and user sessions



You can monitor the numbers of user sessions accessing the browser applications in real time. You can see the overall usage of your browser applications through this. A sharp increase in the session count indicates an increase in the number of users. On the other hand, if the session count decreases, it is likely that many users are no longer using the application or had uncomfortable experiences.

If you use the wired session widget and other widgets together, you can see a comprehensive view of the application usage.

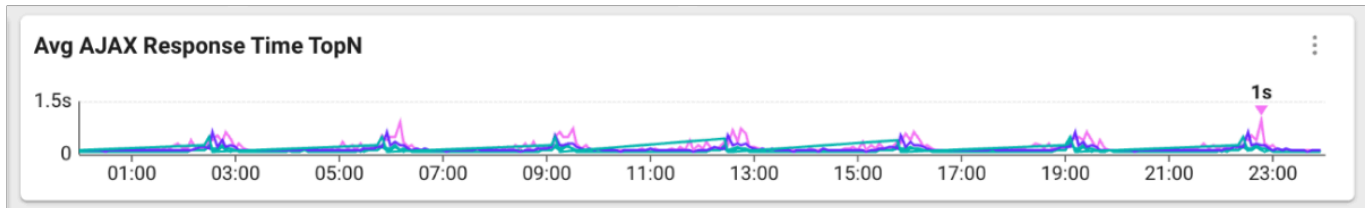
- [User Session Count](#)

It displays the total number of non-overlapping user sessions based on the lookup time.

- [User Sessions](#)

It displays the numbers of non-overlapping user sessions every 5 seconds, 5 minutes, and 1 hour. Select  on the upper right of the widget, and then select Change Metric. You can select a desired time interval.

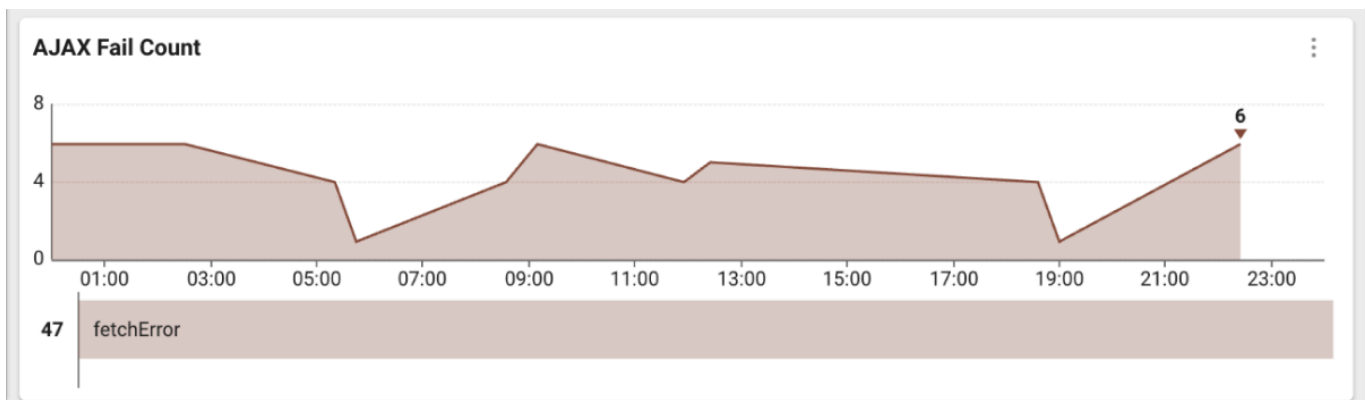
Average AJAX Response Time TopN



Browser applications frequently send and receive data for rendering via AJAX. On pages where AJAX events continue to occur after page loads, the load of AJAX can be used as an important metric for user experience to browser applications. The [Average AJAX Response Time TopN](#) widget displays the average AJAX response times when end users are using the browser applications.

- It provides charts for each AJAX request path. You can select and check the charts by the request host, request page, and overall average. Select  on the upper right of the widget, and then select Change Metric. You can change the chart data.
- By default, the top 10 data appears based on the lookup time. To change the data count, select  on the upper right of the widget and then select . If you select the Top N button on the upper right of the screen, the option to change the count appears.

AJAX failure count



The AJAX error indicates that the status code is 400 or higher or the request has been canceled because the request itself could not be executed. It displays the number of AJAX errors occurring in the browser applications

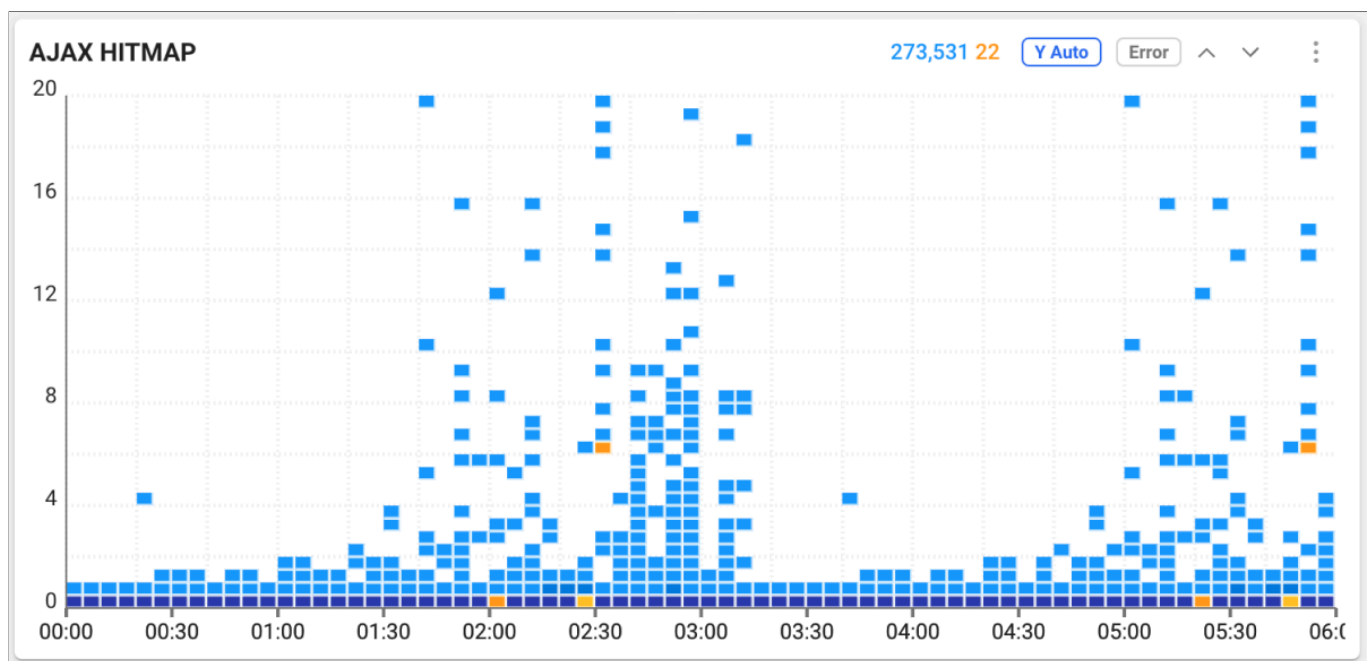
by classifying them by the fetch or XHR type.

AJAX HITMAP

AJAX requests from browser applications are provided in the form of a hitmap chart, and the detailed information on each request can be viewed in a table form. By analyzing each AJAX, you can pinpoint the cause of the performance anomaly.

! For more information about **AJAX Hitmap**, see [the following](#).

Hitmap Chart



The AJAX hitmap chart displays a distribution of the response times of AJAX requests over time.

- The horizontal axis is the AJAX end time, and the vertical axis is the AJAX load time.
- Normal AJAXs are displayed in light blue → blue → Indigo in order.
- Erroneous AJAXs are displayed in yellow → orange → red in order. Their state code is 400. To see only the

errors on the chart, select **Error** on the upper right of the widget.

- If **Y Axis Auto** is active, the maximum value of the Y axis automatically changed depending on the distribution. To fix the Y-axis, select **Y-Axis Auto** to disable it.
- Select **^** or **v** to move the chart location up or down.
- Drag a specific area of the chart to see the detailed information. To display the AJAX event list and details of the dragged area, the **AJAX Analysis** window appears.

AJAX Event Detailed Information

AJAX Analysis 1000 cases have been inquired. ⓘ

All

Column Settings

AJAX URL

Please enter your search term

Index	AJAX URL	Start time	Elapsed time	Page URL	Status code
1	http://demo.browser.monit	2024/06/20 07:19:57	15ms	http://demo.browser.monit	200
2	http://demo.browser.monit	2024/06/20 07:57:18	15ms	http://demo.browser.monit	200
3	http://demo.browser.monit	2024/06/20 07:39:28	14ms	http://demo.browser.monit	200
4	http://demo.browser.monit	2024/06/20 07:08:03	13ms	http://demo.browser.monit	200
5	http://demo.browser.monit	2024/06/20 07:48:56	13ms	http://demo.browser.monit	200
6	http://demo.browser.monit	2024/06/20 06:43:11	12ms	http://demo.browser.monit	200
7	http://demo.browser.monit	2024/06/20 07:13:16	12ms	http://demo.browser.monit	200
8	http://demo.browser.monit	2024/06/20 07:32:13	12ms	http://demo.browser.monit	200
9	http://demo.browser.monit	2024/06/20 07:34:27	12ms	http://demo.browser.monit	200
10	http://demo.browser.monit	2024/06/20 07:46:43	12ms	http://demo.browser.monit	200
11	http://demo.browser.monit	2024/06/20 08:12:21	12ms	http://demo.browser.monit	200
12	http://demo.browser.monit	2024/06/20 08:28:31	12ms	http://demo.browser.monit	200
13	http://demo.browser.monit	2024/06/20 08:36:25	12ms	http://demo.browser.monit	200
14	http://demo.browser.monit	2024/06/20 06:52:45	11ms	http://demo.browser.monit	200
15	http://demo.browser.monit	2024/06/20 07:08:15	11ms	http://demo.browser.monit	200
16	http://demo.browser.monit	2024/06/20 07:16:37	11ms	http://demo.browser.monit	200
17	http://demo.browser.monit	2024/06/20 07:19:24	11ms	http://demo.browser.monit	200
18	http://demo.browser.monit	2024/06/20 07:33:20	11ms	http://demo.browser.monit	200

Summary

User session analysis

Session Replay

AJAX Info

Browser

Device

City

Session ID

Status code

Error message

Valid bandwidth

Delivery Type

AJAX URL

Page URL

User agent

OS

Country/Region

User ID

Client IP

AJAX Abort

Valid connection type

RTT

AJAX Timing Info

Elapsed time

Start time

End time

Size

15ms

06/20 07:19:57

06/20 07:19:57

62B

Elapsed time

Waiting time

Download time

15ms

1ms

1ms

If you select each item in the **AJAX Analysis** window, the **AJAX Info** window appears on the right to display the detailed information.

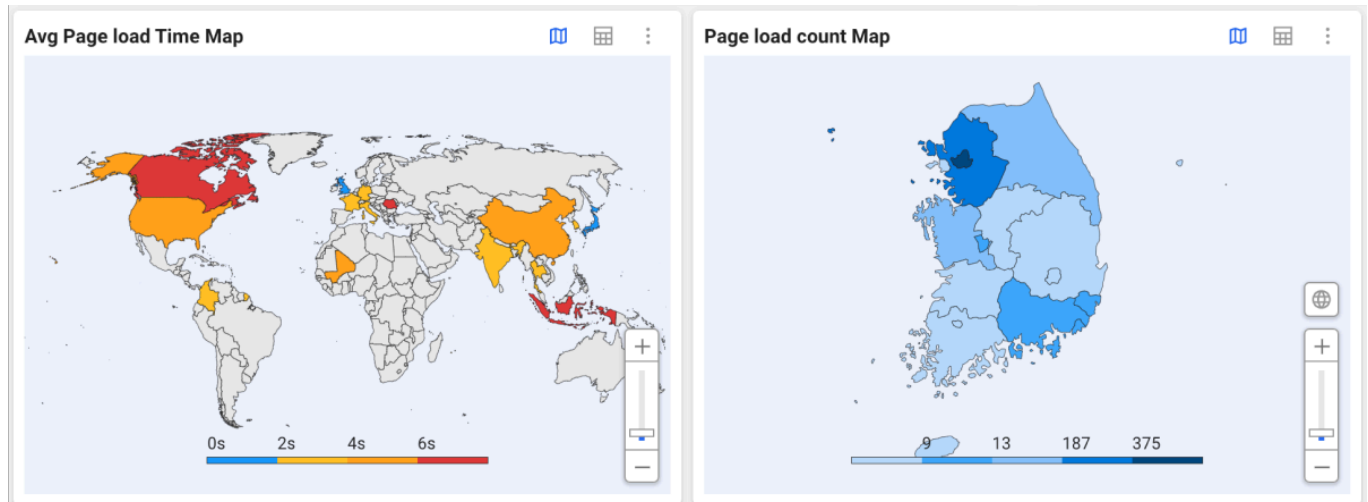
To see only the AJAX list that has errors, on the upper left of the **AJAX Info** window, select from the **Error All** choose list.

Each AJAX has the following information:

- AJAX URL
- URL of the page that requested AJAX
- Browser, device, and OS information
- Error message

- Regional information
- AJAX detailed timing

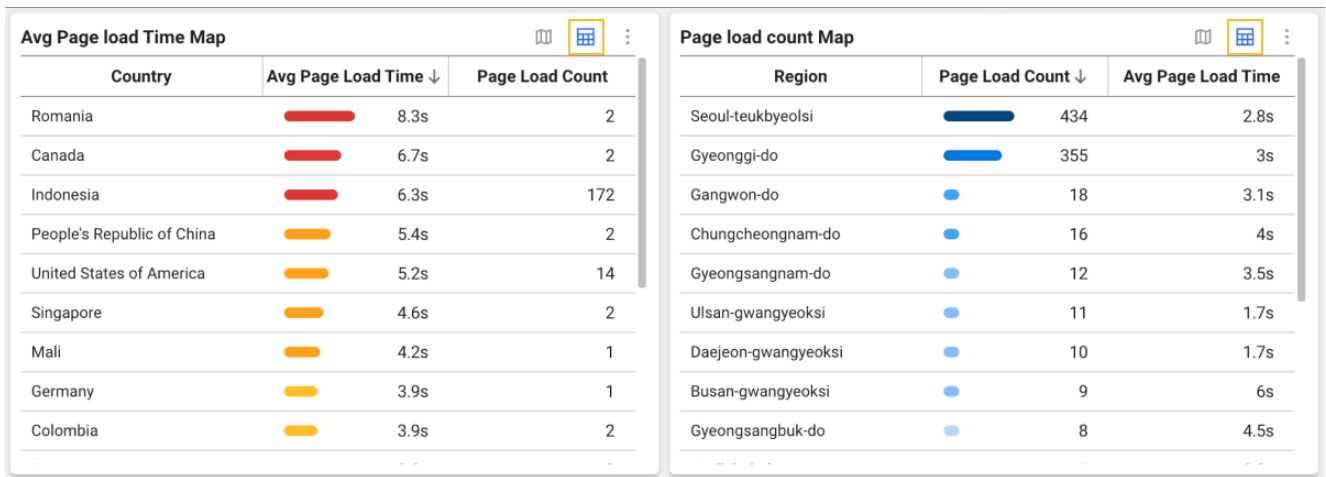
Page load times and counts per region



It displays the page load times and counts by region for browser applications in the map chart and table. You can easily compare the website performances for each region. The page load time displays the legend every 2 seconds, and the page load count displays the legend at 50, 75, 90, and 95 percentile intervals.

The widget displays the countries with the highest page load based on the search time.

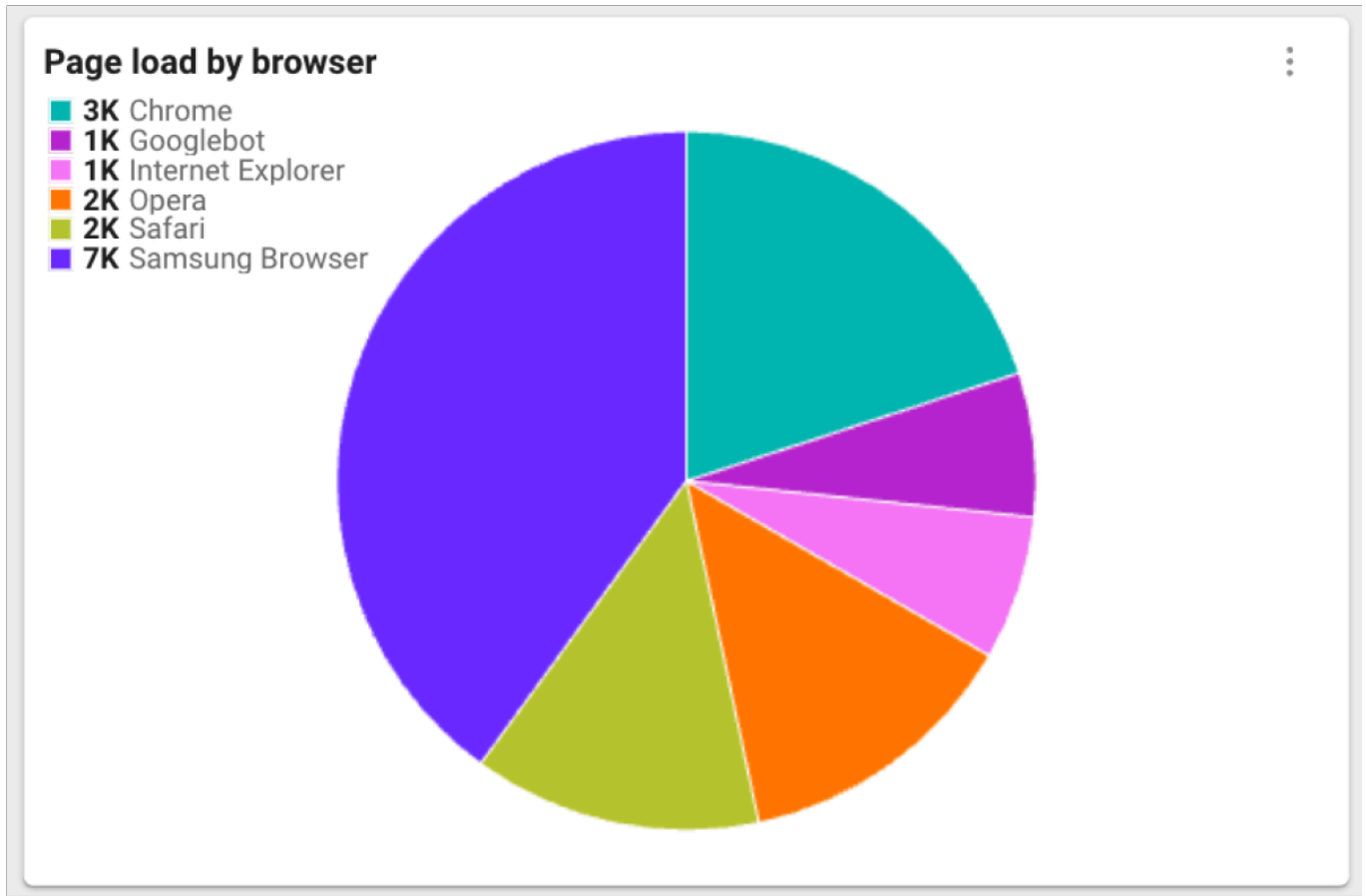
- You can scroll the mouse wheel or use the slider to zoom in or out on the map.
- To view the world map, select . You can select a country to see its regional data.
- To see the data in a table format, select . To return to the map format, select .



- In case of regional maps, data appears based on the upper administrative district. (Korea: province or metropolitan city)
- You can view the map and the series (trend) chart together in a single widget.
- You can toggle the display metric between **Elapsed time** and **Count**. **Elapsed time** shows how long the page load took by region, and **Count** shows the number of page loads by region.

ⓘ This site or product includes IP2Location LITE data available from <https://lite.ip2location.com>.

Page Load By Browser



It provides the page load times and page load counts for browser applications. You can easily compare the performances of different browsers. To change the data to [Page Load Time](#) or [Page Load Count](#), select ⋮ on the upper right of the widget, and then select Metrics.

Core Web Vitals Metric

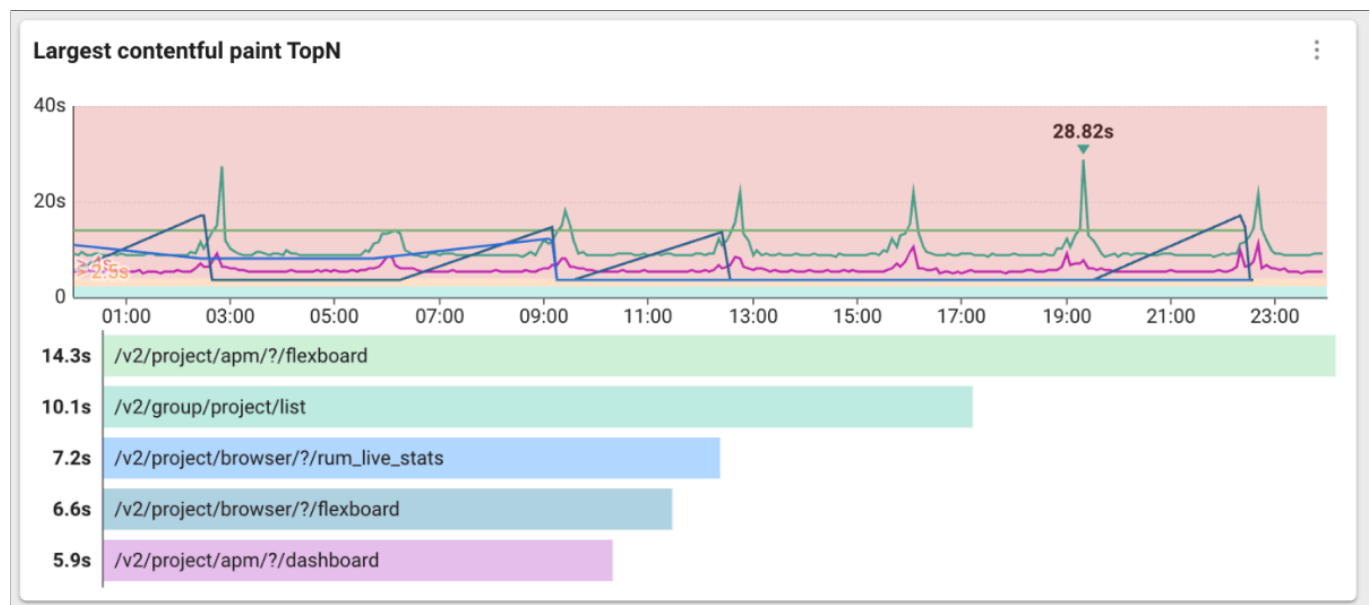
Core web vitals are three important metrics related to the website performance among various metrics that affect user experience as defined by Google. Metrics are displayed differently depending on the website loading and interaction with users.

- **Largest Contentful Paint:** A metric that measures the time loaded for the largest content element on your website.
- **Interaction to Next Paint:** The time taken from a user's interaction with the page until the next screen is painted.
- **Cumulative Layout Shift:** A metric of how often the layout changes while the website is loaded.

These measurements play an important role in evaluating and improving the website performance. Understanding and measuring the core web vitals metrics is very useful to improve user experiences and optimizing the website performance.

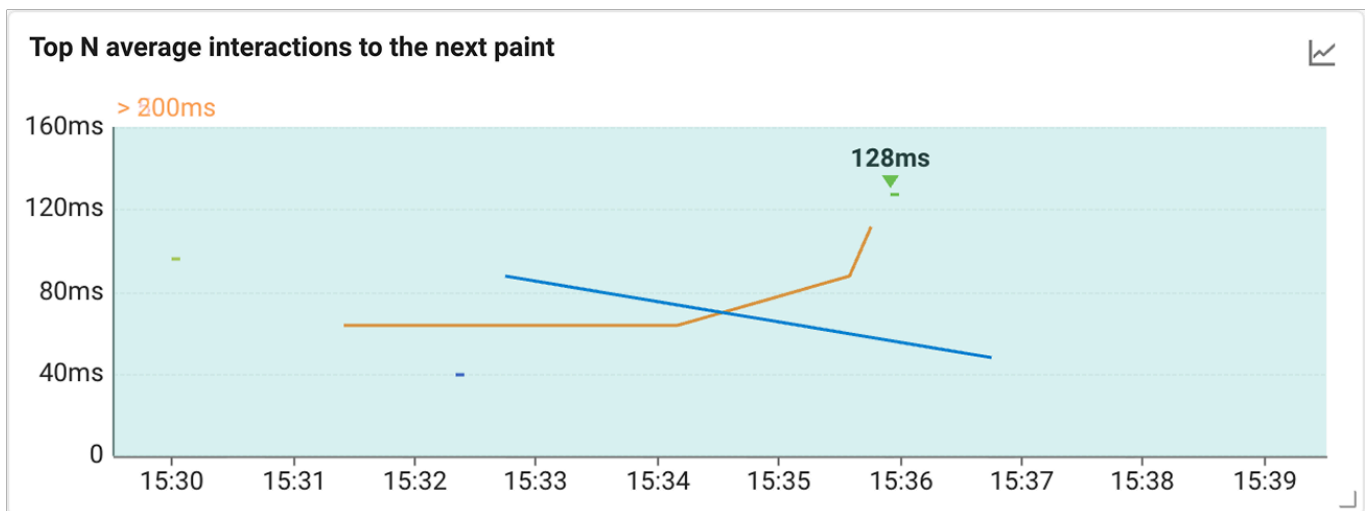
⚠ Key change: The First Input Delay (FID) metric in this section has been replaced with Interaction to Next Paint (INP). The overview list above and the subheading below have been updated accordingly.

Largest Contentful Paint



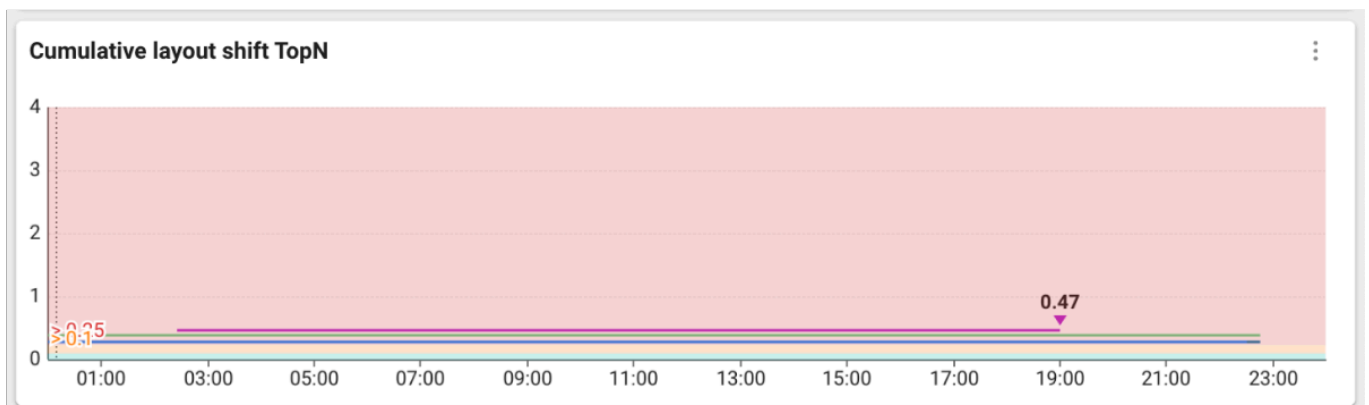
It displays the average maximum contentful paint time for each page group. It displays the averages of maximum contentful paint times for the top 8 groups of each page group during the lookup time in a horizontal bar chart.

Interaction to Next Paint



It displays the average Interaction to Next Paint time for each page group. If you hover your mouse over the chart, the time value for the page group appears.

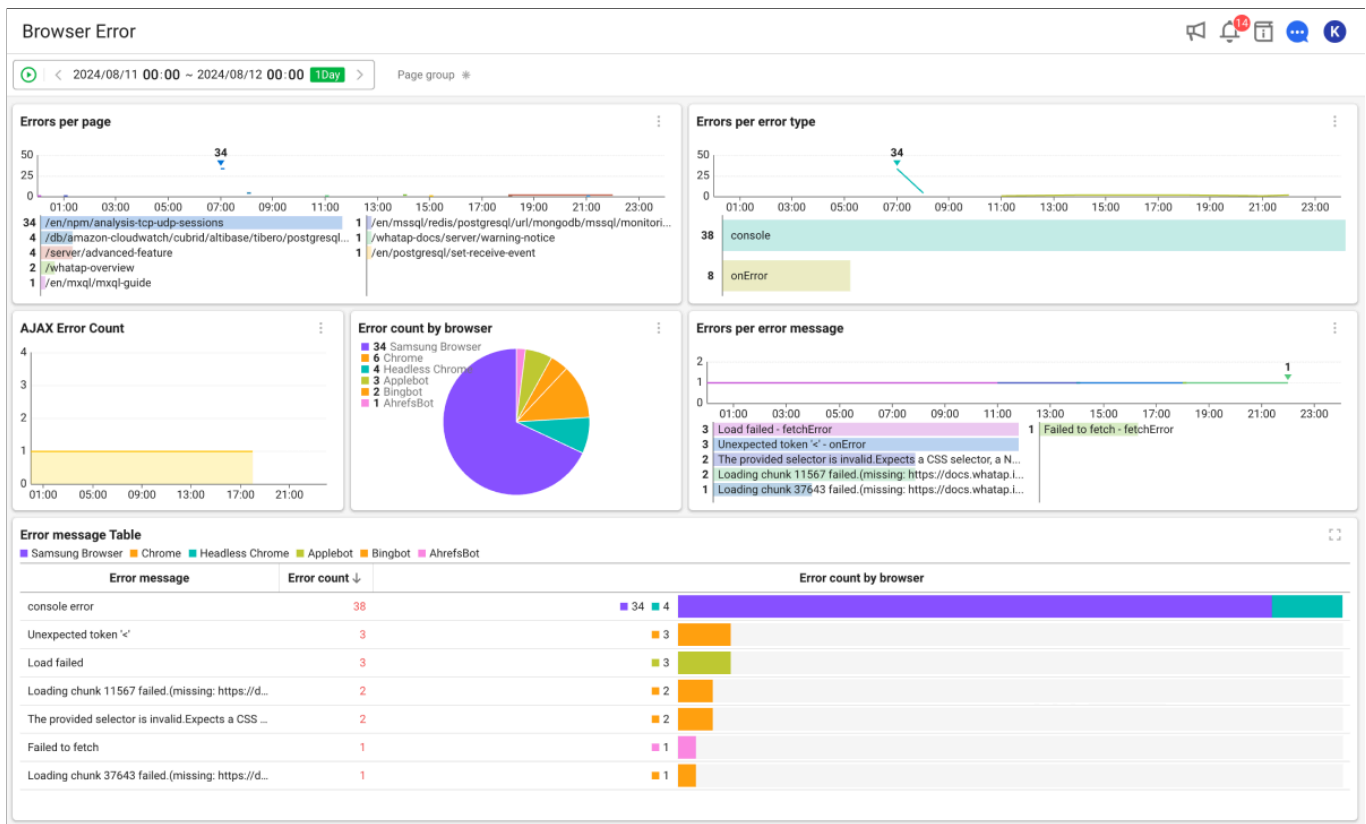
Cumulative Layout Shift



It displays the cumulative layout values for each page group. If you hover your mouse over the chart, the value for the page group appears.

Browser Error

Home > Select Project > **Dashboard** > **Browser Error**



The **Browser Error** dashboard monitors and visualizes errors of the browser application and displays them in real time. Through the dashboard, you can quickly identify errors, find their causes, and fix them. It provides various data such as AJAX error counts for each page and error type and error counts for each browser and error message.

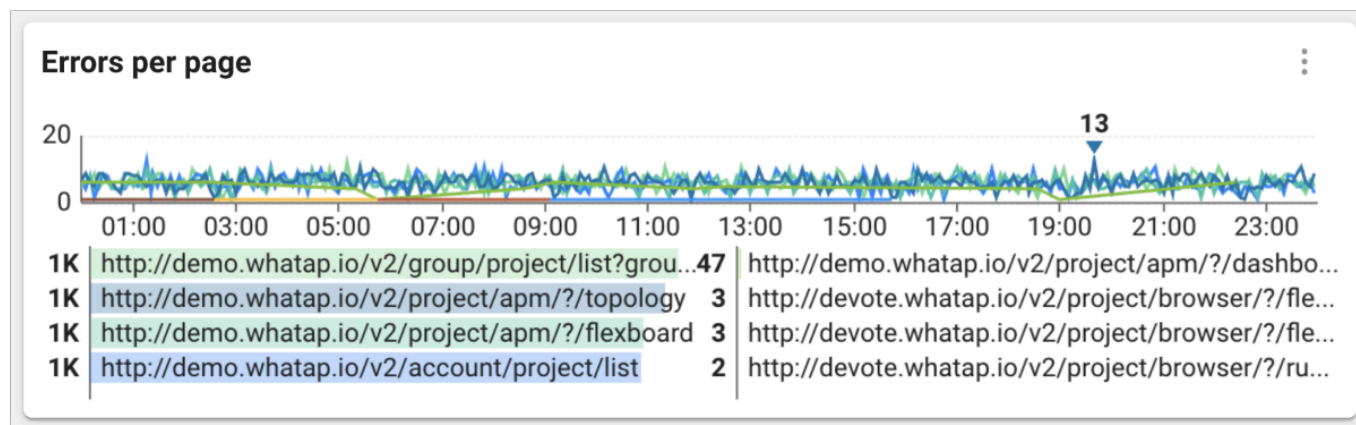
- On the upper right of each widget, select . Then you can see the widget data in full screen.
- If you select on the upper right of the widget, the chart configuration screen appears. You can set the options to display on the chart.
- Depending on the screen size of the web browser, the icon shape on the upper right of a widget may



differ. If you select , the additional icons appear.

- For more information about the widget configuration, see [the following](#).

Errors per page



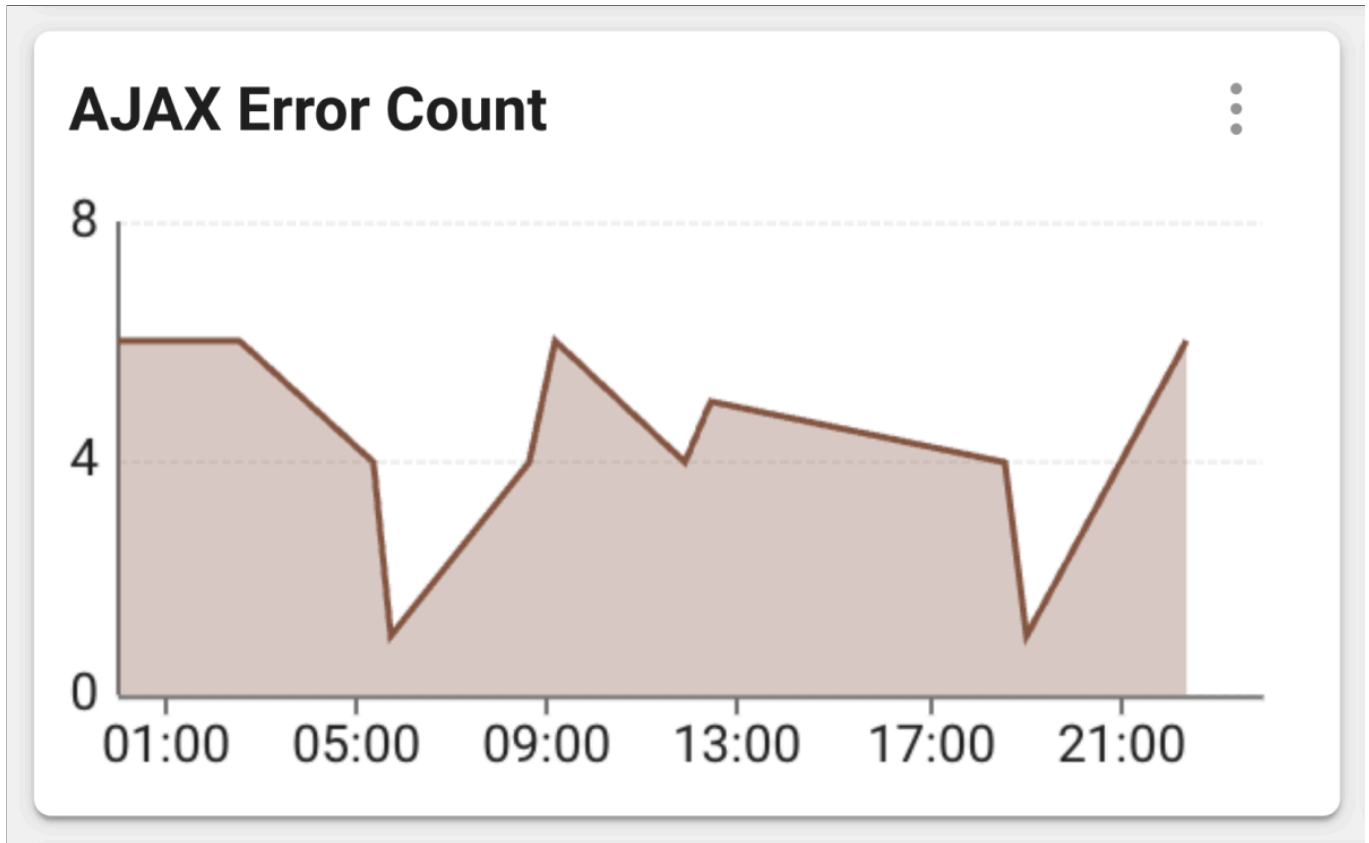
It displays the numbers of all errors on each page. Hover your mouse over the chart to see the number of errors in the page group with a tooltip. The horizontal bar chart is the sum of error counts per page during the lookup time, with only the top 8 errors displayed.

Errors per error type



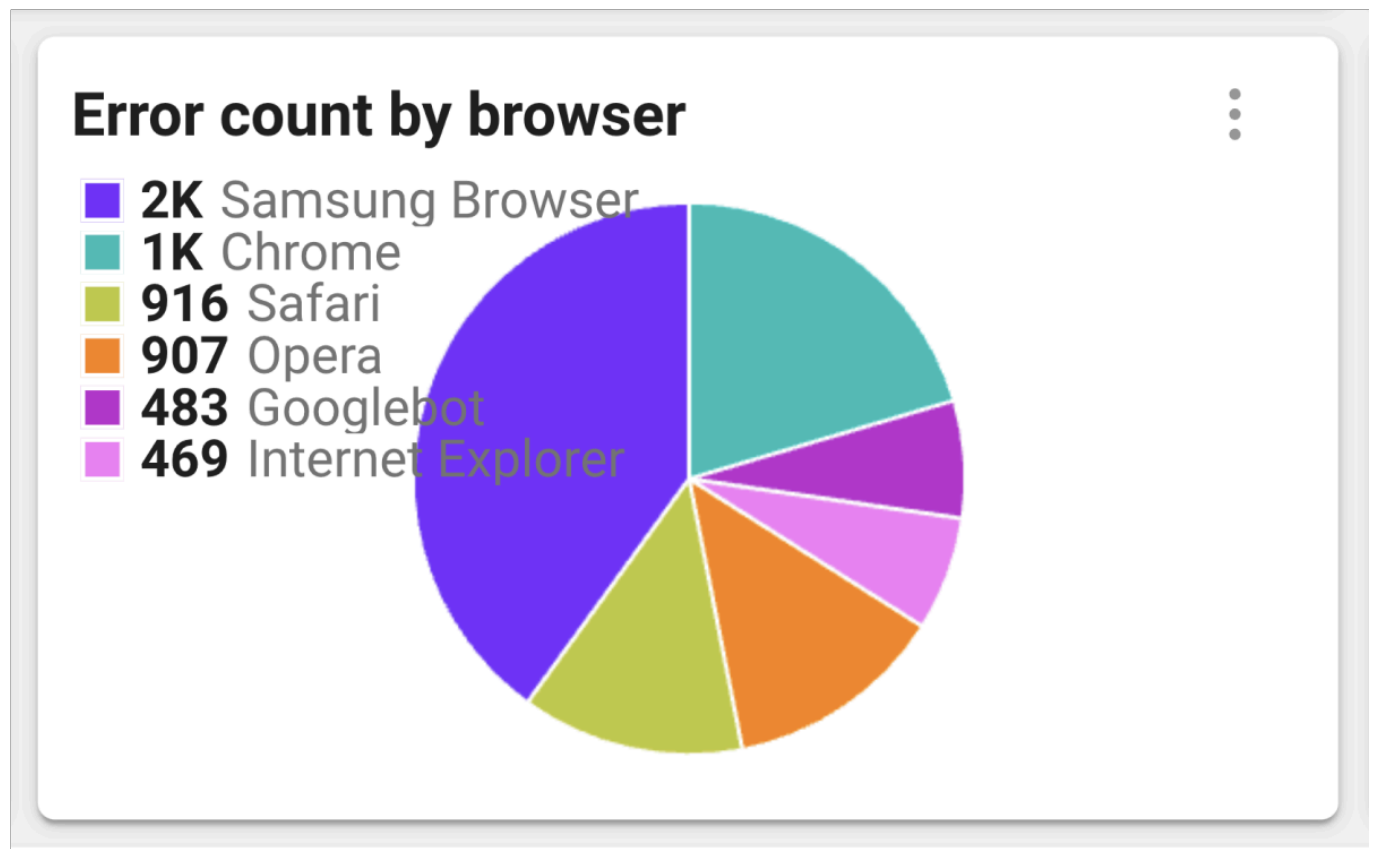
It displays the console and source errors. The horizontal bar chart is the sum of error counts per error type during the lookup time, with only the top 8 errors displayed.

AJAX error count



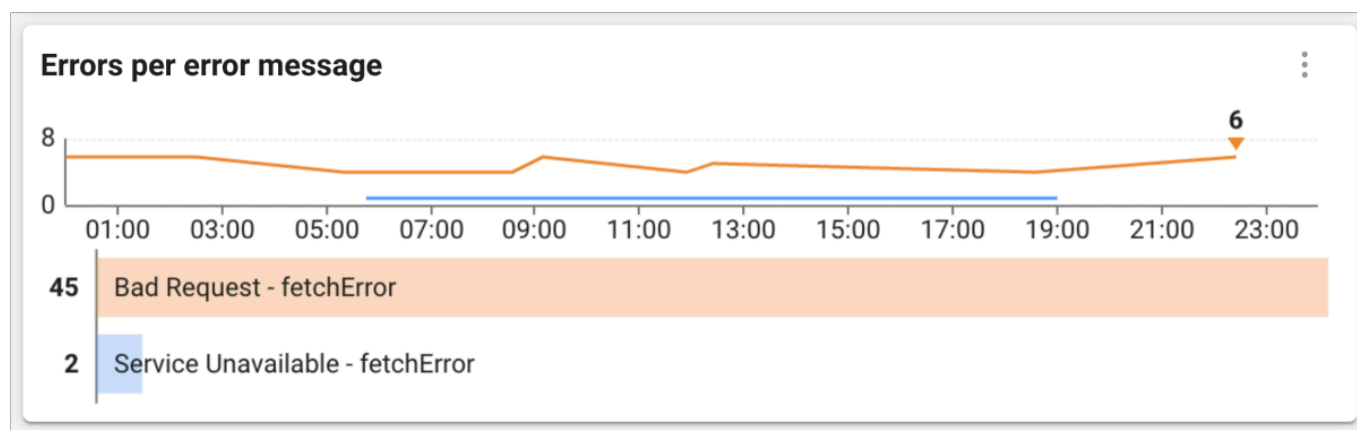
It displays the number of AJAX errors occurring in the browser applications by classifying them by the fetch or XHR type. The AJAX error indicates that the status code is 400 or higher or the request has been canceled because the request itself could not be executed.

Error count by browser



It displays the error counts for each browser in a pie chart. You can easily figure out which errors your browser is experiencing most often. If a specific browser has more errors than other browsers, the developer can proceed with optimization for the browser.

Errors per error message



It displays the messages for source and AJAX errors, excluding console errors. The horizontal bar chart is the sum of error counts per message during the lookup time, with only the top 8 errors displayed.

Error message table

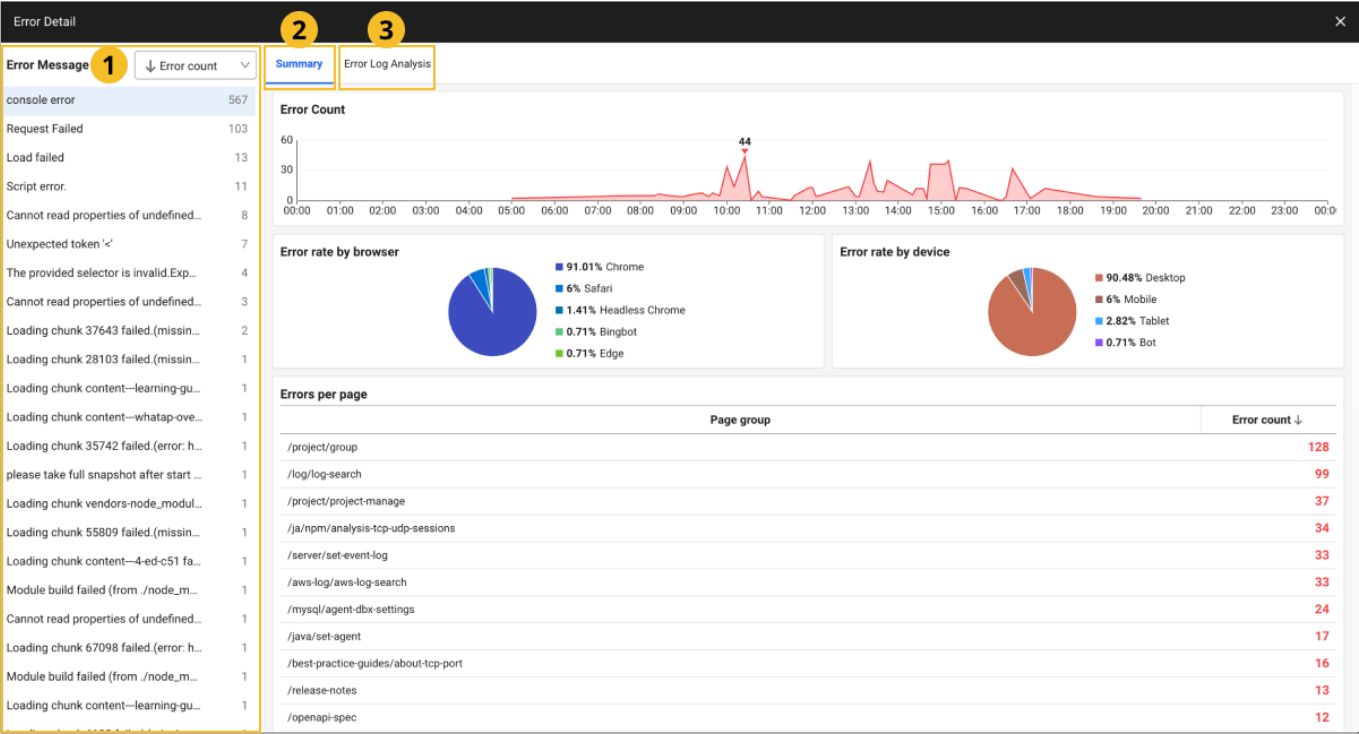
Error message	Error count ↓	Error count by browser
console error	38	34 4
Unexpected token '<'	3	3
Load failed	3	3
Loading chunk 11567 failed.(missing: https://d...	2	2
The provided selector is invalid.Expects a CSS ...	2	2
Failed to fetch	1	1
Loading chunk 37643 failed.(missing: https://d...	1	1

It displays the error counts by error message and the error counts by browser in a table. You can find which browsers are causing most errors and use them to troubleshoot. It is also useful for identifying the root causes of errors. If a new error occurs, you can quickly respond by seeing this table.

Checking the error message details

If you select an error message from the **Error message Table** list, you can see the statistics details based on the

error message.



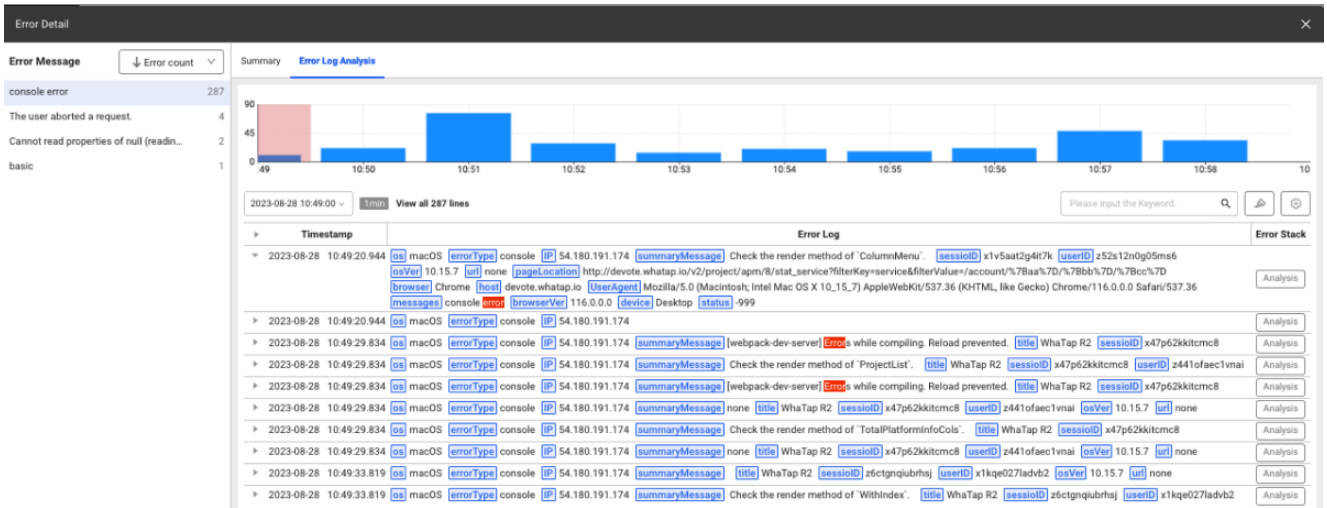
- **1 Error Message**

You can check the list of error messages based on the inquiry time. If each error message is selected, you can see the detailed information.

- **2 Summary**

Through the graph chart, you can see the **Error Count**, **Error rate by browser**, **Error rate by device**, and **Errors per page** for each time zone.

- **3 Error Log Analysis**



You can see the error logs by the time zone. If you select **Analysis** for the **Error Stack**, the error stack information window appears for the selected error log. For more information, see [the following](#).

Page Load

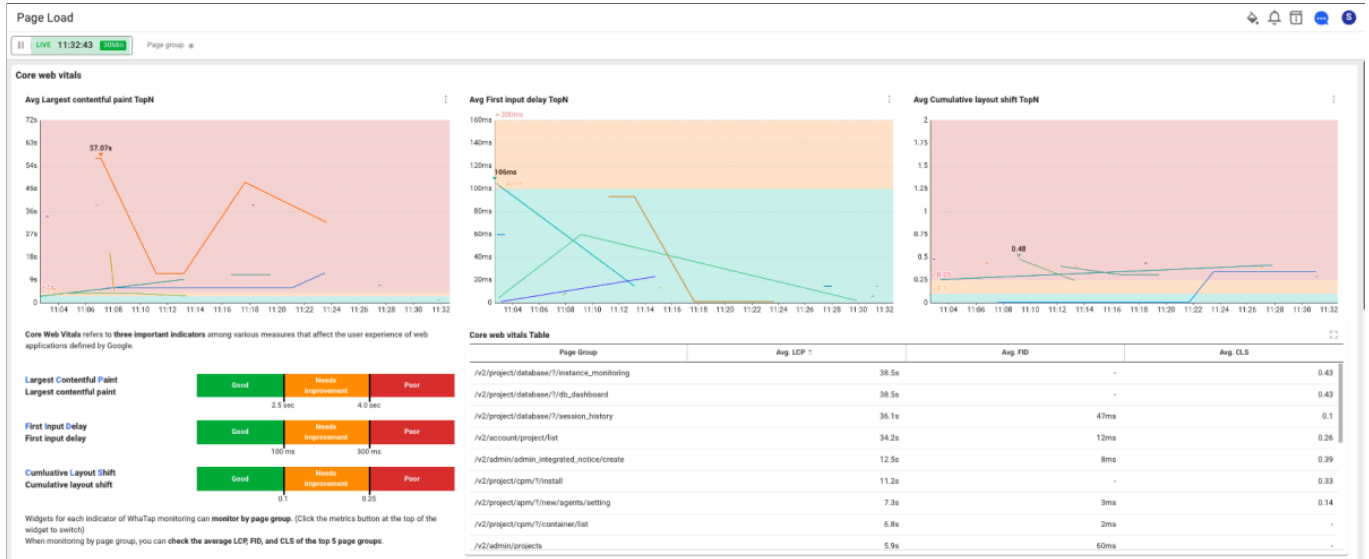
Home > Select Project > [Dashboard](#) > [Page Load](#)

In the [Page Load](#) dashboard, you can monitor Core Web Vitals (CWV) related to page load and performance data such as resources and AJAX in real time. This can help you understand the factors that affect the web application experience and derive improvements.

- On the upper right of each widget, select . Then you can see the widget data in full screen.
- If you select  on the upper right of the widget, the chart configuration screen appears. You can set the options to display on the chart.
- Depending on the screen size of the web browser, the icon shape on the upper right of a widget may differ. If you select , the additional icons appear.
- For more information about the widget configuration, see [the following](#).

CWV-related widgets

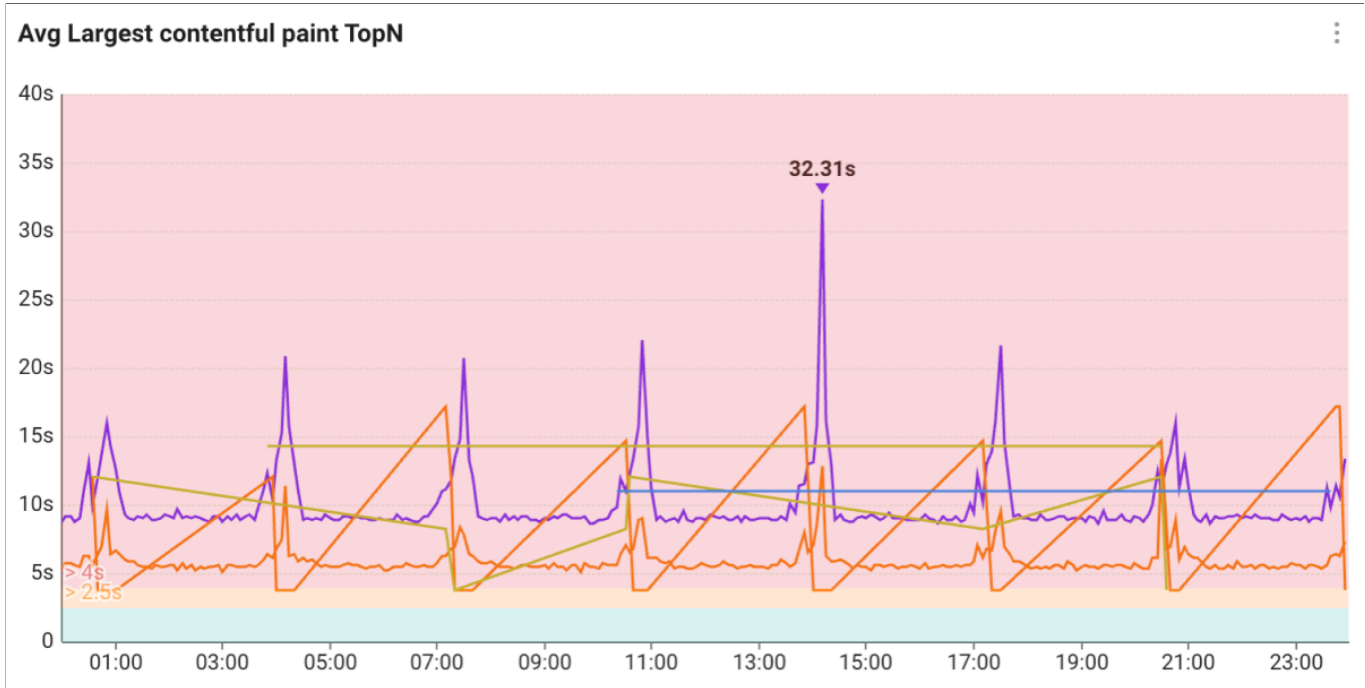
Core Web Vitals (CWV) means three metrics that Google considers important among various measures that affect the end-user experience.



! Through the Google [Core Web Vitals Report](#), you can see how the page performance affects the end user experience.

Average Largest contentful paint

Largest Contentful Paint (LCP) reports the rendering time of the largest image or text block in the browser viewport based on the first page load. It is an important user-centric metric that allows you to measure page load times as perceived by end users. A fast LCP helps end users perceive that the page is available.



To provide an optimized user experience, your web application must have an LCP of **2.5 seconds** or less. The [Avg Largest contentful paint TopN](#) widget allows you to see the average LCP based on the page group or all pages. If you see the chart by page group, it provides you the average LCP for the top 5 page groups.

Average Interaction to Next Paint

⚠ Key change: In the Core Web Vitals time-series widget and legend, the FID (First Input Delay) metric has been replaced with INP (Interaction to Next Paint) (the threshold also changes from FID's 100 ms/300 ms to INP's 200 ms/500 ms). However, the [Core Web Vitals Table](#) below keeps the FID column and adds an INP column, so the metrics shown in the time-series/legend differ from those in the table.

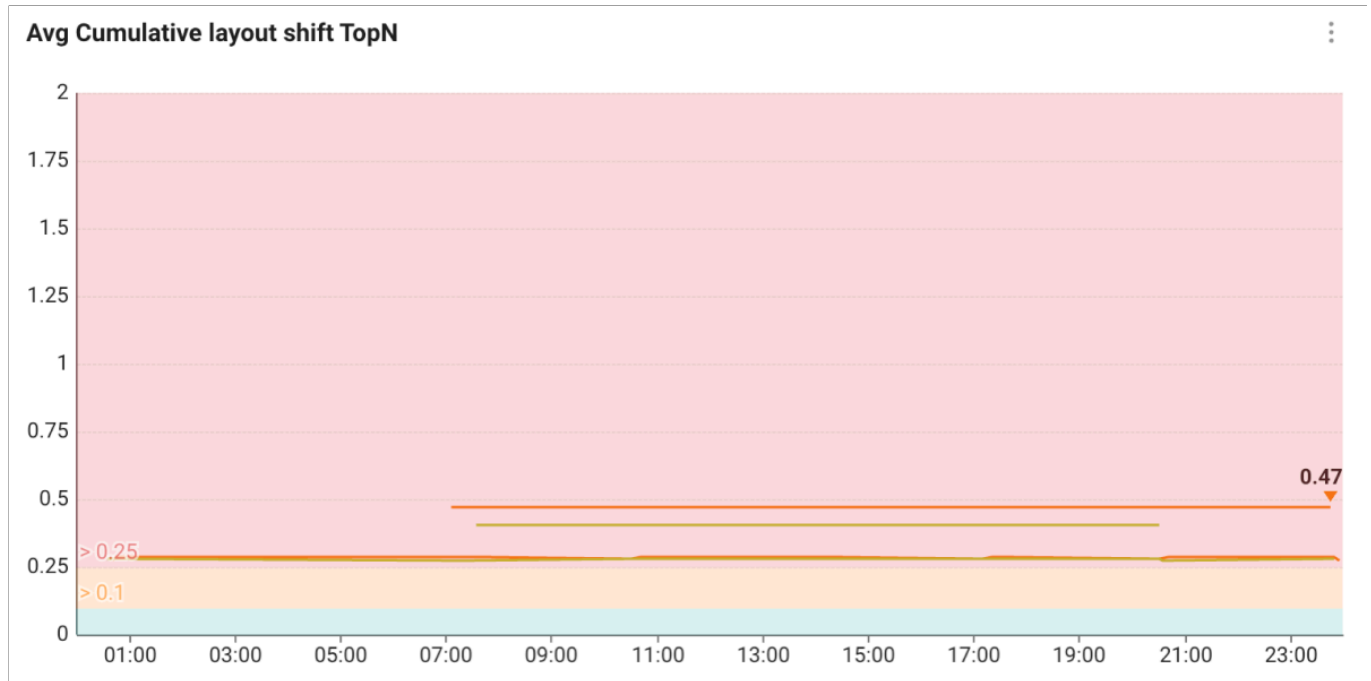
Interaction to Next Paint (INP) measures the time from when an end user interacts with a web application page (click a link, tap a button, or use the custom Javascript-based controls) until the browser paints the next frame in response to that interaction. It is a metric of the user's experience when interacting with an unresponsive page. A low INP means users perceive that the page can be used normally.

To provide an optimized user experience, your web application must have an INP of **200 ms** or less (500 ms or more is considered poor). The [Avg Interaction to Next Paint TopN](#) widget allows you to see the average INP

based on the page group or all pages. If you see the chart by page group, it provides you the average INP for the top 5 page groups.

Average Cumulative layout shift

Cumulative Layout Shift (CLS) is an important user-driven metric for measuring visual stability by quantifying the frequency with which end users experience unexpected layout shifts. A low CLS helps ensure a good user experience.



CLS must be kept below 0.1 to provide an optimized user experience. The [Avg Cumulative layout shift TopN](#) widget allows you to see the average CLS based on the page group or all pages. If you see the chart by page group, it provides you the average CLS for the top 5 page groups.

Core Web Vitals Table

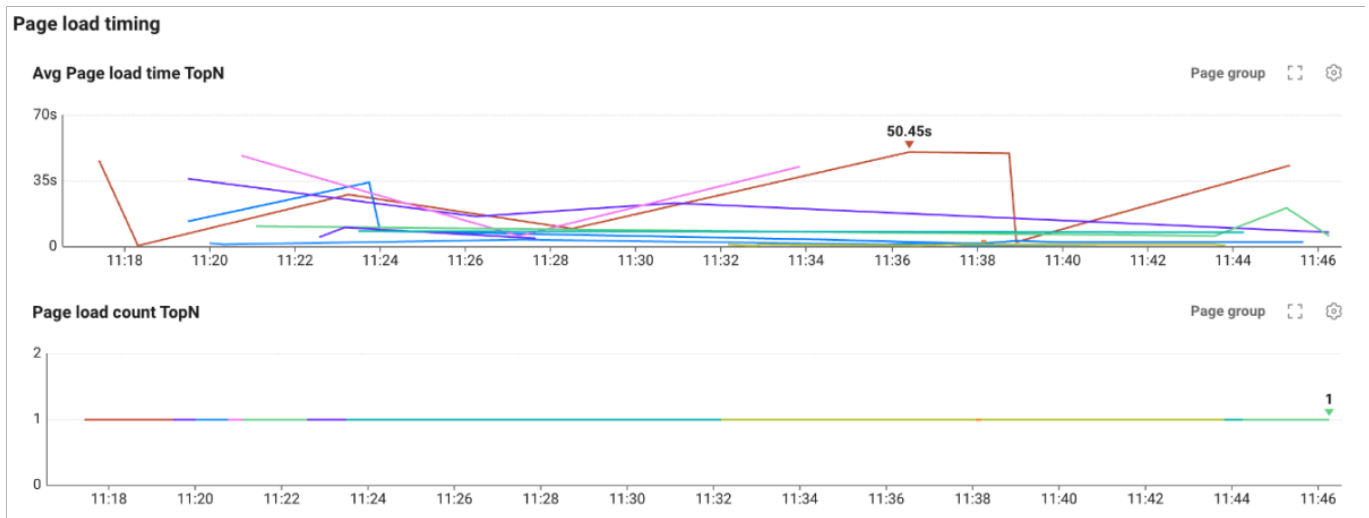
Core web vitals Table			
Page Group	Avg. LCP	Avg. FID	Avg. CLS
/v2/project/database/?/session_history	42.6s	1ms	0.27
/v2/project/database/?/instance_monitoring	38.5s	-	0.43
/v2/project/cpm/?/container/list	25.7s	1ms	-
/v2/admin/admin_integrated_notice/create	23.9s	8ms	0.39
/v2/project/cpm/?/install	11.2s	-	0.3
/v2/project/cpm/?/containerMap	11.2s	13ms	-
/v2/account/billing	9.2s	1ms	0.74
/v2/project/apm/?/new/agents/setting	8.8s	3ms	0.23
/v2/admin/regions	3.8s	-	0.29

Based on the lookup time, for each [page group](#), the average LCP ([Avg. LCP](#)), average FID ([Avg. FID](#)), average INP ([Avg. INP](#)), average CLS ([Avg. CLS](#)) can be checked in the table. You can click the table header column to sort by column.

Page Load Timing-related widget

When an end user accesses a web application, the page load time and load count allow you to determine when loading a page takes so long and which pages are causing load.

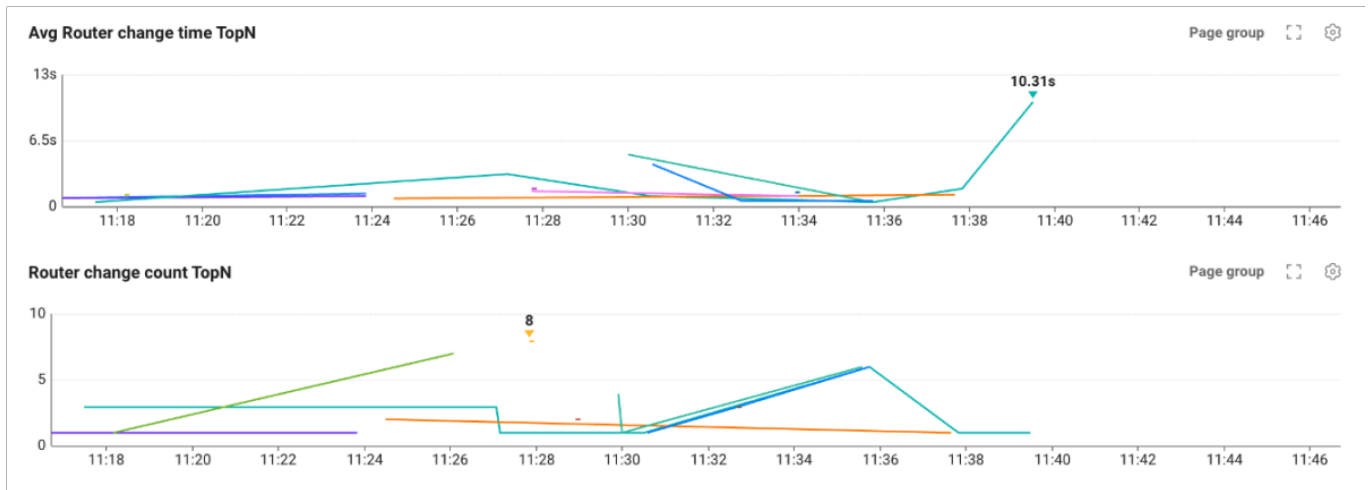
Average Page Load Time / Page Load Count



If an end user accesses the page of the web application, you can see the [Avg Page load time TopN](#) and [Page load count TopN](#) based on the real time or selected time. The load times and load counts are provided for each page group by default.

- When viewed by the page group, it provides the average page load time and page load count for the top 10 page groups.
- If you hover your mouse over the widget chart, you can see the page groups, average page load times, and page load counts in the form of tooltip.

Average Router Change Time / Router Change Count



When the end users switch to a web application page that consists of single page applications (SPA), you can see values for [Avg Router change time TopN](#) and [Router Change Count TopN](#) based on the real time or the selected time. The router change times and change counts are provided for each page group by default.

- When viewed by the page group, it provides the average router change time and change count for the top 10 page groups.
- If you hover your mouse over the widget chart, the page group, average router change time, and change count are provided in the form of tooltip.

Page load table

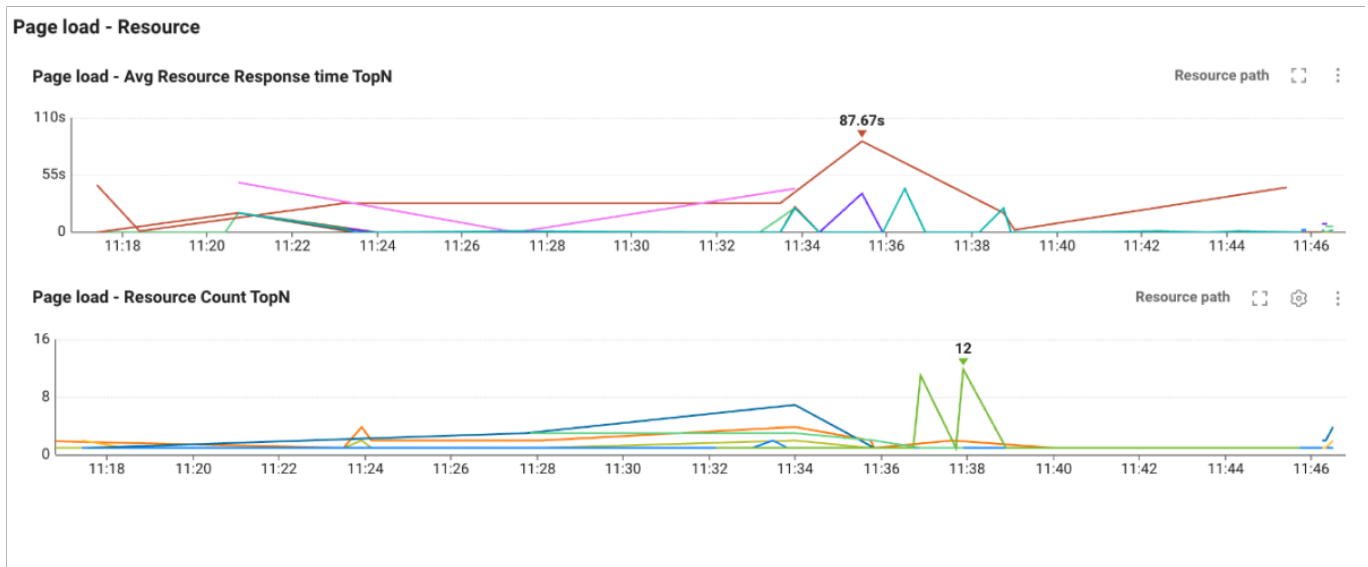
Page load Table						⋮
Page Group	Load Count	 Load Time ↕	Avg. Frontend Ti...	Avg. Backend Ti...	Avg. onload Eve...	
/v2/project/browse r/?/flexboard	1	16.3s	7.2s	8.5s	16.3s	
/v2/project/browse r/?/rum_live_stats	1	14.2s	4.6s	8.9s	14.1s	
/v2/account/projec t/list	203	9s	7.8s	224ms	9s	
/v2/project/apm/?/ topology	200	4.5s	3.8s	238ms	4.4s	
/v2/group/project/l ist	201	4.5s	3.6s	89ms	4.4s	
/v2/project/apm/?/ flexboard	196	3.4s	2.9s	94ms	3.4s	

For each page group ([Page Group](#)), the page load count ([Load count](#)), average load time ([Avg. Load Time](#)), average back-end process time ([Avg. Backend Time](#)), average frontend process time ([Avg. Frontend Time](#)) can be checked in the table. You can click the table header column to sort by column.

Resource-related widget

You can identify resources that affect the page load. It is recommended that the load time of the resource not exceed 800 ms.

Average Resource Response Time / Resource Count



It provides a chart for the average resource response time and resource count that occurred during loading pages. Select [Resource Path](#) on the upper right of the widget to check the data based on [Resource Host](#) and [All](#) resources.

- When viewed based on the [Resource Path](#) and [Resource Host](#), the average response time and count for the top 10 resources are provided.
- If you hover your mouse over the widget chart, the average resource response time and resource count are provided in the form of tooltip.

Resource Table

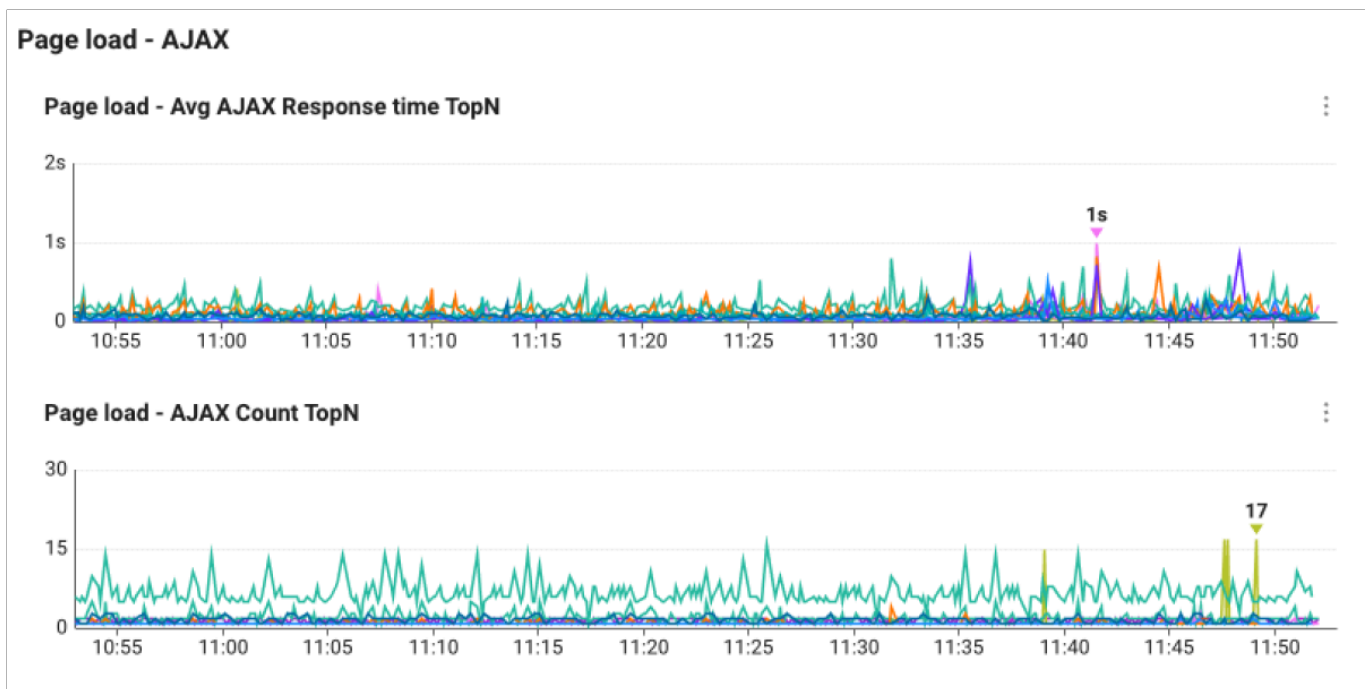
Page load - Resource Table							
Resource ...	Resource ...	... Time ↕	Avg. Dns ...	Avg. Con...	Avg. TTF...	Avg. Dow...	Size
/v2/project/browser/401/r...	1	14.1s	1ms	5ms	199ms	15ms	29.6k
/v2/account/project/list	58	7.3s	-	-	31ms	3ms	29.5k
/v2/project/apm/13/topol...	63	4.3s	1ms	44ms	63ms	3ms	29.6k
/v2/group/project/list	69	3.8s	1ms	43ms	44ms	3ms	29.5k
/v2/project/apm/185/flex...	68	2.9s	1ms	39ms	60ms	3ms	29.6k
/dist/2edca1bf8a77b8e...	1	2.8s	-	-	279ms	2.5s	50.6k
/dist/bab1cfc3e81b59aa4...	228	1.4s	1ms	10ms	103ms	1.3s	69.9k
/dist/553ed0c...	...	...	...	...	...	...	...

In the [Page Load - Resource Table](#) widget, you can see a chart for the path reference of the resource URL and the requested page. You can search data by selecting [Resource Path](#) on the upper right of the widget and switching based on the [page group](#).

AJAX-related widget

You can identify AJAXs that affect the page load. When the rendering data is provided through AJAX, the screen rendering may take long if the response is slow.

Avg AJAX Response Time / AJAX Count



It provides a chart for the average AJAX response time and AJAX count that occurred during loading pages. Select [AHAX Path](#) on the upper right of the widget to check the data based on [AJAX Host](#) and [All](#) AJAX.

- When viewed based on the [AJAX Path](#) and [AJAX Host](#), the average response time and count for the top 10 AJAX cases are provided.
- If you hover your mouse over the widget chart, the average AJAX response time and AJAX count are provided in the form of tooltip.

AJAX Table

Page load - AJAX Table							
AJAX Path	...AX Time ↕	2xx Count	3xx Count	4xx Count	5xx Count	Fail Count	
/yard/api/flush	216ms	2K	0	0	0	0	
/yard/api/v2/merge	133ms	376	0	0	0	0	
/account/api/v4/project/401	109ms	3	0	0	0	0	
/management/api/v4/organiz...	98ms	1K	0	0	0	0	
/account/api/v4/project/13	82ms	405	0	0	0	0	
/account/detail/public	81ms	1K	0	0	0	0	
/account/api/v4/project/list	77ms	23	0	0	0	0	
/account/api/v4/project/135	76ms	613	0	0	0	0	

In the [Page Load - AJAX Table](#) widget, you can see a chart for the path reference of the AJAX URL and the requested page. You can search data by selecting [AJAX Path](#) on the upper right of the widget and switching based on the [page group](#).

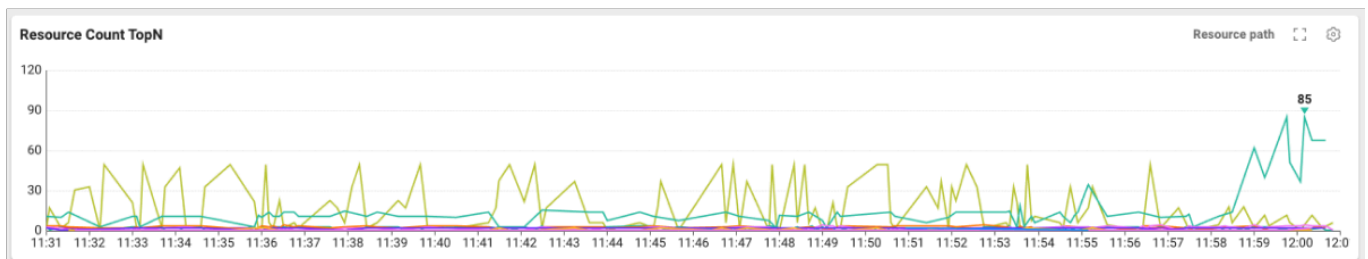
Resources

Home > Select Project > [Dashboard](#) > [Resource](#)

Widget charts are created by reprocessing resource data loaded by end users while using the web application. [Resource](#) dashboard is a real-time statistical data dashboard composed of these widgets. The load time of resources is an important user experience metric for web applications. The faster the resources are loaded, the more end users perceive the page to be available. The load time for resources is less than **800 ms** that provides a better user experience. For more information about the resource load time, see [the following](#).

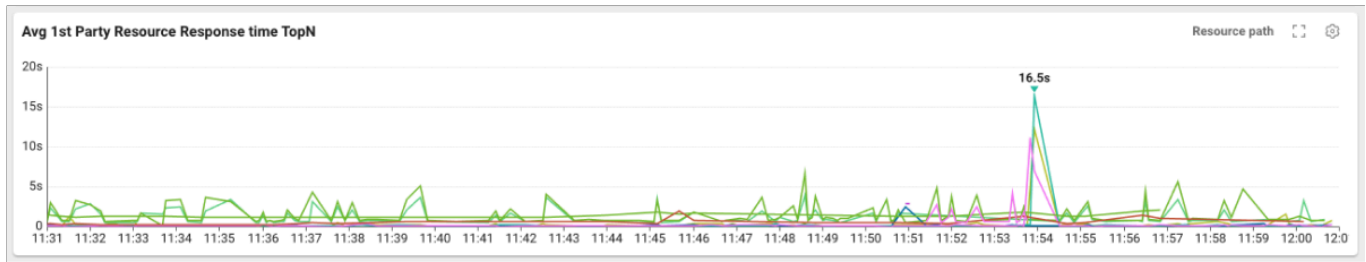
- On the upper right of each widget, select . Then you can see the widget data in full screen.
- If you select  on the upper right of the widget, the chart configuration screen appears. You can set the options to display on the chart.
- Depending on the screen size of the web browser, the icon shape on the upper right of a widget may differ. If you select , the additional icons appear.
- For more information about the widget configuration, see [the following](#).

Resource Count



You can check the resource load count in real time. On the upper right of the widget, select  and then set the time zone to see the resource load count. To change the data query criteria to [Resource Host](#) or [All](#) resources, select [Resource Path](#) on the upper right of the widget.

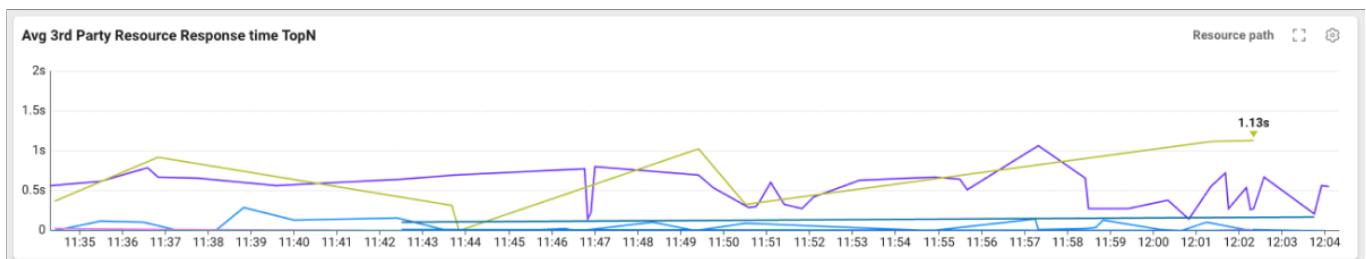
Average 1st Party Resource Response Time



This widget analyzes the resource host and provides the average response time of resources called by the 1st party with a chart.

- To change the data query criteria to [Resource Host](#) or [All](#) resources, select [Resource Path](#) on the upper right of the widget.
- When viewed based on the [Resource Path](#) and [Resource Host](#), the average resource response time and count for the top 10 resources are provided.
- You can identify resources that take a long average response time by specific paths and hosts.
- On the upper right of the widget, select  and then set the time zone to see the resource response time.

Average 3rd Party Resource Response Time



This widget analyzes the resource host and provides the average response time of resources called by the 3rd party with a chart.

- To change the data query criteria to [Resource Host](#) or [All](#) resources, select [Resource Path](#) on the upper right of the widget.
- When viewed based on the [Resource Path](#) and [Resource Host](#), the average resource response time and count

for the top 10 resources are provided.

- You can identify resources that take a long average response time by specific paths and hosts.
- On the upper right of the widget, select  and then set the time zone to see the resource response time.

Resource Table

Resource Host	Resource Count	... Resource Time 	Avg. Dns Time	Avg. Connection Time	Avg. TTFB Time	Avg. Download Time	Type
demo.whatap.io	274	206ms	1ms	25ms	41ms	149ms	image
demo.whatap.io	702	198ms	1ms	16ms	41ms	17ms	css
demo.whatap.io	5K	173ms	1ms	20ms	120ms	1ms	fetch
devote.whatap.io	17	164ms	-	1ms	95ms	25ms	css
demo.whatap.io	1K	137ms	1ms	21ms	30ms	73ms	font
devote.whatap.io	686	98ms	-	1ms	0ms	1ms	fetch
devote.whatap.io	9	81ms	-	1ms	70ms	1ms	font
cdn.channel.io	1	38ms	-	-	-	-	script
devote.whatap.io	1	14ms	-	-	1ms	1ms	image
demo.whatap.io	293	8ms	-	-	-	-	xhr
devote.whatap.io	4	1ms	-	-	-	-	xhr

You can identify resources that take longer to respond and see where the delay occurs. To change the data query criteria to [Resource Host](#) or [All](#) resources, select [Resource Path](#) on the upper right of the widget.

- [Avg. Dns Time](#): Average DNS lookup time for resources.
- [Avg. Connection Time](#): Average time taken to connect the server providing resources.
- [Avg. TTFB Time](#): Average time from connecting to the server to receiving the first byte of the resource.
- [Avg. Download Time](#): Average elapsed time to download resources.
- [type](#): Resource type.

AJAX

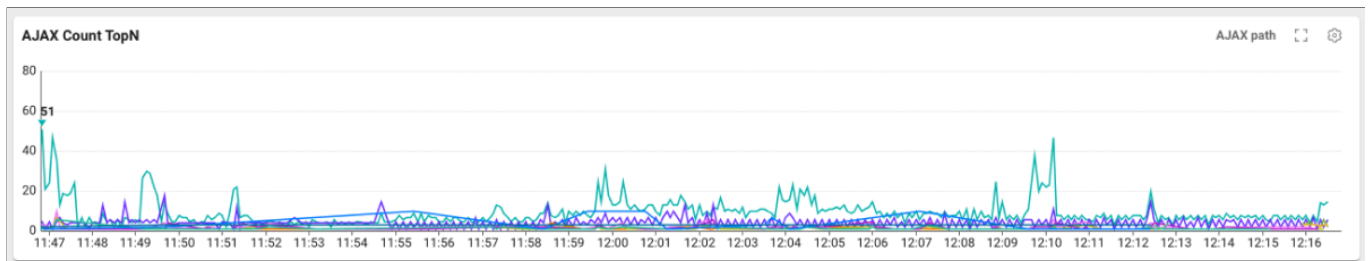
Home > Select Project > [Dashboard](#) > [AJAX](#)

AJAX is short for Asynchronous JavaScript And XML. It refers to using the `XMLHttpRequest` object to communicate with the server.

AJAX is an important data loading method to construct a dynamic screen among resource data. If AJAX is not called normally, the normal screen cannot be provided to end users. The collected data can help us identify what hinders the user experience.

- On the upper right of each widget, select . Then you can see the widget data in full screen.
- If you select  on the upper right of the widget, the chart configuration screen appears. You can set the options to display on the chart.
- Depending on the screen size of the web browser, the icon shape on the upper right of a widget may differ. If you select , the additional icons appear.
- For more information about the widget configuration, see [the following](#).

AJAX Count



You can check the number of AJAX loads in real time.

- On the upper right of the widget, select  and then set the time zone to see the AJAX load count.
- To change the data query criteria to [AJAX Host](#) or [All AJAX](#), select [AJAX Path](#) on the upper right of the widget.
- When retrieving the AJAX load count based on [AJAX Path](#) and [AJAX Host](#), the top 10 AJAX cases are provided.

- If you hover your mouse over the widget chart, AJAX count is provided in the form of a tooltip.

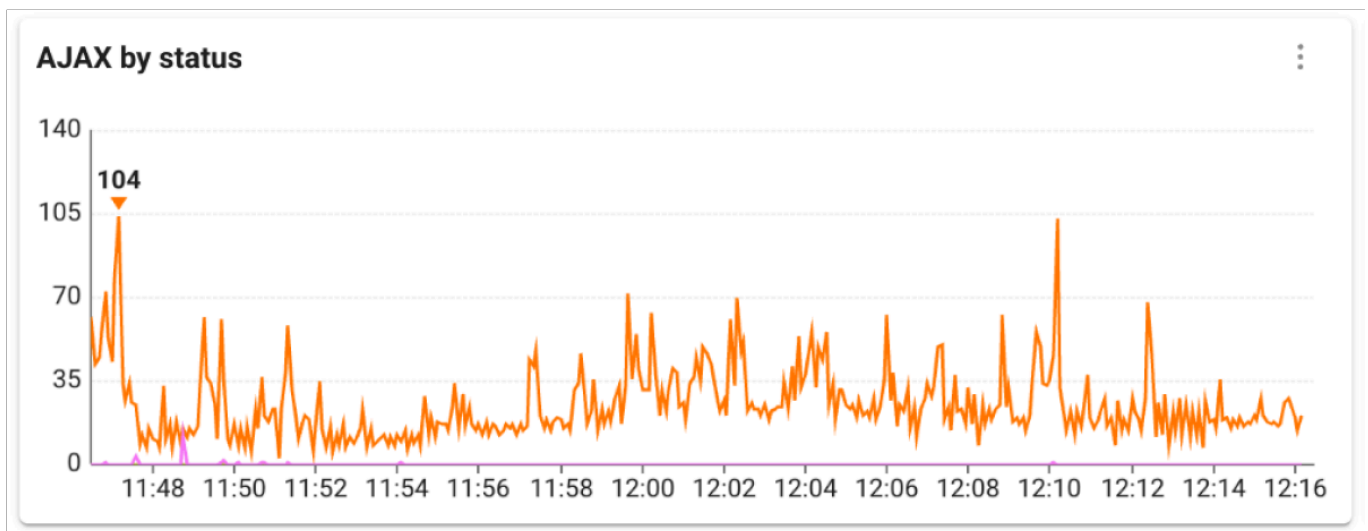
Average AJAX Response Time



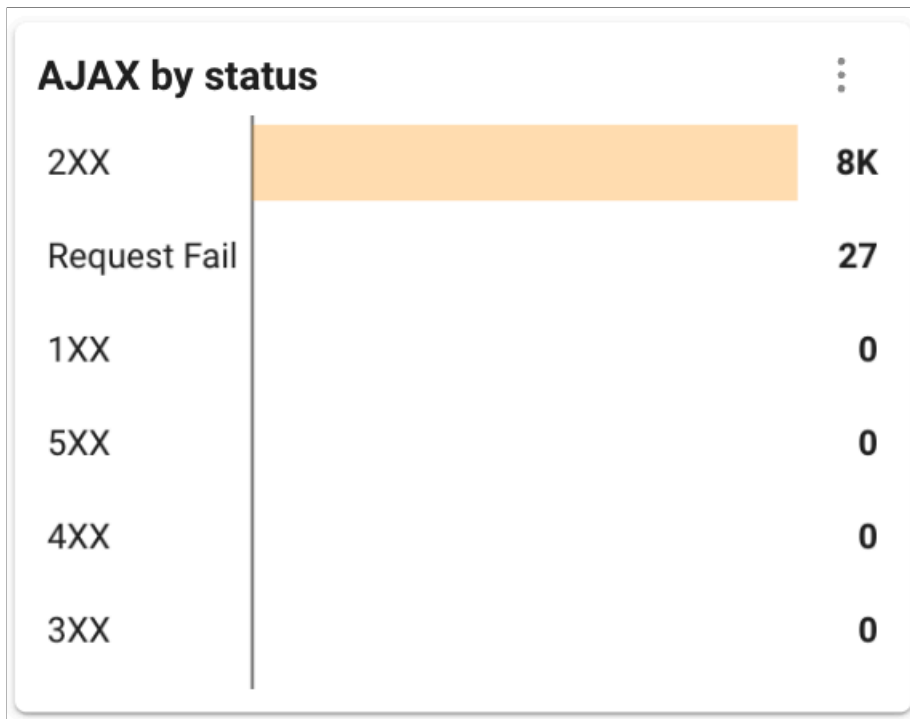
You can check the AJAX response time in real time. You can identify AJAX that takes a long average AJAX response time by specific paths and hosts.

- On the upper right of the widget, select and then set the time zone to see the AJAX response time.
- To change the data query criteria to [AJAX Host](#) or [All AJAX](#), select [AJAX Path](#) on the upper right of the widget.
- When retrieving the AJAX load count based on [AJAX Path](#) and [AJAX Host](#), the top 10 AJAX response times are provided.

AJAX By Status



You can check HTTP status codes related to AJAX by time of the day.



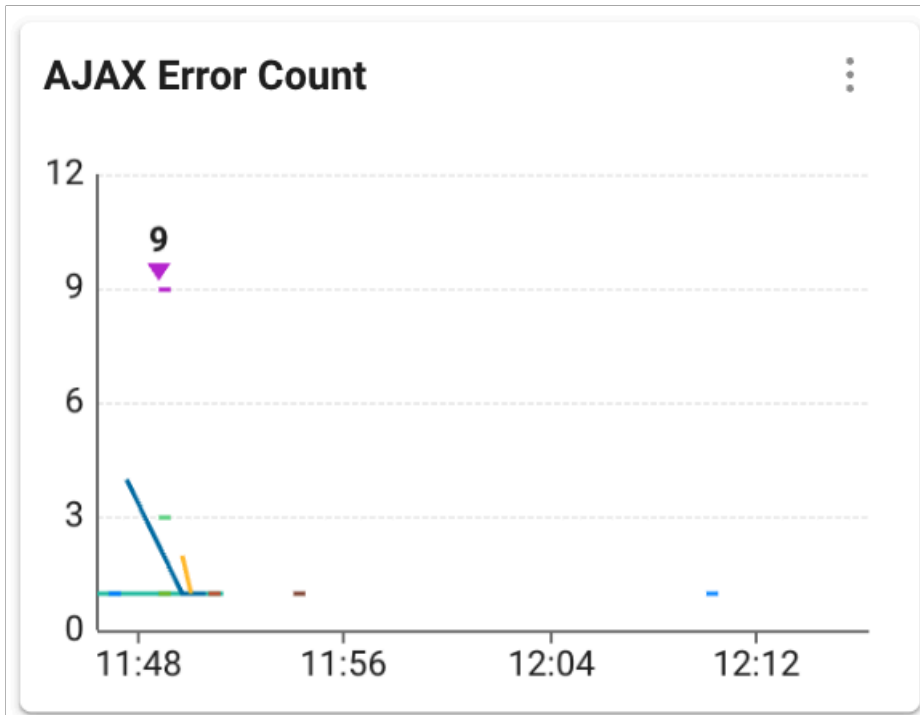
You can check the cumulative number of time-based HTTP status codes related to AJAX.

The statuses related to AJAX are as follows. Check that AJAX is running normally in your web application with the following criteria:

- **1xx** (response): This indicates that the server received any requests and the client connected to the server must proceed with the work.
- **2xx** (Success): The requests have been successfully received, recognized, and accepted.
- **3xx** (Redirection): Additional action is required to complete the request.
- **4xx** (Client Error): The syntax of the request is incorrect or the request cannot be processed.
- **5xx** (Server Error): The server failed to meet a valid request.
- **Request Fail** (request failure): It indicates that no response was received from the server and the request failed. For example, the requests are canceled from the front due to a timeout, or the requests are made incorrectly.

! For more information about the HTTP status codes, see [the following](#).

AJAX Error Count



You can check the number of AJAX-related errors in real time.

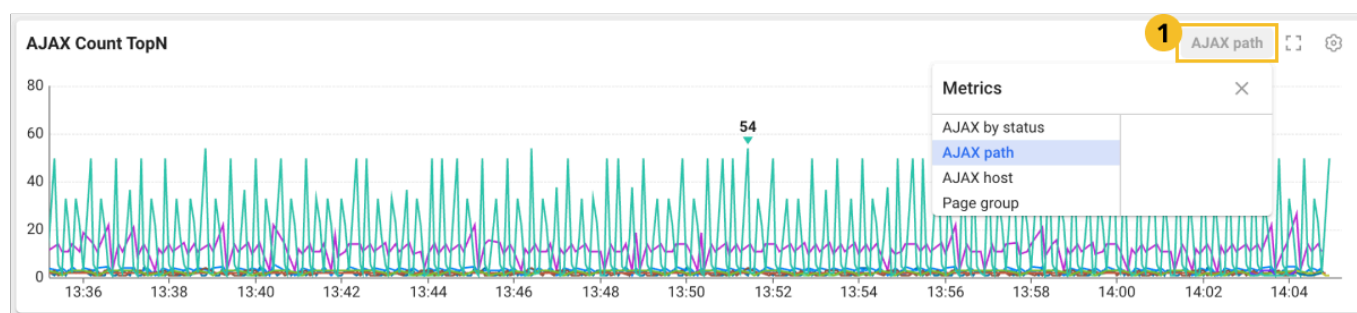
- On the upper right of the widget, select  and then set the time zone to see the AJAX error count.
- To change the data query criteria to [AJAX Host](#) or [Page Group](#), select [AJAX Path](#) on the upper right of the widget.
- If you hover your mouse over the widget chart when searching based on [Page Group](#), tooltips allow you to identify AJAX errors on a specific group of pages.

Setting A Widget

Widgets placed in the dashboard menu can be set with the desired options.

- ! Depending on the screen size of the web browser, the icon shape on the upper right of a widget may differ. If you select , the additional icons appear.

Changing data based on the graph chart

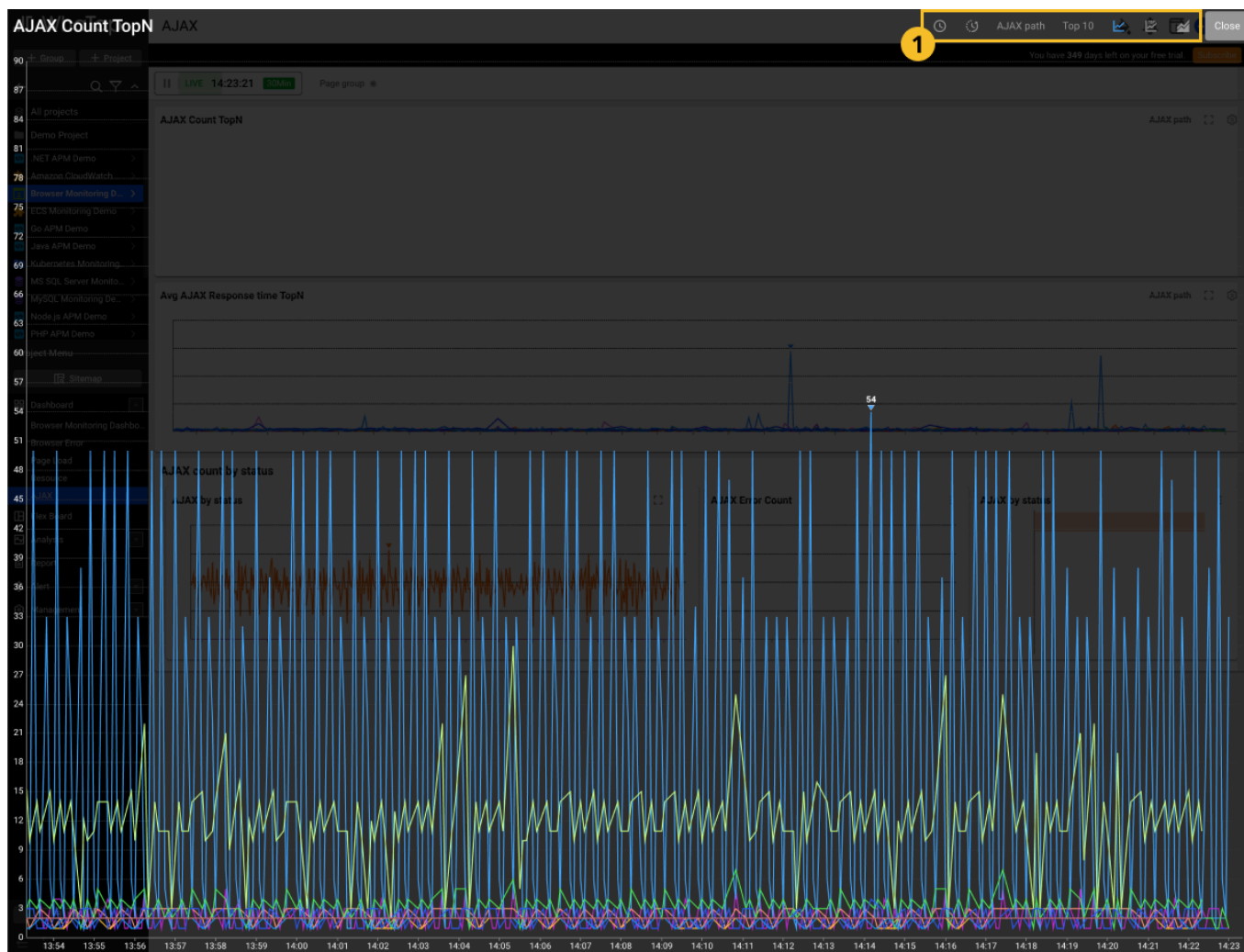


On the upper right of the widget, select the **1** Metrics button. The **metrics** pop-up window appears. You can change the data of the graph chart by selecting the desired item.

- ! • The items that can be selected in the **Metrics** pop-up window may differ depending on the widget.
- Depending on the widget's data, the Metrics button may not be available.

Viewing the full screen

Select  on the upper right of the widget to view on the full screen. You can check the widget's graph chart in full screen on the web browser.

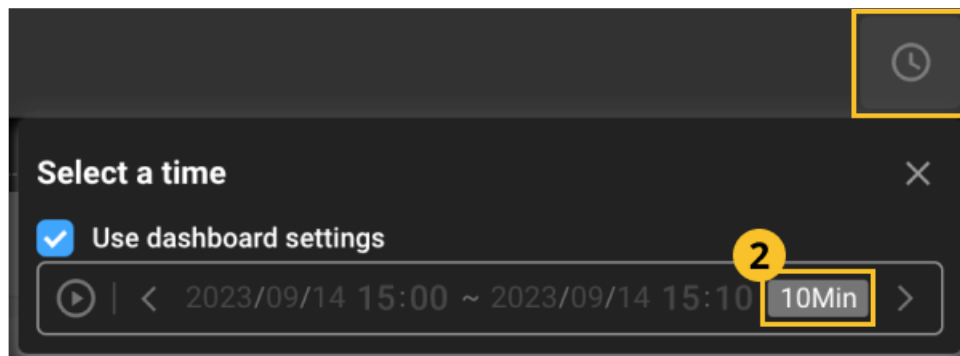


By selecting the **1** option buttons on the upper right of the screen, you can view the data with various ways.

ⓘ Depending on the widget's data type, available options may differ.

Changing the data lookup time

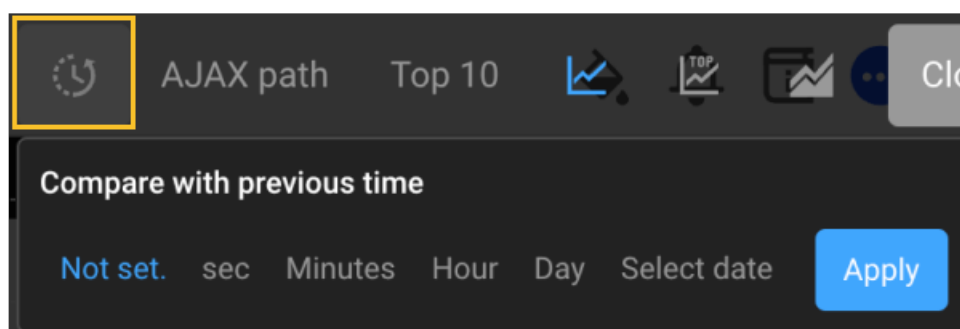
You can see the current data flow in real time and also check the data of the past. Select . The [Time Selector](#) pop-up window appears.



- To change the data lookup time on the graph chart, deselect [Use dashboard settings](#).
- To perform monitoring for graph charts in real time, select .
- The query time can be changed by selecting .
- By selecting  or , you can shift the time. The shifting interval is the  time interval set on the right.
- By clicking the  time interval button, you can select a time interval.
- Click the Date & Time area to manually select the desired date and time.

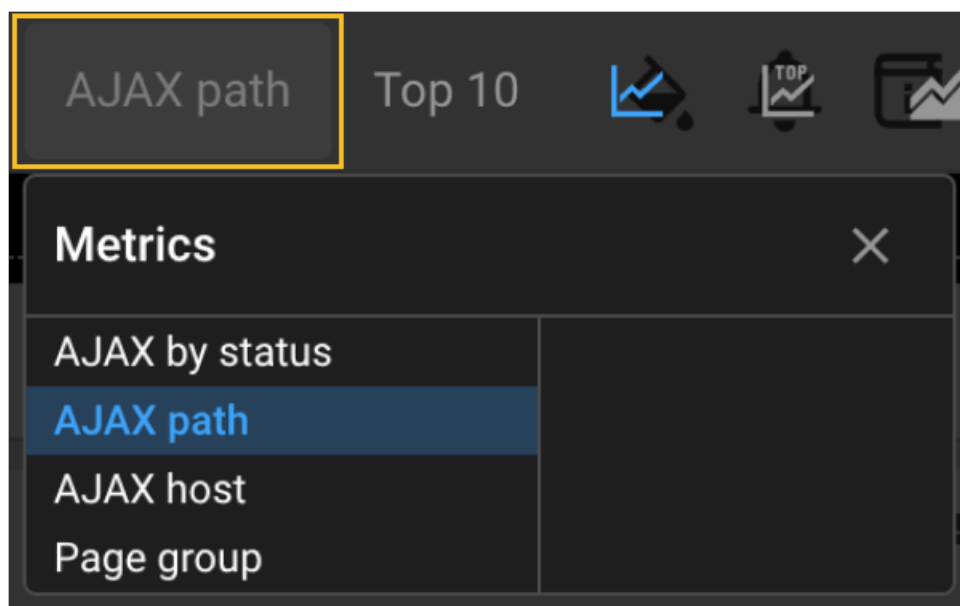
Comparing with the previous time

You can compare the past data with the current data. Select . The [Compare with previous time](#) pop-up window appears.



You can compare the time in seconds, minutes, hours, and dates, or select a date for comparison. Select the desired options and then click [Apply](#) to apply on the graph chart. To cancel comparison with the previous data, click [Not set.](#) and then select [Apply](#).

Changing the reference data

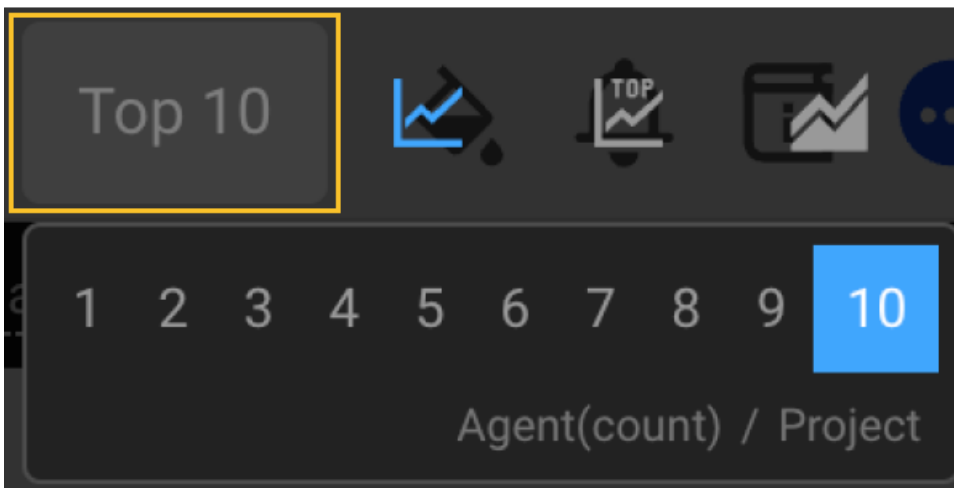


1. From the options on the upper right, select a text that corresponds to the widget's metric. (It is usually located at the third position.)
2. The metrics pop-up window appears.
3. Select a desired item.

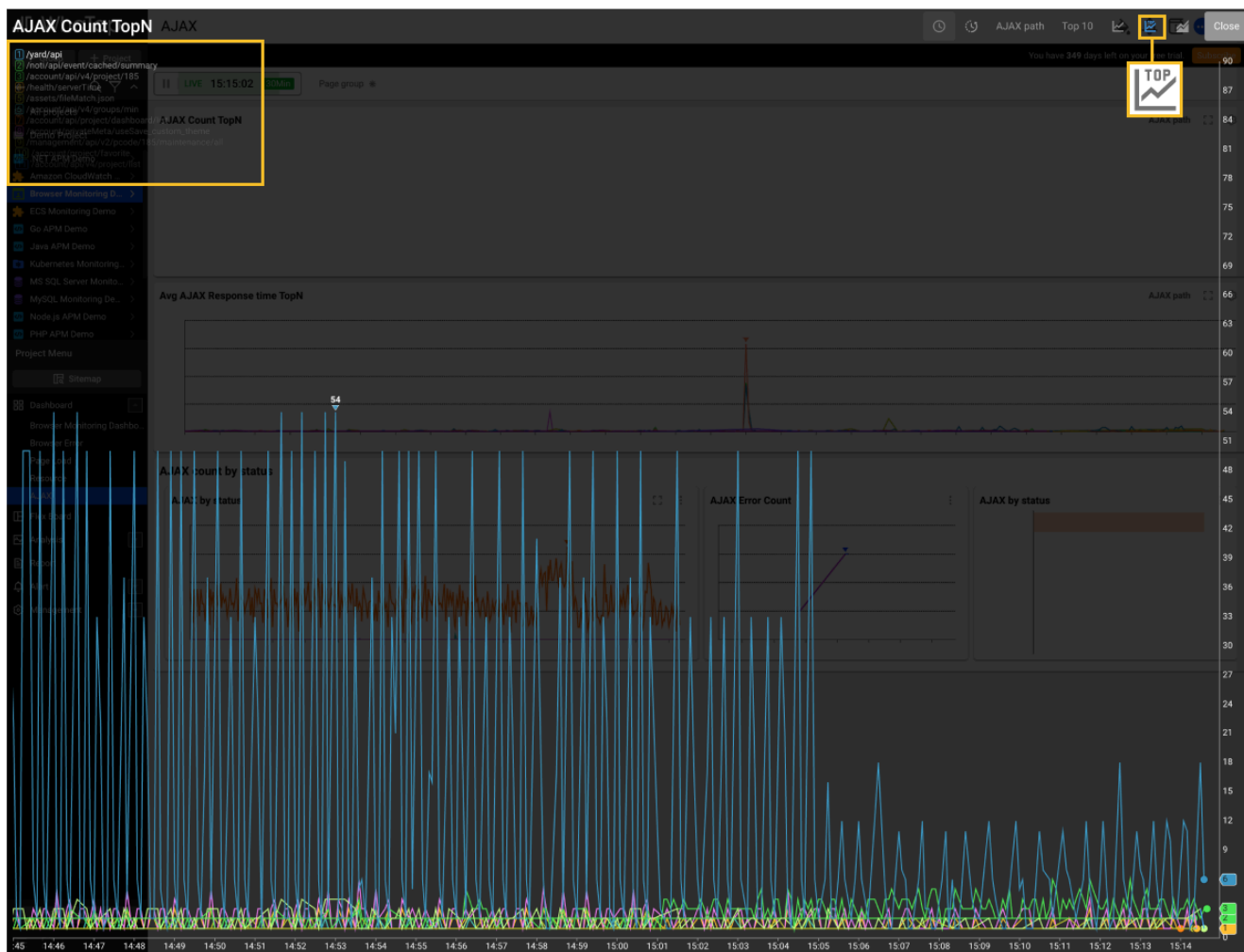
Change the data in the graph chart with the selected item.

Checking the top N data

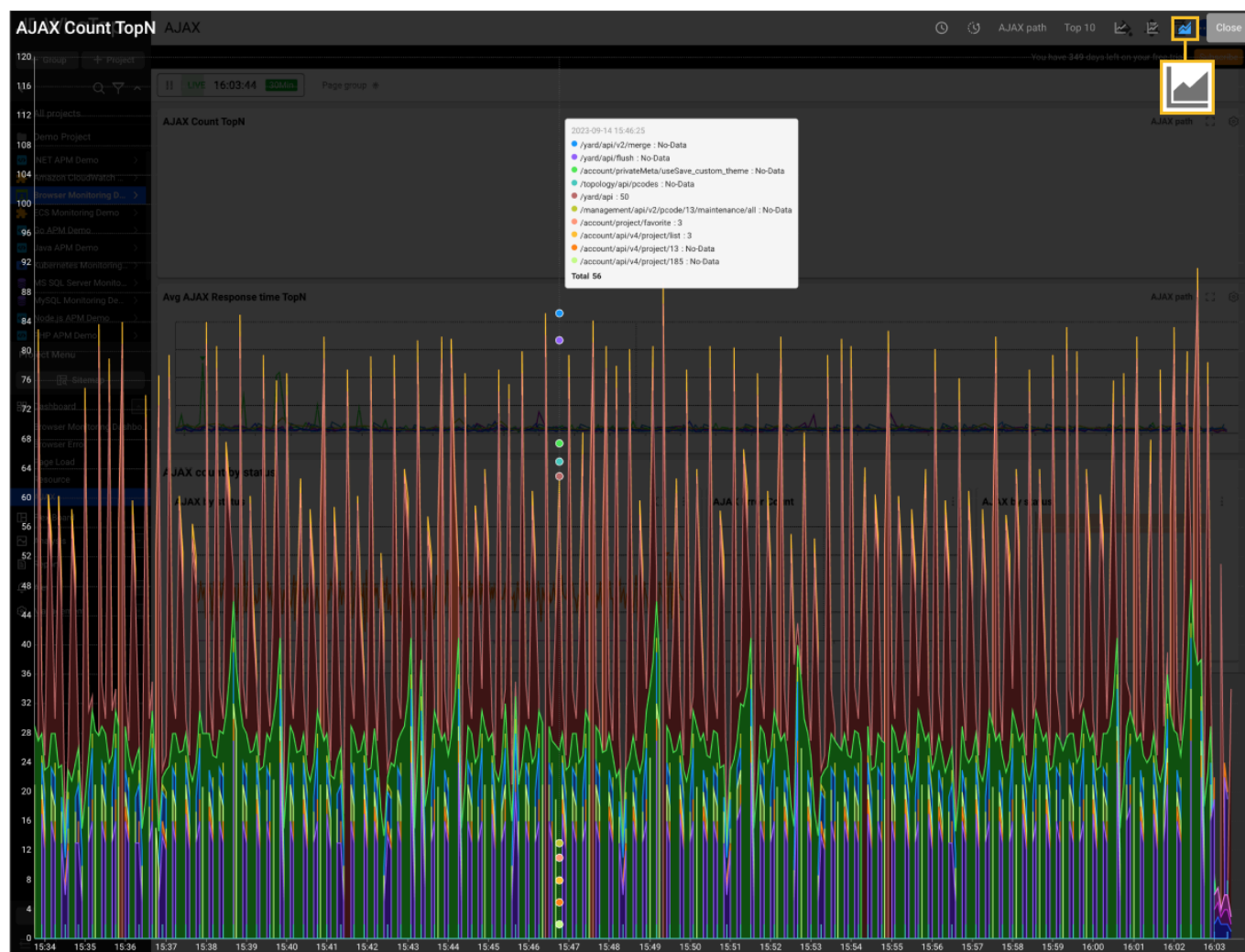
- If you click [Top 10](#), you can select up to top 10 items. You can apply the data to the graph chart in ascending order by the selected number.



- By selecting , you can display the top N data in the overlay format on the graph chart. To make the top N data disappear from the graph chart, select .



Checking all data



To compare all the data, select . Hover your mouse over the chart to see the figures for each data through a tooltip.

Viewing the data with MXQL

To search data by using MXQL, select . MXQL editing window appears.

For more information about MXQL, see [the following](#).

Changing the table columns

In case of the data table widgets, you can change columns in full screen. Select  on the upper right of the table. If the [Column Setting](#) window appears, you can hide or add any columns in the table. If you select [Save](#) after configuration, the settings are applied to the data table chart.



Configuring the chart

To set the maximum value of Y axis and add the threshold horizontal line, on the upper right of the widget, select ⚙️.

The configuration panel for the chart widget includes the following sections:

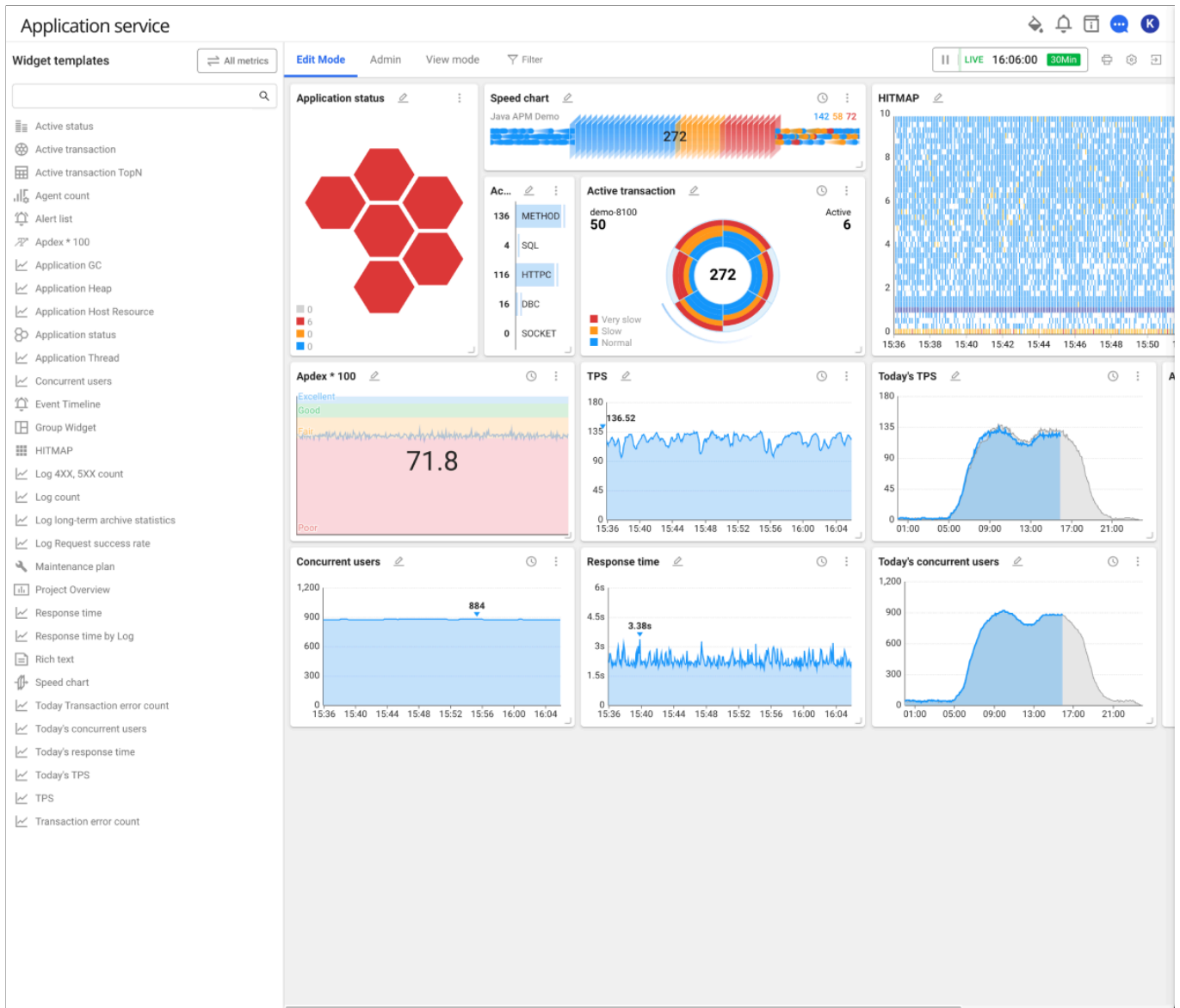
- y axis Chart max value:** Two buttons, **Auto** (selected) and **Fix**.
- Set horizontal line:**
 - Label:** A text input field containing "Text" and a delete icon (trash can).
 - Color:** A red circular color picker.
 - Threshold:** A button (selected) and a **Range** button.
 - Value:** A numeric input field containing "10".
- + Add:** A button to add a new horizontal line.
- Apply:** A blue button to save the configuration.
- Show chart max value:** A green toggle switch that is turned on.
- Show chart legend:** A grey toggle switch that is turned off.

- **Y-Axis Chart Max Value:** You can set the maximum value for the Y-axis by selecting **Auto** or **Fix**. If **Fix** is selected, you can enter a numeric value. If a number is input, select **Apply**.
- **Set Horizontal Line:** Add horizontal lines to the graph chart to set thresholds and monitor them efficiently.
 - Select **+ Add**, enter values for **Label** and **Threshold**, and then select a color. In **Threshold**, by selecting **Range**, you can set the horizon area. Select **Apply** to finish the configuration.
 - To delete the horizontal line, select **🗑️**.
- **Show Chart Max Value:** Displays the maximum value on the graph chart. It displays the highest number on the chart.
- **Show chart legend:** The names of each graph data under the chart widget are displayed. Comparing the graph colors and legends help you understand the chart.

Flex Board

Flex Board is a custom integrated dashboard. You can create a real-time dashboard with the project data such as applications, servers, databases, and containers.

You can quickly complete the initial setup with preconfigured templates and build dashboards in your preferred format. By adding data widgets or modifying properties, you can display information in the form you need, and with filtering functions, you can narrow down monitoring targets. When you set a time range, you can check data from a specific point in time, and by using auxiliary charts, you can analyze data in various ways. Dashboards can be added to favorites for easy access, and personalized dashboards can be copied to other accounts for reuse.



The usage scope of the Flex board differs depending on the following entry:

Home > Integrated Flex Board

- When creating a widget, all projects that can be viewed are provided as options.
- The dashboard is saved in your account and can be shared with other users with the "To copy" feature.
- Individual account dashboards are not affected by the role, but the shared dashboards are read-only and cannot be edited.

Home > Select Project > Dashboard > Flex Board

- When creating a widget, the related project data is automatically entered.
- The created Flex board is automatically shared with other project members.
- The users with the project **Edit** role or **Project flexboard update** role, and **site admin** role can modify the board.

✓ Users with edit permissions for the Flex Board can use the following features:

- Export and import dashboards as JSON files
- Check data requests and responses within the dashboard
- View and edit widget configuration options in JSON format

ⓘ In the **Flex Board** menu within the project, only the users with the dashboard editing role can access **Edit Mode**, **Admin Mode**, and **Filter**.

- Accessible member permissions
 - Project **Edit** role
 - **Project flexboard update** role
 - **Site Admin** role
- For more information about the **View mode** and **Filter**, see [the following](#).
- For more information about the member roles, see [the following](#).

Using Flex Board

- ✓ You can build dashboards using various widget templates, including metrics widgets.
For details on the features and usage of each template, see the [Widget template](#) document.

Apply a preset template

Use a preset template to speed up the initial setup, then modify it as needed.

1. Select the **Flex Board** menu.
2. From the **Template** list on the left, select the template you want.
3. When the **Flex board management** window appears, enter a name in **Rename**.
 - In **Layout**, select a widget and drag it to the position you want.
4. Click the **Save** button. A Flex board is created based on the selected template.
 - You can check the created board in the board list of **Flex Board**.

Create a Flex board

You can create a Flex board and place widgets to build your own dashboard.

1. In **Flex Board**, click the **Create Dashboard** button at the top right.
2. When the **Create Dashboard** window appears, enter the dashboard **Name** in the input box.
3. Select a widget placement method, then click the **Create Dashboard** button.
 - **Fixed layout flexboard**: free widget placement with pixel basis
 - **Responsive flexboard**: responsive grid layout based on browser size (configure the grid, then click the **Create Dashboard** button)
4. In the **Widget templates** selection screen, choose the widgets to place on the dashboard.

- To add metrics widgets, click the **All Metrics** button on the right of **Widget templates**.

- ! • **Widget templates:** choose from a predefined widget list that summarizes key indicators commonly used in monitoring.
- **All metrics:** create widgets based on all metrics data collected in your project. For details about metrics widgets, see [the following document](#).

5. Move the placed widgets or adjust their size.
 - When a crosshair cursor appears at the top of a widget, click and drag the widget to move it.
 - When an arrow cursor appears at the bottom right of a widget, click and drag to resize it.
6. After you finish placing widgets, select **View mode** to review the layout. To edit the layout again, select **Edit Mode** and adjust widget placement.
7. When all steps are complete, click the  (**Go to list**) icon at the top right.
 - You can check the created board in **Flex Board** under **Dashboard list**. Select the created board to view the new Flex board.

- ! We plan to add dashboard templates or widgets based on customer feedback. If you need a specific dashboard template or widget, contact us at support@whatap.io.

Edit a Flex board

! Flex board edit permissions

- **Home > Integrated Flex Board:** this is a personal account dashboard, so permissions do not apply; however, dashboards shared as read only cannot be edited.
- **Home > Select project > Dashboard > Flex Board:** users with project edit permission, project Flex board edit permission, or Site Admin permission can edit.

For details on member permissions, see [the following document](#).

1. In **Flex Board** under **Dashboard list**, click the edit icon . The **Flex board management** window appears on the right.
2. In the **Flex board management** window, configure Name, Project, Layout, widget.json, and option.json.
 - **Rename:** edit the dashboard name.
 - **Project:** select the project from which widgets will fetch data.

 You can configure the **Project** option only in **Home > Integrated Flex Board**. If you entered from **Integrated Flex Board**, you can select the data source project for each widget.

1. Click the **+** button at the top right of a widget, then search and select one or multiple projects.
2. After selecting projects, click the **Apply** button.

- **Layout**
 - Drag widgets to move them.
 - Click and drag the bottom right of a widget to resize it.
 - To delete a widget, click the  icon at the top right of the widget.
- **Widget.json:** import dashboard widget settings in JSON format. Click the  icon to copy the JSON.
- **Option.json:** import dashboard option settings in JSON format. Click the  icon to copy the JSON.

Delete a Flex board

1. In **Flex Board > Dashboard list**, click the  icon for the item to delete.
2. In the confirmation message, click the **Delete** button.

 Dashboard deletion behavior differs for owners and shared users.

- If the dashboard owner deletes a dashboard, it is deleted for all user accounts with whom it was shared.



- If a user who received a shared dashboard deletes it, it is deleted only from that user's **Dashboard list**. The original dashboard remains.

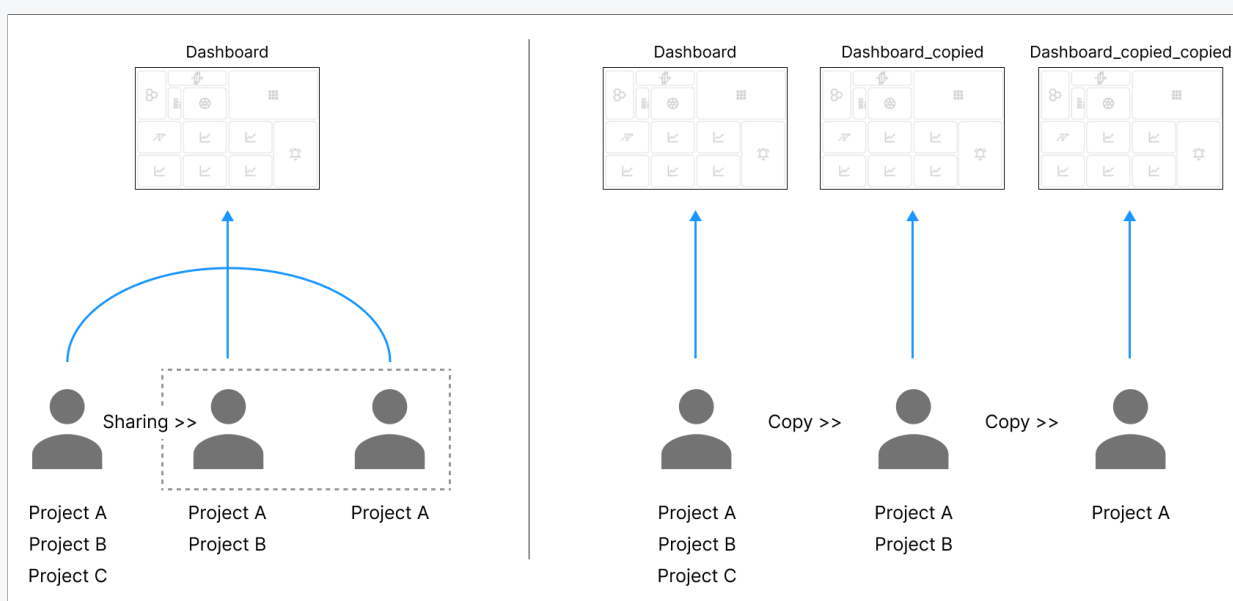
Share a Flex board



- The Flex board sharing feature is available only in **Home > Integrated Flex Board**. It is not provided in a project's **Flex board** menu.



Difference between share and copy



- Share: all members you share with view the same dashboard.
- Copy: if you copy a dashboard to another member or to yourself, you view a separate copied dashboard.
- Dashboards shared or copied as **Read only** cannot be edited. They can be edited only if shared as **Edit mode**.
- Members who receive a dashboard as **Read only** can also **re-share as read only** or **copy** that dashboard.



- You can share a dashboard only with members who belong to at least **one** of the projects included in the dashboard. Check whether the member to share or copy belongs to the relevant project.

☆ test

K8S-JINS-OKD-TEST, K8S-JINS-TEST-MICROK8S, DB-M...

Updated: 2023/12/22 11:10:24 AM



You can share a Flex board created in **Integrated Flex Board** with other members. When the owner edits a shared dashboard, the changes apply to all members it was shared with.

- In **Integrated Flex Board** under **Dashboard list**, click the icon for the dashboard to share.
- When the **Sharing a dashboard** window appears, select one or multiple members to share with, or enter accounts directly.
 - If you select **Read only**, members cannot edit the shared dashboard.
 - Share targets must belong to at least one of the projects included in the dashboard.
- Click the **Share** button.
 - Shared items display the **Shared** tag. Hover over the **Shared** tag to view information about the members you shared with.
 - Items shared as read only display the **Read only** tag.

When you receive a shared dashboard	When you share a dashboard (dashboard owner)



Dashboard sharing conditions

- The dashboard sharing feature is available only in **Home > Integrated Flex Board**.



- Anyone can share their own dashboard, and you can re-share or copy a dashboard you received.
- Dashboards received as **Read only** cannot be edited; dashboards received as **Edit mode** can be edited.
- If the **owner** deletes a dashboard, it is deleted for all accounts that received it.
- If a **recipient** deletes a dashboard, it is removed only from that user's **Dashboard list** and the original remains.
- For projects in a received dashboard where you do not have **View** permission, you cannot view data.

Copy a Flex board

Users can copy a Flex board and send it to other members.

1. In **Integrated Flex Board** under **Dashboard list**, click the  icon.
2. Choose a copy option.
 - **Copy to me:** copy to your own account.
 - a. An item with the `_copied` suffix is added to **Dashboard list**.
 - **Copy to others:** copy to another member's account.
 - a. Click **+ Account**.
 - b. Select the email of the member to copy to and click the **Copy dashboard** button.
 - c. An item with the `_copied` suffix is added to that member's **Dashboard list**.

Share as a JSON file

You can export Flex board settings to a JSON file to share with other members, or import settings from other members.

Export

1. Go to **Integrated Flex Board** or a project's **Flex board** menu.
2. In **Dashboard list**, click the  button.
3. Choose an export option.
 - **Include Project Information:** include dashboard project information in the JSON file when sharing.
 - **Except Project Information:** share without including dashboard project information in the JSON file.
4. Download the JSON file and send it to the member you want to share with.

Import

1. Go to **Integrated Flex Board** or a project's **Flex board** menu.
2. Click the  **Import** button at the top right.
3. Select the downloaded JSON file.

 Only members with **Edit** permission for the project can use this feature.

Flex board screen modes

Features vary by Flex board screen mode.

Accessible member permissions

Only users with **Edit** permission in a project's **Flex Board** menu can access **Edit Mode**, **Admin**, and **Filter**. For details on member permissions, see [the following document](#).

- **Project Edit** permission
- **Project flexboard update** permission
- **Site Admin** permission

Modes

Available features differ by **Flex Board** screen mode.

- **Edit Mode:** place widgets from the widget template list on the Flex board layout; edit, delete, or reposition widgets.
- **Admin:** manage widget properties in JSON format. Click the  button at the top right of a widget on the layout.
- **View mode:** after finishing layout, sizing, and detailed options, monitor each widget's data in real time like a dashboard.

Common features

- Data filtering
- Real time data time range settings
- Print mode
- Flex board management
- Data merge options

Data filtering

Use data filtering to narrow monitoring targets and build an efficient Flex board.

1. At the top of the screen, click the  **Filter** tab.
2. In the **Filter** window on the right, select filter targets. To select all filter targets, choose **Select all**.
 - **Agent** option: select agent types or an application list, with **Search** available
 - **Metrics** option: select a project or agent type; add **+ And** conditions, or auto select by string with **Contain** or **Exclude**
3. After selecting filter targets, click the **Apply** button.
4. Save frequently used conditions with the **Save filter** button.
 - a. Enter a filter name and select **Save**. **Saved filters** appears at the bottom of the **Filter** window.

Filtering conditions

Filtering conditions are divided into agent type and metrics type.

Agents provide only basic monitoring targets as selectable options, while metrics provide selectable tags for all metrics data collected in the project. The metrics type is useful when you collect data in a non agent form and want to apply filtering. For details on metrics, see [the following document](#).

Type	Agent	Metrics
Selection options	WhaTap project monitoring targets (applications, servers, databases, containers, Kubernetes PODs and NODEs)	All metrics tags including agents
Scope	All widgets	All widgets that use metrics data
Filtering method	Checkboxes	Equal, Contain, Exclude

Set the lookup time

Use the time selector at the top right of **Flex Board** to set data lookup time for widgets.

- : stop real time mode and set a specific lookup time
- : switch back to real time mode

Print

You can save the Flex board screen as a PDF and print it.

1. On the Flex board screen, click the  icon at the top right to switch to print mode.
2. Configure options.
 - **Viewing filter options:** toggle filter options on the screen
 - **Convert to {mode} orientation / Convert to {mode} orientation:** change orientation to horizontal or vertical
3. After finishing, select **PDFDownload**.

4. To exit print mode, click the  **Exit print mode** at the top right.

Flex board management

You can change the Flex board layout; move, resize, or rename widgets; and use other management features. You can also open **Flex board management** by clicking the  icon in **Dashboard > Flex Board** under **Dashboard list**.

1. On the Flex board screen, click the  icon at the top right to open the **Flex board management** window.
 - **Rename:** change the Flex board name
 - **Layout:** move, resize, and delete widgets
 - **Don't Allow Other Users to Modify.:** restrict modifications
 - Members with edit permission can edit, delete, and modify; if you enable the option, it changes to **Don't Allow Other Users to Modify.: On**
 - **Widget.json:** edit and manage widgets on the Flex board in JSON
 - **Option.json:** edit and manage Flex board settings in JSON (project info, Flex board filters, time selection, and other settings)

 The **Widget.json** and **Option.json** features are available only to dashboard administrators.

2. After completing all settings, click the **Save** button.

Add a dashboard to favorites

You can add a Flex board you created to favorites for quick access. In **Dashboard > Flex Board** under **Dashboard list**, click the  button for the dashboard to add. It changes to .

Managing The Widgets

You can customize your dashboard by handling the widgets placed on the Flex board.

- ✓ You can build dashboards using various widget templates, including metrics widgets.
For details on the features and usage of each template, see the [Widget template](#) document.

Adding a widget

You can add any widget to the created Flex board.

1. In **Dashboard list** of **Flex Board**, select a dashboard to add the widget.
 2. On the upper left of the screen, select **Edit Mode**.
 3. Select a desired widget from the **Widget templates** list on the left. To add a metrics widget, select **All metrics** and then click a desired widget.
- If you have entered **Integrated Flex Board** on the Home screen and added a widget, click **Project selection** and then select target projects to collect.
 - If you have more widgets to add, repeat the same procedure.
4. After all steps are finished, select **View mode** to check the layout and then select  (**Go to list**).

Changing the widget attributes

In **Dashboard list** of **Flex Board**, select a board to change the widget properties. It goes to **Edit Mode**.

Changing the widget name

1. On the right of the widget name to change, select .
2. Set the widget values for **Title**, **Font size**, and **Color**, and then select **Save**.

- If the **Apply to the entire widget** toggle button is enabled, the font size and color are applied to other widgets.
- To lock the widget title, select [Configuration icon](#) and then click **Save**.

Time Setting

1. On the upper right of the widget to set the time, select ⌚.
2. Set the time in the **Select a time** window.
 - To apply the set time to the dashboard, check **Use dashboard settings**. The time setting on the dashboard is located on the upper right.
 - To apply a desired time setting, uncheck **Use dashboard settings**. Use < or > to set the desired time.
3. After setting the time, select ✕.

ⓘ Depending on the data type of the widget, the time setting may not be available.

Setting the output data options

By setting the output data options of the chart widget, you can select the desired data.

1. At the top right of the widget, click the ⋮ icon to configure the output data options.
 - You can filter data by selecting types for project, agent, application, etc.
 - You can change the type of the graph displayed on the chart by selecting individual or merged data.
 - Select ⋮ > ⚙️. You can change the option details through the **Chart settings** window. The options that can be set are as follows.
 - **Chart type:** This option allows you to display the current aggregated figures on the chart.
 - **y axis Chart max value:** The chart maximum value on the Y axis can be selected with one of **Auto** and **Fix**.
 - **Set horizontal line:** Select **Add** to display the threshold or range as a horizontal line. You can

apply multiple items added. After adding the horizontal line, select **Apply** to apply it to the chart.

- **Show chart max value:** The maximum value of the graph is displayed on the chart.
 - **Sub-chart:** You can add a sub-chart to the main charts of the widget. Select a type of sub-chart and then select its layout, size, and whether or not to merge data. To apply to other widgets, select **Apply to the entire widget**. Sub-charts are useful to view data in various ways.
2. After setting all options, on the left of the **Chart settings** window, click **×** or click the outside of the **Chart settings** window.

ⓘ Depending on the widget's data type, available options may differ.

Changing the metrics widget data

You can change the data query conditions of the metrics widget.

1. In **Flex Board > Dashboard list**, select the dashboard where you want to change the widget data.
2. In **Edit Mode**, click the **⋮** icon at the top right of the widget you want to modify.
3. Click the **<>** icon to open the data settings edit window.
4. Change the settings referencing the following, and then select **< Apply to the widget**.
 - Metrics options that are created by default when creating a widget appear on the tab.
 - Metrics options are **Category**, **Field**, and **Tag**.
 - The **Category** currently restricts the editing function.
 - In **Field** and **Tag**, multiple items can be selected. The chart data is grouped using the selected tag options.
 - Select **🔍** to query the original data.
 - For more information about **Object Merge** and **Time merge**, see the [Data merge options](#).

Data merge options

The metrics widget placed on the **All metrics** list of the Flex board provides the data merge options. Data merge provides the following methods: **Object Merge** and **Time merge**.

- **Object Merge**

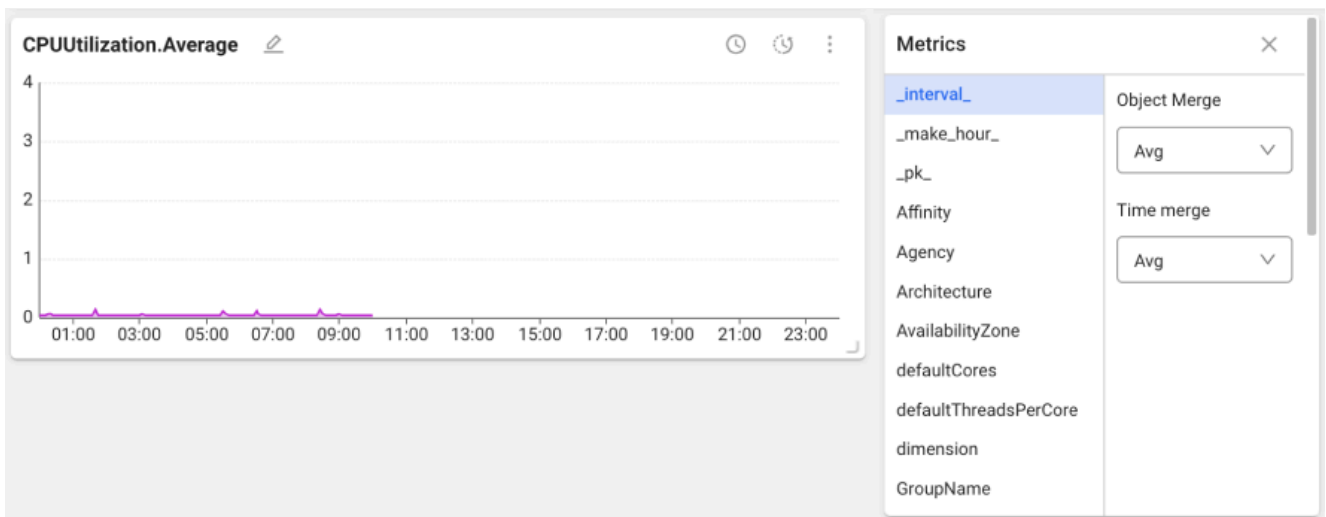
It is used to group the chart data based on the tag (e.g. agent name) selected on the left. At this time, if the tags match among the data with different fields (e.g. CPU), the corresponding data is merged.

- **Time merge**

It is mainly used if the lookup time is too long. Data with the same fields in the original data are merged at regular time intervals (5 minutes or 1 hour) depending on the time range. This method is called "Time Merge."

- **AWS ColudWatch Metrics**

In case of AWS monitoring metrics, the statistical method is added after the field name in the form of suffix. Data merging defaults reference this suffix.



1. In **Edit Mode**, select **All metrics** and then change the **Widget templates** list to the **All metrics** list.
2. Place the metrics widget on the layout.
3. On the upper right of the metrics widget, select **:**.
4. If a tag (e.g. agent name) option is selected, the data merge options appear. Depending on the widget size, the metrics option's location may differ.

Copying a widget

1. Click  on the upper right of the widget.
2. Click .
3. If the **Copy** window appears, click **+ Add widget**.
4. Change the widget name in the list.
5. Click **Apply**.
6. In **Edit Mode**, you can see the copied widget.

Delete Widget

1. Click  on the upper right of the widget.
2. Click .
3. If the confirmation message appears, click **Delete**. To cancel the deletion, select **Cancel**.

User Action Hitmap

! Feature reorganization notice

The former **Page Load Analysis** feature has been reorganized into **User Action Analysis**. Its scope now covers not only page load events but also clicks, touches, and other user interactions, so you can analyze them together.

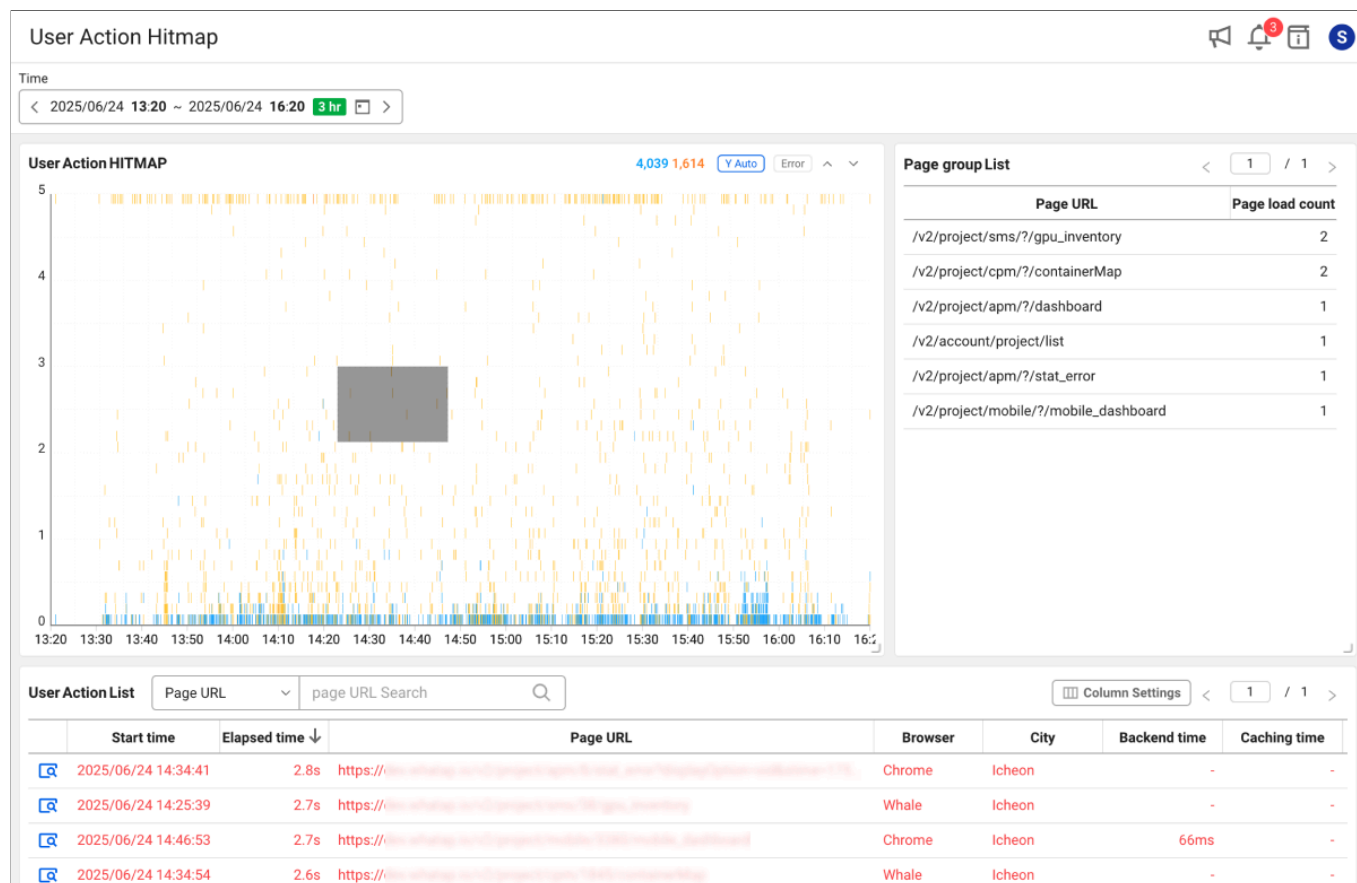
Home > Select Project > **Analysis** > **User Action Hitmap**

A [user action](#) event occurs when an end user visiting your web application finishes loading a page or interacts with it by clicking, touching, and so on. These events are provided in a hitmap chart, and the detailed information about each event can be viewed in a timeline. This lets you identify performance problems and the factors that degrade the user experience. The following information is included.

- **Page load events**
 - Time details for each page load event section
 - Details of the loaded resources
 - Browser error data
- **Click and touch events**
 - Click response time
 - Information about the resources loaded on the click
 - Information about the errors that occurred on the click

Checking the user action event list

You can analyze a single user action event. A hitmap and various metrics are provided to help you analyze user action performance.



1. Drag the chart area of the **User Action Hitmap** widget.
2. The event list for the dragged area appears as the **User Action List** below the **User Action Hitmap**, and you can check it by page group in the **Page group List** on the right of the screen.
 - You can search the page list you want in **Page URL** of the **User Action List**.

User Action Hitmap

The **User Action Hitmap** widget shows the response time of user actions over time as a distribution chart. Drag the chart area of the widget. The **User Action List** for the dragged area appears at the bottom of the screen.

- The **horizontal axis** is the end time of the user action event.
- The **vertical axis** is the elapsed time (response time) of the user action event.
- The density of normal user action events is shown in the order of **light blue → blue → navy**.

- The density of user action events with errors is shown in the order of **yellow** → **orange** → **red**. A user action event counts as an error when a browser error occurred while the event was being processed.

Hitmap AI Analysis

Click the **AI Analysis** button on the option bar to have WhaTap AI read the hitmap distribution and report what it finds. The button shows an icon only; hover over it to see its name in a tooltip. The analysis works differently depending on whether an area is selected.

- **No area selected:** WhaTap AI reads the whole hitmap for the queried period and suggests the time ranges and response time ranges to check.
- **Area dragged:** WhaTap AI analyzes the events inside the selected area and presents candidate causes of the slowdown. The selected area appears to the left of the button. To clear it, click the **X** icon on the item.

When you drag an area, the **User Action List** appears together as before. AI Analysis does not change this behavior.

If the button is disabled, hover over it to see the reason.

Message	Description
Loading data	The hitmap query is in progress. The button is enabled once the query finishes.
No data in the selected period.	The queried period contains no event. Widen the time range and query again.

ⓘ For how AI Analysis behaves in common and the data scope it uses, refer to [the following document](#).

User Action List

The user action events in the area you dragged on the **User Action Hitmap** appear in the **User Action List**. Click the  icon on the far left of each item in the list to open the **User Action Analysis** detail window.

The **User Action List** includes the following basic information.

- **Start time:** Start time of the user action event
- **Elapsed time:** Time taken until the user action event completed

- **Page URL:** URL path of the page where the user action occurred
- **Browser:** Browser type used by the end user
- **Device:** Device type used by the end user
- **OS:** OS of the device used by the end user
- **Client IP:** IP address of the device used by the end user

ⓘ WhaTap stores client-related information by default.

Adding and editing column information

You can add other information columns to the table, or hide the columns you added. Click the  **Column Settings** button at the top right of the **User Action List**.

Filtering the user action list

You can enter a search term and filter the user action list for matching items. Select the item you want to query, enter a search term, and press Enter or click .

- **Page URL**
- **Browser**
- **Device**
- **OS**
- **Client IP**
- **User ID**

Filtering the page group list

You can filter the page group list. Select the page group list you want.

Getting a recommended investigation order from the list

Click the **AI Analysis** button above the **User Action List** to have WhaTap AI recommend which of the queried events to check first. The sample considers slow items and items with errors together, so you can narrow down

candidate causes faster than by sorting the list and scanning from the top.

1. Drag an area on the hitmap to show the **User Action List**.
2. Click the **AI Analysis** button above the list.
3. The **AI Analysis** drawer opens on the right of the screen and shows the recommended items with the reasoning.

- Click a recommended item to open the **User Action Analysis** detail window for that event.
- The recommendation does not change the sort order of the list or the filters you applied.
- Query string values in URLs and user IDs are masked before being used for the analysis.

If the button is disabled, hover over it to see the reason.

Message	Description
Loading data	The list query is in progress. The button is enabled once the query finishes.
No log collected.	The dragged area contains no event. Select another area and query again.

! For how AI Analysis behaves in common and the data scope it uses, refer to [the following document](#).

User action event details

Click the  icon on the left of each item in the **User Action List** to open the **User Action Analysis** detail window. The window consists of the event list on the left and the detailed information on the right. Various metrics are provided to help you analyze user action performance and identify what affects it.

Event list

The user action events for the selected time range appear as a list on the right of the **User Action Analysis** detail window, sorted by elapsed time. Click an event in the list to show its detailed information in the right panel.

User action analysis 35 cases have been inquired. ⓘ Copy URL

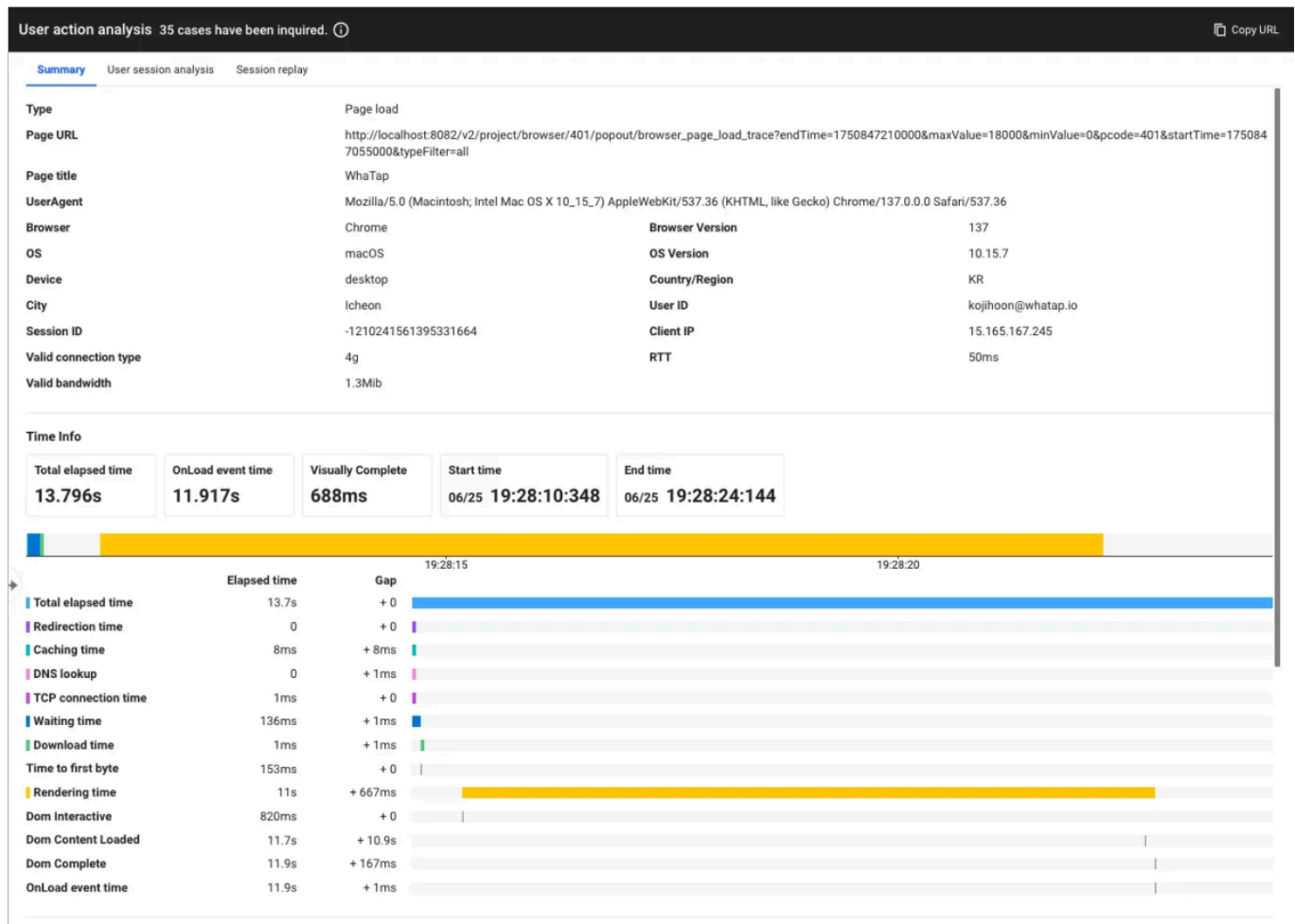
All Column Settings Page URL Search...

Index	Event name	Start time	Total elapsed time	Page URL	Browser	OS	Device	Client IP
1	Page load	2025/06/25 19:28:10	13.7s	http://localhost:8082/v2/project/browser/40...	Chrome	macOS	desktop	15.165.167.245
2	Page load	2025/06/25 19:26:19	13s	http://localhost:8082/v2/project/browser/40...	Chrome	macOS	desktop	15.165.167.245
3	Page load	2025/06/25 19:27:09	12.6s	http://localhost:8082/v2/project/browser/40...	Chrome	macOS	desktop	15.165.167.245
4	Page load	2025/06/25 19:27:14	9s	http://localhost:8082/v2/project/browser/40...	Chrome	macOS	desktop	15.165.167.245
5	Page load	2025/06/25 19:27:00	5.2s	http://localhost:8082/v2/account/project/list	Chrome	macOS	desktop	15.165.167.245
6	동계	2025/06/25 19:27:35	4.5s	http://dev.qa-3631.feature.whatap.io/v2/proj...	Chrome	macOS	desktop	15.165.167.245
7	Page load	2025/06/25 19:28:51	4.2s	http://localhost:8082/v2/account/project/list	Chrome	macOS	desktop	15.165.167.245
8	URL 패턴 동계	2025/06/25 19:27:36	3.6s	http://dev.qa-3631.feature.whatap.io/v2/proj...	Chrome	macOS	desktop	15.165.167.245
9	Created with Sketch.	2025/06/25 19:26:40	3.1s	http://localhost:8082/v2/project/browser/40...	Chrome	macOS	desktop	15.165.167.245
10	no data	2025/06/25 19:27:22	2.8s	http://dev.qa-3631.feature.whatap.io/v2/acco...	Chrome	macOS	desktop	15.165.167.245
11	CSV	2025/06/25 19:27:38	2.3s	http://dev.qa-3631.feature.whatap.io/v2/proj...	Chrome	macOS	desktop	15.165.167.245
12	grid	2025/06/25 19:28:36	2.3s	http://localhost:8082/v2/project/browser/40...	Chrome	macOS	desktop	15.165.167.245
13	Page load	2025/06/25 19:27:18	1s	http://dev.qa-3631.feature.whatap.io/v2/acco...	Chrome	macOS	desktop	15.165.167.245
14	URL Path /v2/account/...	2025/06/25 19:27:34	432ms	http://dev.qa-3631.feature.whatap.io/v2/proj...	Chrome	macOS	desktop	15.165.167.245
15	프로젝트	2025/06/25 19:27:41	316ms	https://dev.whatap.io/v2/admin/projects	Chrome	macOS	desktop	15.165.167.245
16	식재 완료	2025/06/25 19:27:47	130ms	https://dev.whatap.io/v2/admin/projects	Chrome	macOS	desktop	15.165.167.245
17	컬럼 설정	2025/06/25 19:28:15	118ms	https://dev.whatap.io/v2/project/sms/38/ser...	Chrome	macOS	desktop	15.165.167.245
18	개요 사용자 세션 분석 세...	2025/06/25 19:29:10	89ms	http://localhost:8082/v2/project/browser/40...	Chrome	macOS	desktop	15.165.167.245
19	필터 키	2025/06/25 19:26:26	63ms	https://dev.whatap.io/v2/project/sms/38/ser...	Chrome	macOS	desktop	15.165.167.245
20	search	2025/06/25 19:27:49	60ms	https://dev.whatap.io/v2/admin/projects	Chrome	macOS	desktop	15.165.167.245
21	test custom	2025/06/25 19:28:16	54ms	https://dev.whatap.io/v2/project/sms/38/ser...	Chrome	macOS	desktop	15.165.167.245
22	infra_Domain custom	2025/06/25 19:28:17	44ms	https://dev.whatap.io/v2/project/sms/38/ser...	Chrome	macOS	desktop	15.165.167.245
23	적용된 필터가 없습니다.	2025/06/25 19:26:25	34ms	https://dev.whatap.io/v2/project/sms/38/ser...	Chrome	macOS	desktop	15.165.167.245
24	91 Server Beta #92 192...	2025/06/25 19:29:34	29ms	http://localhost:8082/v2/project/sms/38/gpu...	Chrome	macOS	desktop	15.165.167.245
25	ALL SMS APM JAVA P...	2025/06/25 19:27:48	28ms	https://dev.whatap.io/v2/admin/projects	Chrome	macOS	desktop	15.165.167.245
26	91 Server Beta #92 192...	2025/06/25 19:29:30	27ms	http://localhost:8082/v2/project/sms/38/gpu...	Chrome	macOS	desktop	15.165.167.245
27	91 Server Beta #92 192...	2025/06/25 19:29:34	20ms	http://localhost:8082/v2/project/sms/38/gpu...	Chrome	macOS	desktop	15.165.167.245
28	91 Server Beta #92 192...	2025/06/25 19:29:34	18ms	http://localhost:8082/v2/project/sms/38/gpu...	Chrome	macOS	desktop	15.165.167.245
29	적용된 필터가 없습니다.	2025/06/25 19:26:24	0ms	https://dev.whatap.io/v2/project/sms/38/ser...	Chrome	macOS	desktop	15.165.167.245
30	평균 페이지 로드 시간 To...	2025/06/25 19:27:15	0ms	http://localhost:8082/v2/project/browser/40...	Chrome	macOS	desktop	15.165.167.245
31	프로젝트 [Enter]	2025/06/25 19:27:43	0ms	https://dev.whatap.io/v2/admin/projects	Chrome	macOS	desktop	15.165.167.245
32	프로젝트 [Enter]	2025/06/25 19:27:43	0ms	https://dev.whatap.io/v2/admin/projects	Chrome	macOS	desktop	15.165.167.245

- **Response time:** Response time of the user action (ms)
- **Page URL:** URL of the page where the user action occurred
- **Browser:** Browser type used by the user
- **Device:** Device type used by the user
- **OS:** OS used by the user

Detailed information

On the left of the **User Action Analysis** detail window, you can check the detailed information about the selected event on the Overview, User Session Analysis, and Session Replay tabs. The information consists of the following.



Basic information

- **Page URL:** Full URL of the page where the event occurred
- **Page title:** Title of the web page
- **User agent:** Details of the browser and OS
- **Browser/Version:** Browser type and version

- **OS/Version:** OS type and version
- **Device:** Device type used (desktop, mobile, and so on)
- **City:** Access location of the user
- **Session ID:** Identifier of the user session
- **Client IP:** IP address of the user
- **Effective connection type:** Network connection type (3g, 4g, wifi, and so on)
- **Effective bandwidth:** Network bandwidth information
- **RTT:** Round Trip Time

Time information

You can check the total elapsed time, start time, and end time of the event.

Time taken per section (timeline chart)

For a **page load** event, you can check the time taken per section visually.

- **Redirect time:** Time taken to redirect the page
- **Request waiting time:** Time waiting for the server response
- **Download time:** Time taken to download resources
- **Time to first byte:** Time until the first byte was received
- **Dom Interactive:** Point at which the DOM became interactive
- **Dom Content Loaded:** Time at which the DOM content was loaded
- **Dom Complete:** Time at which the DOM completed
- **OnLoad event time:** Time at which the page load completed
- **Caching time:** Time taken to look up cached resources
- **DNS lookup:** Time taken to look up the domain
- **TCP connection time:** Time taken to establish the TCP connection

Overview AI Analysis

Click the **AI Analysis** button at the top right of the **Overview** area to have WhaTap AI read the time taken per

section and the resource list of the selected event, and report which section drove up the overall response time.

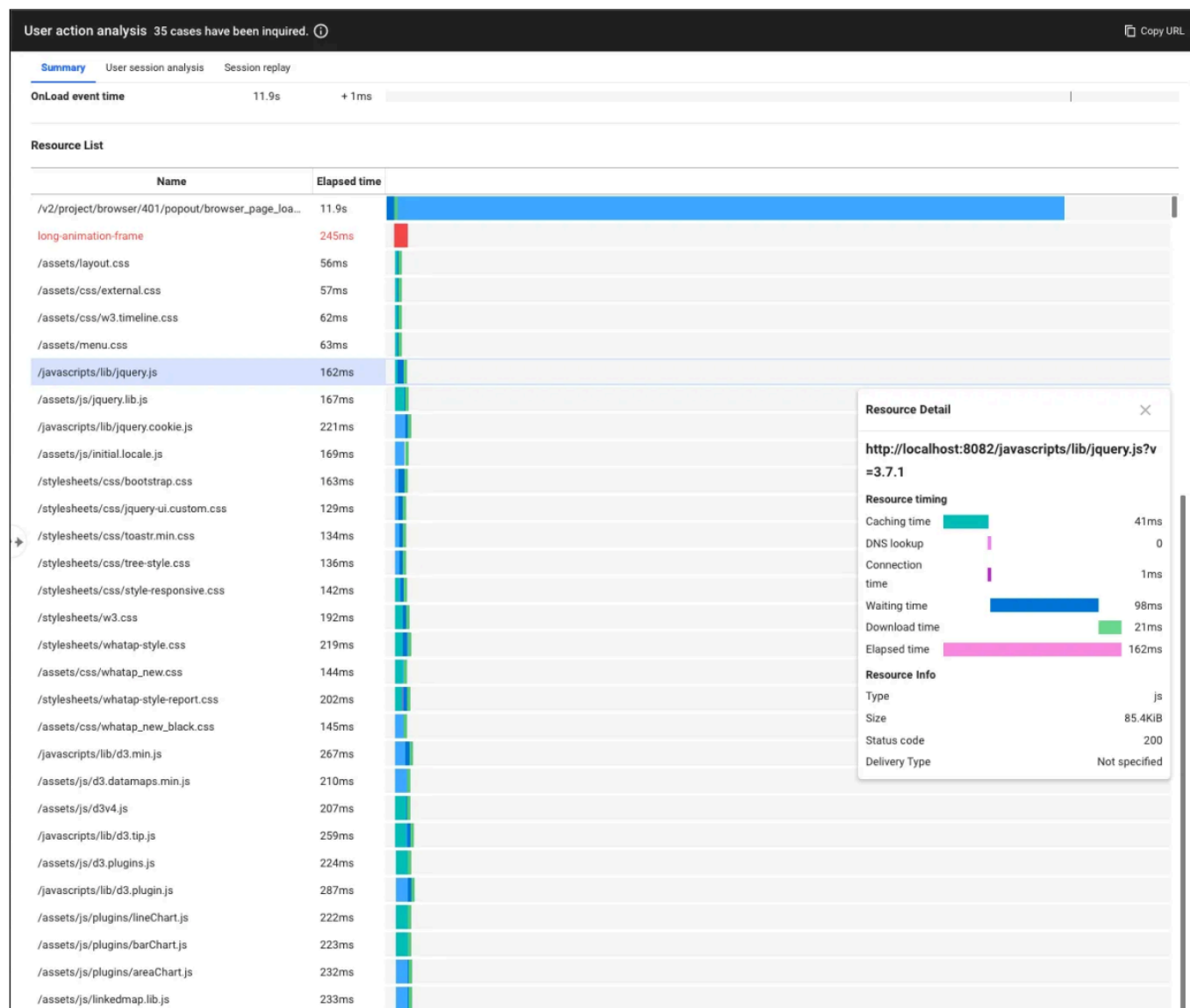
- The analysis uses only the information already shown on the screen, so no additional query occurs.
- Only some of the resources are sent, prioritizing the ones that took longest. The number sent is shown together with the total number.
- If an APM-linked transaction exists, whether it is linked is also taken into account.
- If you select another event in the list, the analysis result for the previous event is cleared.

ⓘ For how AI Analysis behaves in common and the data scope it uses, refer to [the following document](#).

Resource list

The detailed information about the resources loaded while the event was processed is provided. Each resource appears in a timeline so that you can see the loading order and the time taken at a glance.

- **Name:** File name or URL of the resource
- **Elapsed time:** Loading time of each resource



Resource details

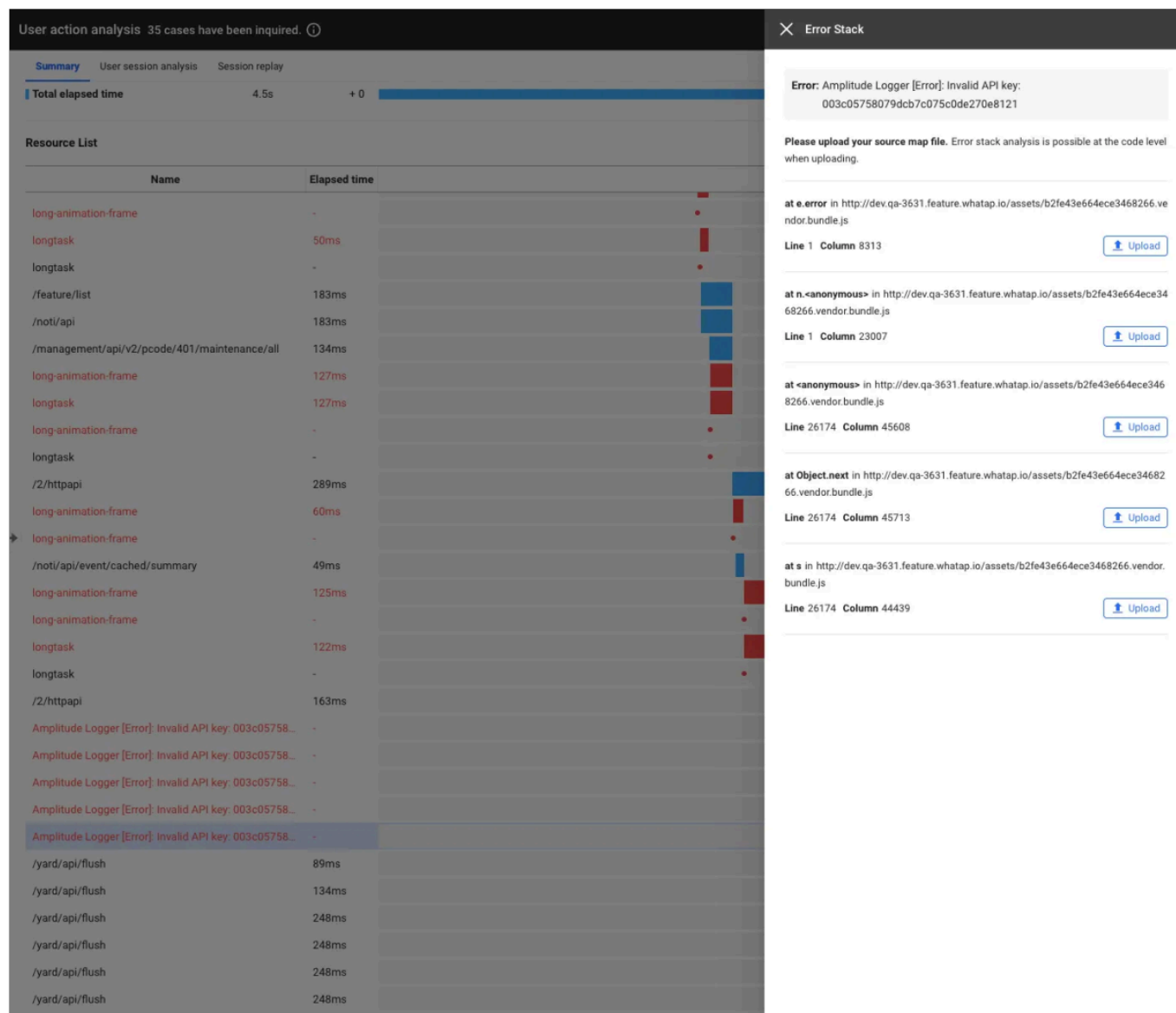
Click a resource in the resource list to see its detailed information in a tooltip. The resource details let you pinpoint performance bottlenecks and identify the resources that need optimization.

- **Resource timing information**
 - **Caching time:** Time taken to look up cached resources
 - **DNS lookup:** Time taken to look up the domain

- **Connection time:** Time taken to establish the connection to the server
- **Request waiting time:** Time waiting for the server response (TTFB)
- **Download time:** Time taken to download the resource
- **Elapsed time:** Total time taken to load the resource
- **Resource information**
 - **Type:** Resource type (css, js, image, and so on)
 - **Size:** File size of the resource
 - **Delivery Type:** Delivery method (cache load, and so on)

Checking the browser errors

A page load event that contains browser error data appears in **red**. You can see the steps in which errors occurred in **Resource List**.



User session analysis

Based on the user session, you can see the events and errors that occur while interacting with the website in chronological order. For more information about **User session analysis**, see [the following](#).

Session replay

It provides the feature to record and replay all events that users perform on the website. The **Session replay** feature allows you to reproduce user behaviors such as clicking, scrolling, typing, and page switching. This helps you understand exactly how users actually interact with the website.

- For information about application of the **Session replay** feature by using agent options, see [the following](#).
- For more information about using the **Session replay** feature, see [the following](#).

AJAX HITMAP

Home > Select Project > **Analysis** > **AJAX Hitmap**

The **Asynchronous JavaScript and XML (AJAX)** technology exchanges data and updates web pages in an asynchronous manner. AJAX allows you to create a more complex web page that update part of the page without reloading the entire page. AJAX also allows you to work asynchronously, enabling the code to continuously run while reloading part of the web page.

AJAX has advantages such as user experience improvement and reduction of requests to the server. However, because AJAX is implemented with JavaScript code running in the browser, it requires monitoring for the following reasons:

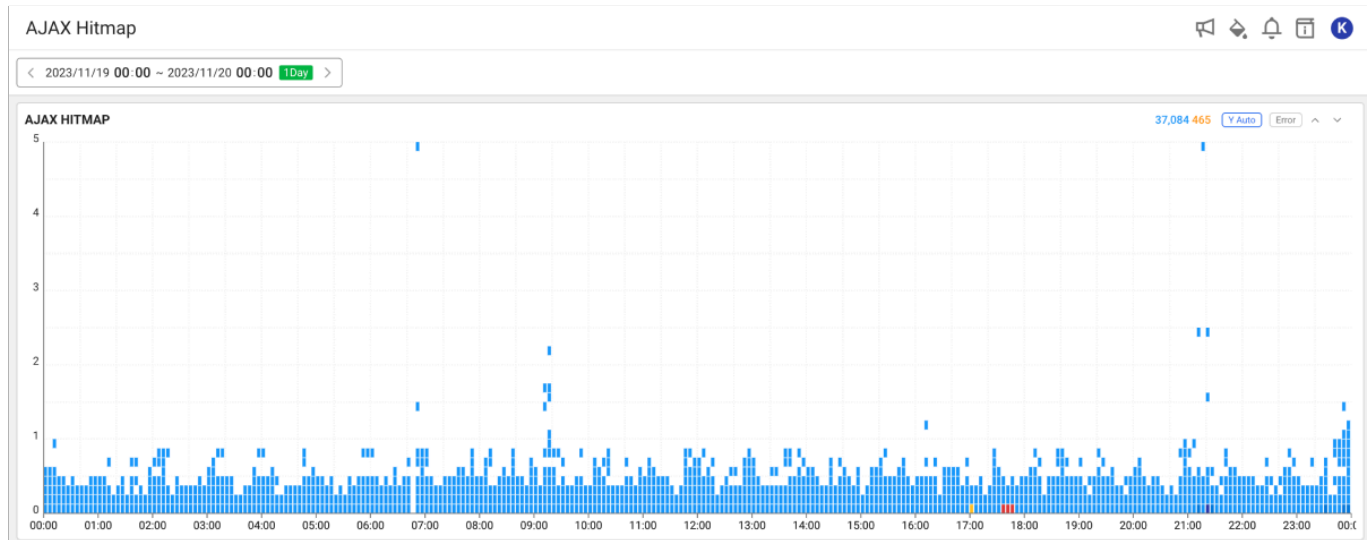
- You have to make sure that AJAX requests are handled properly.
- You have to check the performance degradation caused by AJAX requests.
- You have to understand the user-perceived performance including the network latency, download time, and such.

In the **AJAX Hitmap** menu provided from **Browser Monitoring**, you can see the response times of AJAX events over time on a hitmap chart. You can also see the detailed information for each event. The provided information is as follows:

- Detailed time information for each AJAX event section
- AJAX request URL, calling page information, etc.
- Error message and status code

Basic screen guide

AJAX Hitmap



This widget displays the response times for AJAX events over time in a distribution chart.

- The horizontal axis is the AJAX event end time.
- The vertical axis is the AJAX event's elapsed time.
- The normal **AJAX** events are displayed in the order of **Light Blue** → **Blue** → **Indigo**.
- The error occurrence **AJAX** events are displayed in the order of **Yellow** → **Orange** → **Red**. An error **AJAX** event occurs when the status code is **0** or more than **400**. If the status code is **0**, the request is not passed from the browser to the server.

AJAX Hitmap AI Analysis

Click the **AI Analysis** button at the right end of the option bar to have WhaTap AI read the hitmap distribution and report what it finds. The analysis works at the individual request level rather than the page level, so you can use it to narrow down which endpoint is slow. The button shows an icon only; hover over it to see its name in a tooltip.

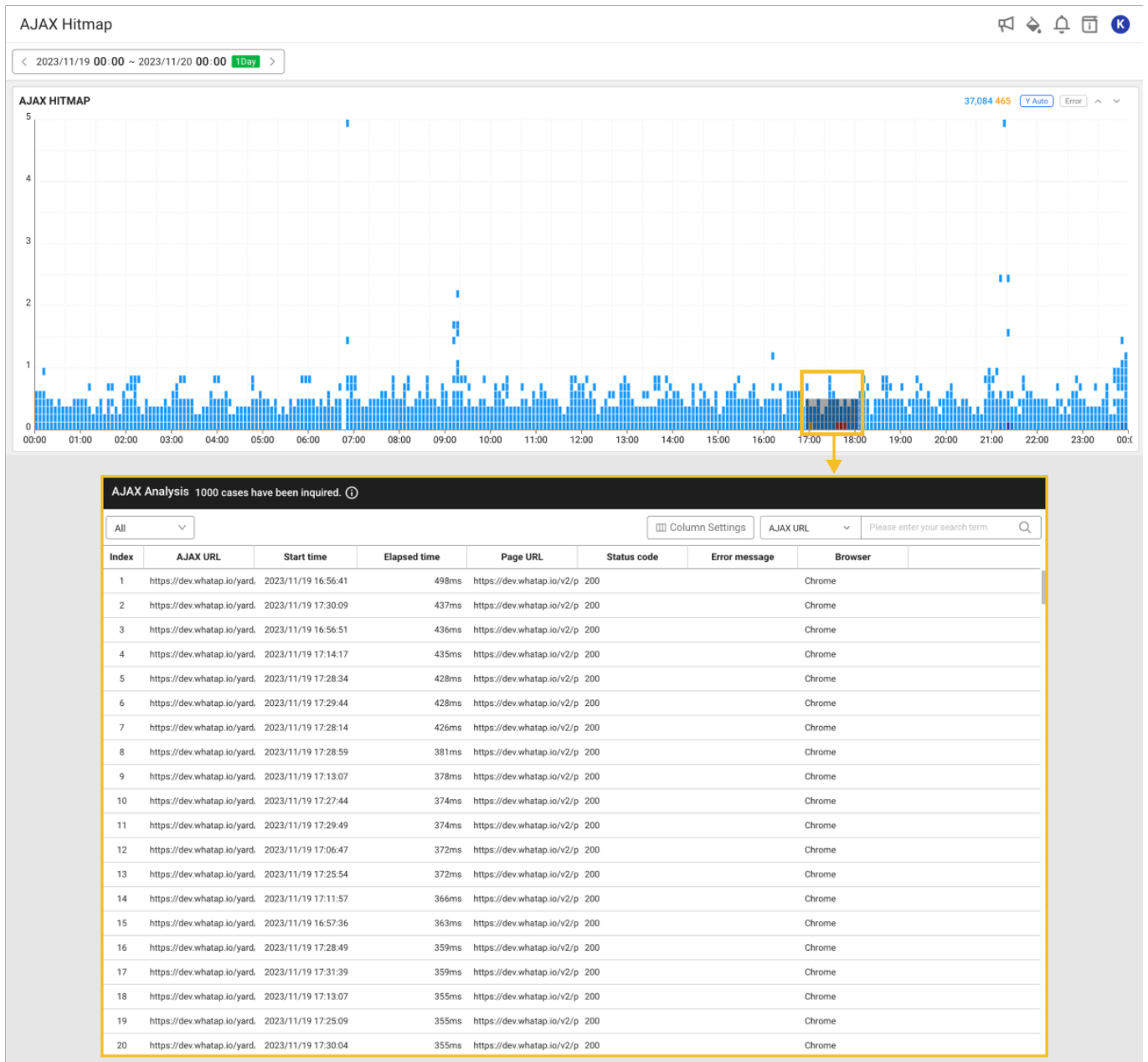
- **No area selected:** WhaTap AI reads the whole hitmap for the queried period and suggests the time ranges

and response time ranges to check.

- **Area dragged:** WhaTap AI analyzes the AJAX events inside the selected area and presents candidate causes of the slowdown. The selected area appears to the left of the button. To clear it, click the **X** icon on the item.
- If you run the analysis while the errors-only view is applied, that condition also applies to the analysis.
- If the queried period contains no event, the button is disabled. Hover over it to see the reason,
No data in the selected period.

ⓘ For how AI Analysis behaves in common and the data scope it uses, refer to [the following document](#).

AJAX List



If any part is dragged in the **AJAX Hitmap** chart, the **AJAX Analysis** window appears to display the AJAX event list and details of the dragged area.

The following show the descriptions for columns in the table.

- **AJAX URL:** AJAX event URL
- **Start time:** AJAX request start time
- **Elapsed time:** Elapsed time until the AJAX event completion
- **Page URL:** URL of the AJAX event called page
- **Status code:** Response status code of the AJAX event
- **Error message:** AJAX response value (`statusText`) when the status code is `0` or more than `400`

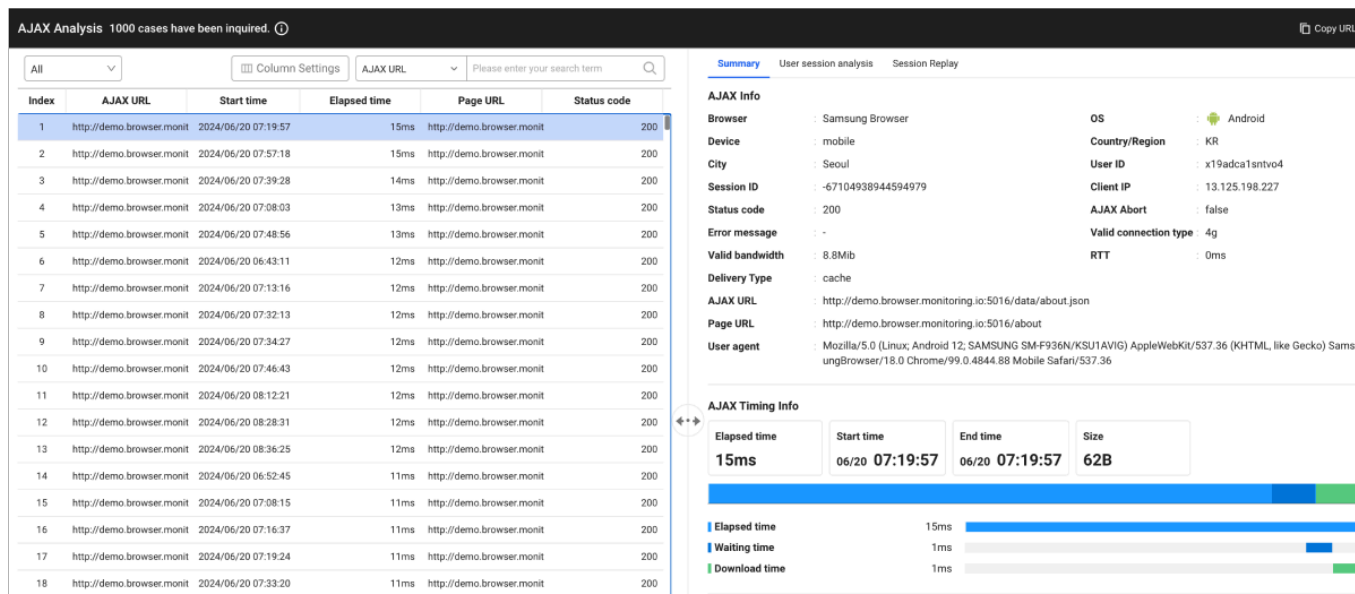
! AJAX events whose status code is `0` or more than `400` appear in **red** in the list.

- **Browser:** Type of the browser used by the end user

! For more information about the column that can be added using the **Add a column** button, see [the following](#).

Checking the AJAX event details

If you select each item in the **AJAX Analysis** window, the **AJAX Info** window appears on the right to display the detailed information.



The times required for each section is provided in a graph chart while loading the AJAX events. The chart helps you identify slow sections while loading AJAX events and enhance them.

To see only the AJAX list that has errors, on the upper left of the **AJAX Info** window, select from the **Error All** choose list.

AJAX information

You can see the properties of the AJAX event. You can see the information such as AJAX request URL, end user access environment, access page information, **User ID**, **Session ID**, and **Client IP**.

! WhaTap basically stores the client-related information.

AJAX timing information

The AJAX event loading process is divided into the following steps and the times required for each section are displayed.

- **Redirection time:** Time taken by the browser to redirect an AJAX request
- **Caching time:** Time taken by the browser to retrieve cached AJAX resources

- **DNS lookup:** Time taken by the browser to retrieve the AJAX request domain
- **Connection time:** Time taken to establish TCP connection with the AJAX request target server
- **SSL time:** Time taken to complete the SSL/TLS handshake
- **Waiting time:** Time taken from the start of the browser's HTTP request to the time when the server receives the request and begins responding. The transaction processing time in the server is included in this section.
- **Download time:** Time taken by the browser to download the AJAX response values from the server

User session analysis

AJAX Analysis 1000 cases have been inquired. ⓘ						Copy URL	
All Column Settings AJAX URL Please enter your search term							
Index	AJAX URL	Start time	Elapsed time	Page URL	Status code		
1	http://demo.browser.monit	2024/06/20 07:19:57	15ms	http://demo.browser.monit	200		
2	http://demo.browser.monit	2024/06/20 07:57:18	15ms	http://demo.browser.monit	200		
3	http://demo.browser.monit	2024/06/20 07:39:28	14ms	http://demo.browser.monit	200		
4	http://demo.browser.monit	2024/06/20 07:08:03	13ms	http://demo.browser.monit	200		
5	http://demo.browser.monit	2024/06/20 07:48:56	13ms	http://demo.browser.monit	200		
6	http://demo.browser.monit	2024/06/20 06:43:11	12ms	http://demo.browser.monit	200		
7	http://demo.browser.monit	2024/06/20 07:13:16	12ms	http://demo.browser.monit	200		
8	http://demo.browser.monit	2024/06/20 07:32:13	12ms	http://demo.browser.monit	200		
9	http://demo.browser.monit	2024/06/20 07:34:27	12ms	http://demo.browser.monit	200		
10	http://demo.browser.monit	2024/06/20 07:46:43	12ms	http://demo.browser.monit	200		
11	http://demo.browser.monit	2024/06/20 08:12:21	12ms	http://demo.browser.monit	200		
12	http://demo.browser.monit	2024/06/20 08:28:31	12ms	http://demo.browser.monit	200		
13	http://demo.browser.monit	2024/06/20 08:36:25	12ms	http://demo.browser.monit	200		
14	http://demo.browser.monit	2024/06/20 06:52:45	11ms	http://demo.browser.monit	200		
15	http://demo.browser.monit	2024/06/20 07:08:15	11ms	http://demo.browser.monit	200		
16	http://demo.browser.monit	2024/06/20 07:16:37	11ms	http://demo.browser.monit	200		
17	http://demo.browser.monit	2024/06/20 07:19:24	11ms	http://demo.browser.monit	200		

Summary

User session analysis

Session Replay

User session information

Browser Samsung Browser

OS Android

Device mobile

User ID x19adca1sntvo4

Session ID -67104938944594979

IP 13.125.198.227

User agent Mozilla/5.0 (Linux; Android 12; SAMSUNG SM-F936N/KSU1AVIG) AppleWebKit/537.36 (KHTML, like Gecko) SamsungBrowser/18.0 Chrome/99.0.4844.88 Mobile Safari/537.36

User session flow

Start time	Category	Title	Page URL	Elapsed time
2024/06/20 07:14:53	#UserEvent	Custom: start_blog	http://demo.browser.monitoring.io:5	-
2024/06/20 07:14:53	#UserEvent	Custom: purchase_product	http://demo.browser.monitoring.io:5	-
2024/06/20 07:14:53	#UserEvent	Custom: start_blog	http://demo.browser.monitoring.io:5	-
2024/06/20 07:14:53	#UserEvent	Custom: purchase_product	http://demo.browser.monitoring.io:5	-
2024/06/20 07:14:53	#UserEvent	Custom: start_blog	http://demo.browser.monitoring.io:5	-
2024/06/20 07:14:53	#UserEvent	Custom: purchase_product	http://demo.browser.monitoring.io:5	-
2024/06/20 07:14:53	#UserEvent	Custom: start_blog	http://demo.browser.monitoring.io:5	-
2024/06/20 07:14:53	#UserEvent	Custom: purchase_product	http://demo.browser.monitoring.io:5	-
2024/06/20 07:14:53	#UserEvent	Custom: start_blog	http://demo.browser.monitoring.io:5	-
2024/06/20 07:14:53	#UserEvent	Custom: purchase_product	http://demo.browser.monitoring.io:5	-
2024/06/20 07:14:53	#UserEvent	Custom: start_blog	http://demo.browser.monitoring.io:5	-

Based on the user session, you can see the events and errors that occur while interacting with the website in chronological order. For more information about **User session analysis**, see [the following](#).

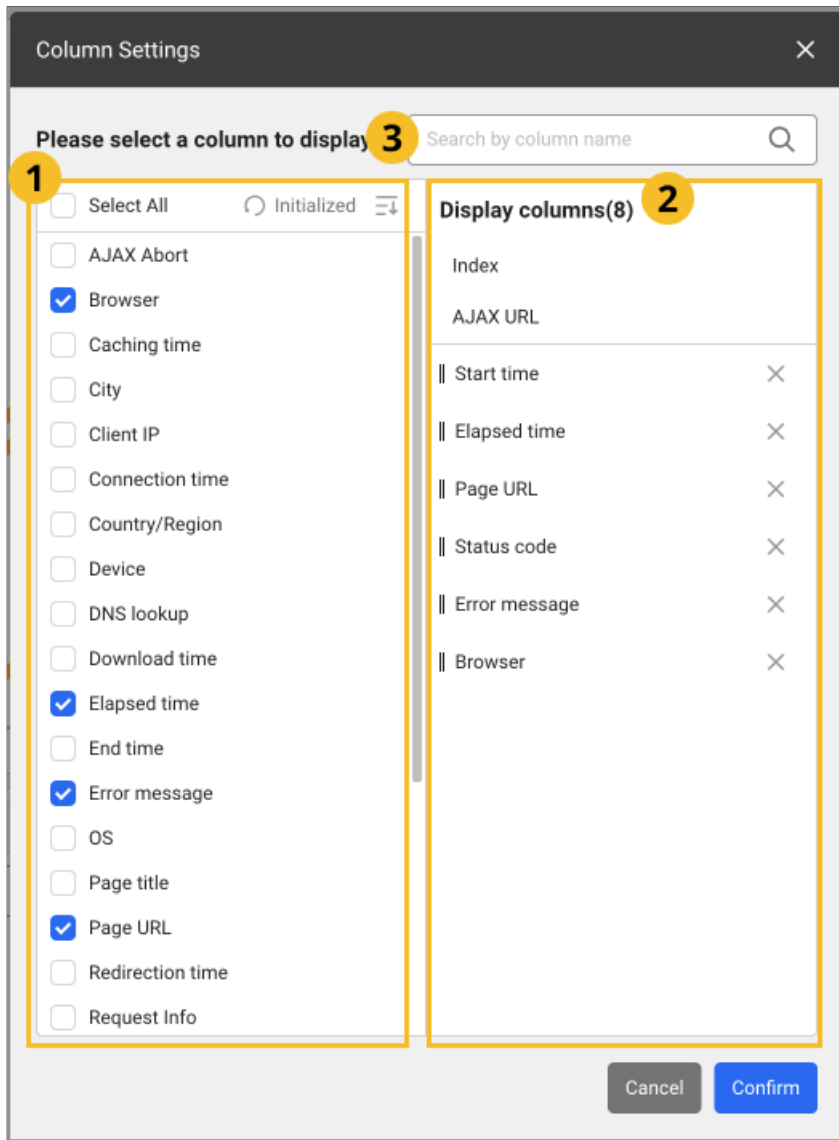
AJAX URL ▾	Please enter your search term 	
AJAX URL	Browser	Request In
Page URL	Samsung Browser	
Browser	Chrome	
OS	Samsung Browser	
Device	Samsung Browser	
Client IP	Internet Explorer	
User ID	Safari	

The query criteria that can be selected for filtering are as follows:

- **AJAX URL**
- **Page URL**
- **Browser**
- **OS**
- **Device**
- **Client IP**
- **User ID**

Setting the table columns

You can hide table header columns in the AJAX list or add desired items. You can also change the column order. Select [Column settings](#).



- After configuration, select **OK** to apply the settings in the table.
- In the **3** search bar, enter text to search the desired columns. Only the columns that meet the entered text are displayed.

Adding columns

From the **1** list, select the items to add as table header columns. To select all items, select **Select All**. The column

items that can be added are as follows:

Column name	Description
Client IP	IP address of the device used by the end user
AJAX Abort	If the AJAX request is stopped in the browser level, <code>true</code> appears.
User agent	End user's agent string collected by the browser agent
Size	Capacity for AJAX response values indicating the actual uncompressed capacity
Country/Region	End user's country code
City	End user's city name
Download time	Time taken by the browser to download the AJAX response values from the server
Device	End user's device type
Redirection time	Time taken by the browser to redirect AJAX requests
User ID	End user ID assigned randomly
Session ID	End user's session ID
Connection time	Time taken to establish TCP connection with the target server of the AJAX request
Waiting time	Time elapsed from the start of the browser's HTTP request to the time the server receives the request and starts responding
Browser	Type of the browser used by the end user
OS	OS type of the device used by the end user
End time	AJAX event end time

Column name	Description
Caching time	Time taken by the browser to search the cached AJAX resources
Page title	Title of the HTML page accessed through the browser
DNS lookup	Time taken by the browser to search the AJAX request domain
SSL time	Time taken to complete the SSL/TLS handshake
Valid connection type	Network performance measured after the browser accesses the page
Valid bandwidth	Value rounded to the 25 kilobits per second whose downlink speed is in Mbps.
Delivery Type	How to send resources (e.g. <code>cache</code> or <code>navigational-prefetch</code>)
RTT	The round-trip time (RTT) is the time (in milliseconds) taken to initiate a network request and then receive the response.

ⓘ WhaTap basically stores the client-related information.

Deleting columns

From the ❶ list, unselect the columns to delete. Alternatively, select ✕ on the right of the item to delete from the ❷ list.

Changing the column order

Drag an item to reposition from the ❷ list, and then move it to the desired position.

Initializing the configuration

To cancel all changes and reset them, select ↺ [Reset](#).

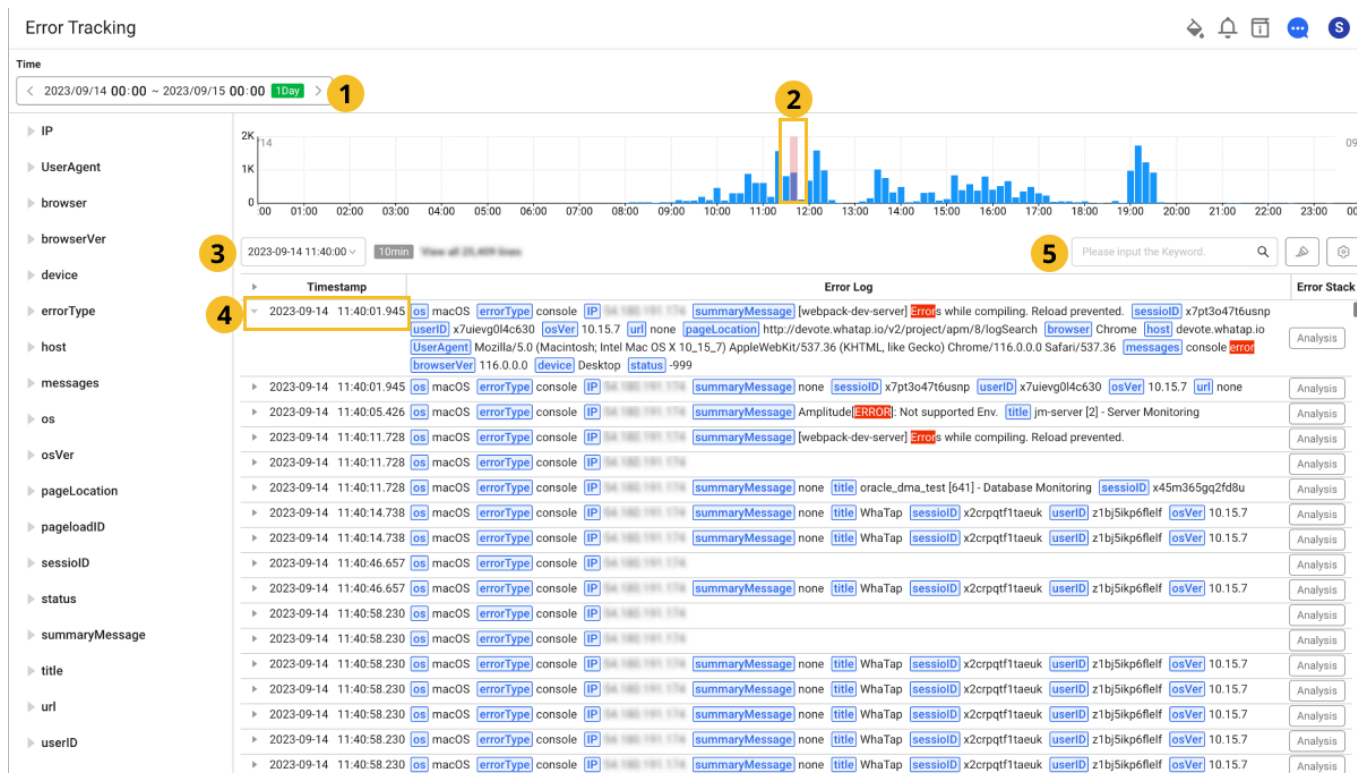
Error Tracing

Home > Select Project > **Analysis** > **Error Tracking**

Browser Monitoring collects the browser front-end errors. It includes the error messages and error stack information, browser, OS, device, and page URL.

Exploring a error

You can see browser errors that have occurred over time.



1. In ①, set a desired time. It can be set up to 1 day in advance.
2. Select one of the ② bar graphs displayed on the chart.

From the ③ time list, you can select the time detail when an error occurred among the data collected every 10 minutes.

3. Below the chart, you can see a list of browser errors that occurred at the selected time.
If you type specific keywords in the 5 input field, the matching text is highlighted in the list.
4. To check the details of the error, select 4 ► on the utmost left.

Checking the error information details

The browser error has the following information:

- **errorType**: Indicates the error type. There are source errors including AJAX errors, console errors, and syntax errors.
- **pageLocation**: URL of the page visited by the user.
- **title**: Title of the page visited by the user.
- **browser**: Browser type of the user.
- **browserVer**: Browser version of the user.
- **os**: OS type of the user device.
- **osVer**: OS version of the user device.
- **device**: Type of the user device.
- **host**: Domain name of the visited server.
- **userAgent**: User-Agent string received from the user browser.
- **message**: Error message.
- **stack**: Error stack information. The error stack basically contains obfuscated file names, lines, and columns.

Source map uploading and error code provision

If the released frontend code is obfuscated or compressed, it is difficult to provide the collected browser error's stack information at an actual code level. You can access the file or code line that caused the browser error through the source-map built at the time of releasing the frontend code.

1. In the **Exception stack** column, select **Analysis**.

Timestamp	Error Log	Error Stack
2023-09-14 11:40:01.945	os macOS errorType console IP 116.0.0.0 summaryMessage [webpack-dev-server] Errors while compiling. Reload prevented. sessionId x7pt3o47f6usnp userId x7uevgll4c630 osVer 10.15.7 url none pageLocation http://devote.whatap.io/v2/project/apm/8/logSearch browser Chrome host devote.whatap.io UserAgent Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/116.0.0.0 Safari/537.36 messages console error Analysis	

- If the window in which the error stack information for the selected error log is displayed appears, select **Upload**.

Error Tracking

Time

2023-09-14 00:00 - 2023-09-15 00:00

Libs

IP

UserAgent

browser

browserVer

device

errorType

host

messages

os

osVer

pageLocation

pageLoadID

sessionId

status

summaryMessage

title

url

userId

2023-09-14 11:40:01.945

os

macOS

errorType

console

IP

116.0.0.0

summaryMessage

[webpack-dev-server] Errors while compiling. Reload prevented.

sessionId

x7pt3o47f6usnp

userId

x7uevgll4c630

osVer

10.15.7

url

none

pageLocation

http://devote.whatap.io/v2/project/apm/8/logSearch

browser

Chrome

host

devote.whatap.io

UserAgent

Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/116.0.0.0 Safari/537.36

messages

console

error

Analysis

Error Stack

Error: [webpack-dev-server] Errors while compiling. Reload prevented.

Please upload your source map file. Error stack analysis is possible at the code level when uploading.

at logger in webpack-internal:///.../node_modules/webpack-dev-server/client/module.js

Line 493 Column 23

Upload

at WebpackLogger.eval in webpack-internal:///.../node_modules/webpack-dev-server/client/modules/logger/index.js

Line 634 Column 7

Upload

at WebpackLogger.error in webpack-internal:///.../node_modules/webpack-dev-server/client/modules/logger/index.js

Line 156 Column 23

Upload

at Object.error in webpack-internal:///.../node_modules/webpack-dev-server/client/index.js

Line 292 Column 52

Upload

at eval in webpack-internal:///.../node_modules/webpack-dev-server/client/socket.js

Line 70 Column 29

Upload

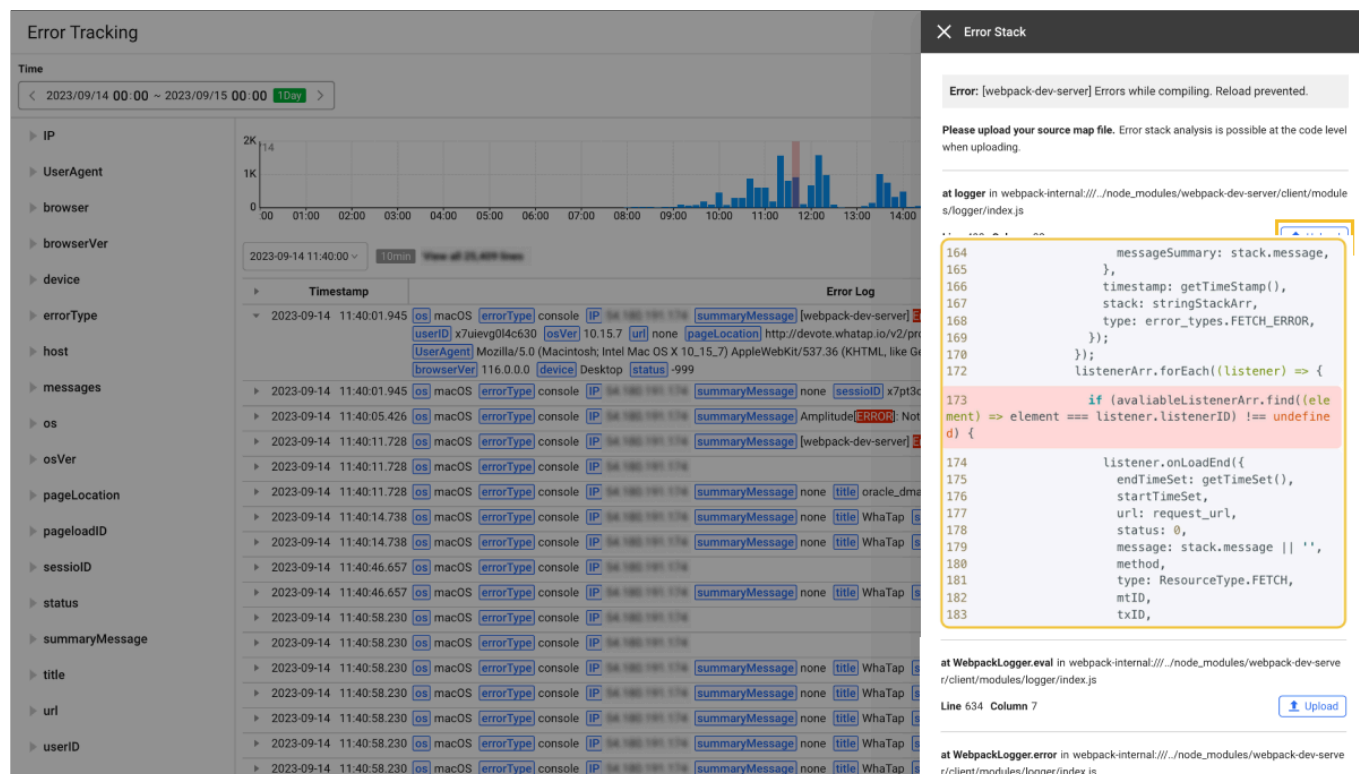
at client.onmessage in webpack-internal:///.../node_modules/webpack-dev-server/client/modules/WebSocketClient.js

Line 10 Column 9

Upload

- Load the source map for the code stack.

By comparing obfuscated stack information with JavaScript source map, you can see the code where the actual error occurred.



AI Error Stack Analysis

Beta

Click the **Error Stack Analysis** button in the error stack area to have AI automatically analyze error stacks, environment information (browser, OS, device), user behavior logs, and error statistics. It identifies the root cause and impact scope of errors, and suggests solutions.

Analyze

1. Go to **Analysis > Error Tracking**.
2. Select an error from the error list to open the error detail view.
3. Click the **Error Stack Analysis** button in the error stack area.

Analysis results

AI collects and analyzes the following data. Available data may vary depending on project settings and error conditions.

Analysis targets

Data	Description
Error stack	Uses restored stack frames if a source map is available, otherwise uses obfuscated stack information.
Environment info	Uses browser, OS, device, and screen size information included in error logs.
Error statistics	Aggregates occurrence counts by page, browser, OS, and device based on the error date.
Session logs	Collects user behavior logs before the error (page navigation, clicks, failed API calls, etc.).

Result structure

The analysis results consist of the following 4 sections:

- **Summary:** Summarizes the key details of the error in 2-3 sentences.
- **Cause Analysis:** Classifies and displays error causes by severity (**High**, **Medium**, **Low**).

Severity	Criteria
High	Unhandled exceptions causing app crashes, data loss, security vulnerabilities
Medium	Impacts user experience but workarounds exist
Low	Minor issues, warnings, cosmetic errors

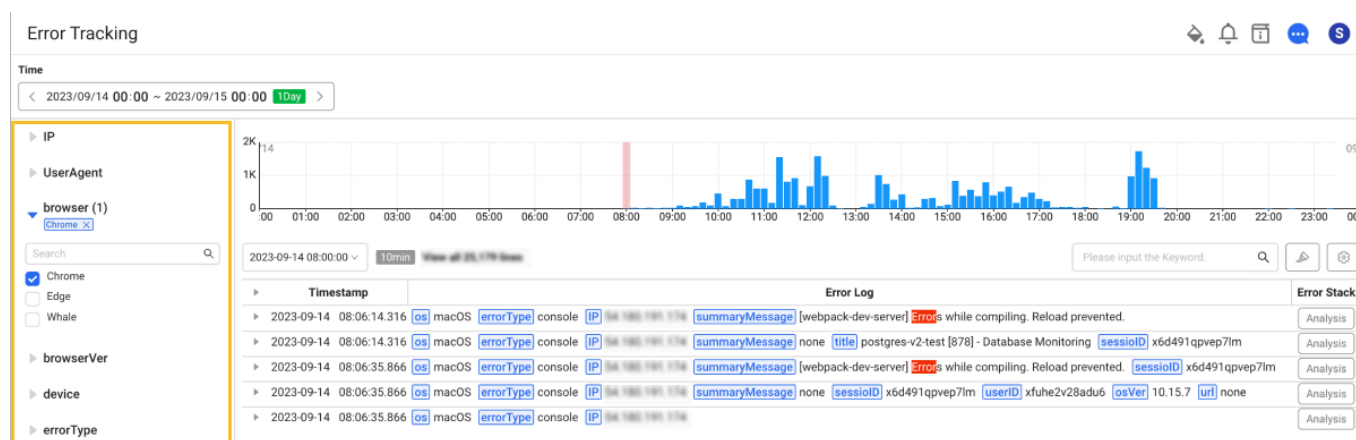
- **Impact Scope:** Displays the affected browsers, OS, pages, devices, and occurrence frequency.
- **Resolution:** Provides action items by priority along with code fix examples.

Priority	Criteria
High	Immediate fix required — affects core features or a large number of users
Medium	Early fix recommended — impacts user experience
Low	Improvement recommended — minor enhancements

- The analysis results are generated by AI and may differ from actual conditions. Use them as reference material.
- If a source map is uploaded, the analysis uses the restored stack information.
- Available only in SaaS environments.
- **Member** role or above is required.

Searching by filter the error data

You can search the errors by filtering based on the data included in the browser errors.



1. In the left panel of the screen, select a desired browser error item for filtering.
2. If the sub list appears, select a desired item. You can also search if the list is too many.

It displays a list of browser errors corresponding to the selected item.

Table Column Settings

Displaying or hiding the table columns

You can display or hide column header items in the browser error list table.

1. Select  on the upper right of the table.
2. If the **Column settings** window appears, select or unselect the desired item in the list on the left.
You can also hide columns by selecting the  button in the list on the right.
3. After all settings are finished, select **Confirm**. The set content is applied to the table.

Reordering the table columns

You can change the order of column headers in the browser error list table.

1. Select  on the upper right of the table.
2. When the **Column settings** window appears, change the order by dragging the desired items from the list on the right.
3. After all settings are finished, select **Confirm**. The set content is applied to the table.

Setting to display the table content

You can set the content displayed in the browser error list. It can be used when too much information appears so it is difficult to see, or it is required to add necessary information.

1. Select  on the upper right of the table.
2. When the **Log view detail setting** window appears, set the desired options.
 - Select **content** or **Tag** to determine whether or not to display the log content.

- **Tag management:** Tags are displayed in the log in the order of addition. The order of tags can be changed by dragging. Disabled tags do not appear in the logs.
3. After all settings are finished, click **×**. The set content is applied to the table.

! For more information about **Log view detail setting**, see [the following](#).

Searching The User Session Log

Home > Select Project > **Analysis** > **User Session Log Search**

End users can search using various conditions and identify the user session log by collecting the performance data, error data, and click data when interacting with the website. User session logs help analyze end user paths, identify reasons for user attrition, and analyze the performance and user experience.

User Session Log Search

Time

< 2024/03/20 13:26 ~ 2024/03/20 14:26 60Min >

Filter

type == custom X name == purchase_product X Enter Filter and press enter to add.

type == custom && name == purchase_product

#BrowserError (4)

#BrowserError IP browser browserVersion browseragenttime.n device errorType host messages os osVersion pageLocation pageUri pageLoadID screenHeight.n screenWidth.n sessionID status summaryMessage title type url userAgent userID

#PerformanceEvent (26)

#PerformanceEvent IP browser browserVersion browseragenttime.n device os osVersion pageUri pageUriPath screenHeight.n screenWidth.n sessionID type userAgent userID

#UserEvent (504)

#UserEvent IP browser browserVersion browseragenttime.n customDuration.n device duration.n name os osVersion pageUri pageUriPath screenHeight.n screenWidth.n sessionID type userAgent userID

Of the total 534 results, 144 match your search.

Oldest Newest Please input the Keyword.

Timestamp	Log	User session analysis
2024-03-20 13:52:09.443	os macOS screenHeight.n 934 pcode 421 IP 13.125.196.206 userAgent Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/122.0.0.0 Safari/537.36 sessionID 2162061845021904416 type custom userID x5jfa4n0a5tf3d duration.n 3670 osVersion 10.15.7 browseragenttime.n 1710910322272 customDuration.n -1 pageUriPath / browser Chrome browserVersion 122.0.0.0 name purchase_product agenttime 1710910326340 screenWidth.n 1920 pageUri http://localhost:8000/ category #UserEvent device Desktop { "order": "sample_order_1", "cost": 12345 }	Analysis
2024-03-20 13:52:09.443	os macOS screenHeight.n 934 pcode 421 IP 13.125.196.206 { "order": "sample_order_1", "cost": 12345 }	Analysis
2024-03-20 13:52:09.443	os macOS screenHeight.n 934 pcode 421 IP 13.125.196.206 { "order": "sample_order_1", "cost": 12345 }	Analysis
2024-03-20 13:52:14.627	os macOS screenHeight.n 934 pcode 421 IP 13.125.196.206 { "order": "sample_order_1", "cost": 12345 }	Analysis
2024-03-20 13:52:19.627	os macOS screenHeight.n 934 pcode 421 IP 13.125.196.206 { "order": "sample_order_1", "cost": 12345 }	Analysis
2024-03-20 13:52:19.627	os macOS screenHeight.n 934 pcode 421 IP 13.125.196.206 { "order": "sample_order_1", "cost": 12345 }	Analysis
2024-03-20 13:52:19.627	os macOS screenHeight.n 934 pcode 421 IP 13.125.196.206 { "order": "sample_order_1", "cost": 12345 }	Analysis
2024-03-20 13:52:24.299	os macOS screenHeight.n 934 pcode 421 IP 13.125.196.206 { "order": "sample_order_1", "cost": 12345 }	Analysis
2024-03-20 13:52:24.299	os macOS screenHeight.n 934 pcode 421 IP 13.125.196.206 { "order": "sample_order_1", "cost": 12345 }	Analysis
2024-03-20 13:52:24.299	os macOS screenHeight.n 934 pcode 421 IP 13.125.196.206 { "order": "sample_order_1", "cost": 12345 }	Analysis
2024-03-20 13:52:29.484	os macOS screenHeight.n 934 pcode 421 IP 13.125.196.206 { "order": "sample_order_1", "cost": 12345 }	Analysis
2024-03-20 13:52:29.484	os macOS screenHeight.n 934 pcode 421 IP 13.125.196.206 { "order": "sample_order_1", "cost": 12345 }	Analysis
2024-03-20 13:52:29.484	os macOS screenHeight.n 934 pcode 421 IP 13.125.196.206 { "order": "sample_order_1", "cost": 12345 }	Analysis

Set the search time to view in the **Time** option, and then select . You can set multiple search conditions consisting of keys and values in **Filter** and then extract only the log data that matches the conditions. Log data is loaded by lines and displayed in a list. When you go to the bottom of the list, the following data is automatically retrieved for display.

✔ Basic operations in **User Session Log Search** is the same as those of the **Log Search** menu. For more information about the search methods and other options by setting filters, see the following.



- For more information about the log search using the filter settings, see [the following](#).
- For more information about tag options, see [the following](#).
- For more information about the content highlighting, see [the following](#).
- For more information about the **Table settings** function, see [the following](#).



- This function can be used by the members with the **log search** role. For more information about the member role scheme, see [the following](#).

Guide to major search tags

#Category

The user session logs collected by the browser monitoring can be categorized as follows: Each **#Category** has a list of tags for searching by filters.

▼ #BrowserError (4)

- #BrowserError IP browser
- browserVersion browseragenttime.n
- device errorType host messages
- os osVersion pageLocation
- pageUri pageloadID screenHeight.n
- screenWidth.n sessionID status
- summaryMessage title type url
- userAgent userID

▼ #PerformanceEvent (26)

- #PerformanceEvent IP browser
- browserVersion browseragenttime.n
- device os osVersion pageUri
- pageUriPath screenHeight.n
- screenWidth.n sessionID type
- userAgent userID

▼ #UserEvent (644)

- #UserEvent IP browser
- browserVersion browseragenttime.n
- customDuration.n device duration.n
- name os osVersion pageUri
- pageUriPath screenHeight.n
- screenWidth.n sessionID type
- userAgent userID

- **#BrowserError**: Collects and provides source errors, network errors, and such that occur from the website.
- **#PerformanceEvent**: Collects and provides events that are related with page loading, AJAX, and router change.
- **#UserEvent**: Collects and provides user click events and custom events.

User session log

User session logs include the end-user environment and the data collected by **#Category**. If you select ▶ on the utmost left, you can see the whole data that makes up the log.

▶	2024-03-21	13:25:04.200	os	macOS	screenHeight.n	934	pcode	421	IP	13.125.196.206	{ "width": 94, "height": 24, "selector": "#gatsby-focus-wrapper>DIV.global-wrapper>HEADER.global-header>A.header-link	
▼	2024-03-21	13:25:04.200	os	macOS	screenHeight.n	934	pcode	421	IP	13.125.196.206	userAgent Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/122.0.0.0 Safari/537.36	
					sessionID	-6827335224841178182	type	ajax	userID	x5jfa4n0a5tf3d	osVersion	10.15.7
					browseragenttime.n	1710995096925	pageURLPath	/new-beginnings/	browser	Chrome	browserVersion	122.0.0.0
					agenttime	1710995101977	screenWidth.n	565	pageURL	http://localhost:8000/new-beginnings/	category	#PerformanceEvent
					device	Desktop	{ "total": 25, "txid": "-2876790671108083878", "mtid": "0", "status": 200 }					

The content of the first line consists of the end user's environment and **#Category**-based collection data as keys and values. Keys and values are used for search in **Filter**. The second line is the message of the user's session log.

os	macOS	screenHeight.n	934	pcode	421	IP	13.125.196.206					
userAgent	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/122.0.0.0 Safari/537.36											
sessionID	2162061845021904416	type	pageLoad	userID	x5jfa4n0a5tf3d	osVersion	10.15.7	browseragenttime.n	1710910172056			
pageURLPath	/	browser	Chrome	browserVersion	122.0.0.0	agenttime	1710910173475	screenWidth.n	1920	pageURL	http://localhost:8000/	
category	#PerformanceEvent											
device	Desktop											
{ "total": 928, "pageTxid": "4217223616367407890" }												

The following are descriptions for major keys in the user session log.

- **PageUrl** : URL of the page where the event occurred.
- **userID** : ID of the user who generated the event. A unique ID that is issued by the browser agent when a user accesses a website for the first time. It has no expiration time. IDs are stored in browser cookies.
- **sessionID** : ID of the session that generated the event. A unique ID issued when a user starts navigation of the website. The maximum duration of a session is 4 hours, and the ID expires after 15 minutes of inactivity. IDs are stored in browser cookies.
- **type** : Type of the generated event.
 - **pageLoad**: Event that occurs upon loading the website.
 - **routeChange**: Event that occurs upon change in the route of the SPA website.
 - **ajax**: AJAX request-related event
 - **click**: Event that occurs when an actual user clicks on the website.
 - **rapid_click**: Event that occurs when an actual user quickly clicks on a single coordinate multiple times.

- **custom:** Custom event.
- **browser_error:** Source errors, network errors, and the like in the website.

- ! • For the agent configuration option (`collectUserClick: true`) to collect user click events, see [the following](#).
- For more information about collection of custom events, see [the following](#).

Analyzing the user session

In the viewed user session log list, select **Analysis** in the **User session analysis** column. The **User session analysis** window appears where you can see the details of the user session log for the selected item.

Based on the user session, you can see the events and errors that occur while interacting with the website in chronological order. You can use the user session analysis as the following and enjoy better user experience.

- **On-page behavior analysis:** You can understand how the user navigates the website. It can analyze user activities on how you navigate, which pages you stay on, and which elements you click on. This helps you gain insight into how users use the website and understand the user behavior patterns by analyzing user paths.
- **Error tracing:** You can trace all types of technical issues (Java script errors, network errors, etc.) that occur during user sessions. You can monitor and analyze user behaviors such as clicks and page movements to find causes of failures.
- **Finding the cause of customer attrition:** You can reduce the churn rate by tracing user attrition from a specific page or failures to complete specific tasks, analyze the causes, and improve the area.

User session analysis

You can check the information about the user session to analyze. You can view the end user's browser, OS, user agent, user ID, session ID, client IP, and other information.

1. Go to **Analysis > User Session Log Search**.
2. In the queried list, click the **Analysis** button in the **User session analysis** column.
3. The **User session analysis** window appears, where you can check the detailed information about the

user session log of the selected item.

User session flow

Based on the user session, you can check the performance data, events, and errors that occur when actual users interact with the website.

User session flow AI Analysis

Click the **AI Analysis** button at the top right of the **User session flow** list to have WhaTap AI read the event flow of the session in chronological order and report where the user got stuck and whether the event you are looking at is a cause or a consequence.

1.

In the **User session analysis** window, click the **AI Analysis** button at the top right of the **User session flow** list.

2.

The **AI Analysis** drawer opens on the right of the screen and the result is displayed as a stream.
- For a session with many events, only a subset is used for the analysis, prioritizing the events that need attention. AJAX requests that responded normally are excluded, and the same event repeated consecutively is merged into one.
 - Query string values in URLs, IP addresses, and user IDs are masked before being used for the analysis.
 - If you load more of the list and the analysis scope widens, the previous result is cleared and the analysis runs again.

If the button is disabled, hover over it to see the reason.

Message	Description
Loading data	The session log query is in progress. The button is enabled once the query finishes.
No log collected.	No event was collected for the session.

 For how AI Analysis behaves in common and the data scope it uses, refer to [the following document](#).

Checking the event information details

The types of events displayed in the **Title** column of the **User session flow** list are divided into the following. Select a user session event to see the information details about the event.

Page load(pageLoad)

Event that occurs when the website is loaded. You can check the detailed time and loaded resources for each event section. If you select the **User session analysis** tab, the related information is displayed consequently.

! For more information about the page load analysis, see [the following](#).

Router change(routeChange)

This event occurs when the route is changed on the SPA website. You can see the time taken to change the router.

AJAX load-ajax)

AJAX request-related event. You can check the detailed time information, AJAX request URLs, call pages, error messages, and such for each AJAX event section. If you select the **User session analysis** tab, the related information is displayed consequently.

! For more information about the AJAX event analysis, see [the following](#).

Click(click) / Rapid click(rapid_click)

You can see the selector that is the ID of the element clicked by the user, and the text information of the element. This helps you find which elements users clicked on which pages.

! To collect click events, it is required to set the option (`collectUserClick: true`) in the browser agent. For more information, see [the following](#).

Custom(custom)

You can check the information collected by custom events. It displays the information included in the title (`name`) and contents that are representative values of custom events.

! For more information about collection of custom events, see [the following](#).

Browser error(browser_error)

You can see source errors, network errors, and the like that occur from the website.

! For more information about error tracing, see [the following](#).

Event-based filtering

1. On the upper right of the **User session flow** list, select .
2. In the **Filter** list, select an event type.
3. Select **Apply**.

Session replay

✕ User session analysis

User session analysis
Session Replay

The screenshot shows a web application interface. At the top, there's a header with a logo and navigation links. Below it, a large light purple box contains a form with fields for email and password, and a 'Log In' button. Underneath the form are four colored boxes (orange, teal, green, blue) representing different user roles or plans. At the bottom, there are three smaller white boxes with red outlines.

User session information

Browser	Chrome
OS	macOS
Device	Desktop
User ID	x54j6cs3jo0ph0
Session ID	1480748341338840640
IP	15.165.167.245
User agent	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/124.0.0.0 Safari/537.36

00:0030:44

>

1x

2x

3x

4x

User session flow

Detail	Start time	Category	Title	Page URL	Elapsed time
	2024/06/19 18:29:08	#PerformanceEvent	AJAX load	http://test-client.jtest.ai.s3-website.i	40ms
	2024/06/19 18:29:08	#PerformanceEvent	AJAX load	http://test-client.jtest.ai.s3-website.i	58ms
	2024/06/19 18:29:08	#PerformanceEvent	AJAX load	http://test-client.jtest.ai.s3-website.i	38ms
	2024/06/19 18:29:08	#PerformanceEvent	AJAX load	http://test-client.jtest.ai.s3-website.i	39ms
	2024/06/19 18:29:08	#PerformanceEvent	AJAX load	http://test-client.jtest.ai.s3-website.i	48ms
	2024/06/19 18:29:08	#PerformanceEvent	AJAX load	http://test-client.jtest.ai.s3-website.i	56ms
	2024/06/19 18:29:08	#PerformanceEvent	AJAX load	http://test-client.jtest.ai.s3-website.i	40ms
	2024/06/19 18:29:10	#UserEvent	Click: Go to Cart Page (#basic-navbar-nav>Dl)	http://test-client.jtest.ai.s3-website.i	-

It provides the feature to record and replay all events that users perform on the website. The **Session replay** feature allows you to reproduce user behaviors such as clicking, scrolling, typing, and page switching. This helps you understand exactly how users actually interact with the website.

- For information about application of the **Session replay** feature by using agent options, see [the following](#).
- For more information about using the **Session replay** feature, see [the following](#).

Session Replay

Session replay is a feature that allows to record and replay all events performed by users on the website. This feature allows you to reproduce user behaviors such as clicking, scrolling, typing, and page switching. This helps you understand exactly how users actually interact with the website.

- **Record and replay user behaviors:** Session replay records user behaviors. Recorded user behaviors include mouse movements, clicks, scrolling, and more. You can replay recorded user behaviors to visually see what they were doing at a specific time.
- **Error tracing and failure reproduction:** You can analyze the cause of a failure by tracing the occurrence time of a specific event or error. You can visually check how users see when a page takes a long time to load or other technical glitches occur on the screen.

ⓘ The **Session replay** feature can be used by the members with the **View log** role. For more information about the member role scheme, see [the following](#).

Supported browsers

Data collected for session replay is collected from only the browsers that support the `requestIdleCallback` API. Check the supported versions for each browser.

Chrome	Edge	Safari	Firefox	Opera	Chrome Android	Safari iOS	Samsung Internet
> 47	> 79	> 8 *	> 6	> 55	> 47	> 9 *	> 5

ⓘ To avoid overloads during **session replay** data collection, data is collected in idle time of the main thread. For this purpose, use the `requestIdleCallback` API.

The `requestIdleCallback` API is designed to perform a callback when the browser is idle, making sure that it does not interfere with the main tasks while collecting the **session replay** data. For more information, see the [following link](#).

- ✓ • Safari and Safari iOS do not support the `requestIdleCallback` API, so session replay data is not collected by default. Setting the `sessionReplayCollectAllBrowser` option to `true` enables session replay data collection on these browsers. However, in this case, data is collected without checking main thread idle time, which may impact performance. For more details on applying this option, refer to [this link](#).

Collection methods

Instead of recording the actual screen or capturing it as an image, it collects DOM changes and CSS style information in text format. To avoid overhead, it only collects changes and transfers data when the browser's main thread is idle.

Applying the agent option

To apply the session replay feature, apply the browser agent option. See the following example:

```

1 <script>
2 (function (w, h, _a, t, a, b) {
3   w = w[a] = w[a] || {
4     config: {
5       projectAccessKey: {projec_access_key},
6       pcode: {pcode},
7       sampleRate: 100,
8       sessionReplaySampleRate: 50,
9       sessionReplayMaskAllTexts: false,
10      sessionReplayMaskAllInputs: false,
11      proxyBaseUrl: "https://rum-ap-northeast-2.whatap-browser-agent.io/",
12    },
13  };
14  a = h.createElement(_a);
15  a.async = 1;
16  a.src = t;
17  t = h.getElementsByTagName(_a)[0];
18  t.parentNode.insertBefore(a, t);
19 })(window, document, 'script', 'https://repo.whatap-browser-agent.io/rum/prod/v1/whatap-browser-agent.js',

```

```
'WhatapBrowserAgent', ");
20 </script>
```

- **sessionReplaySampleRate** Number

The percentage of session replay collection during the entire collection session can be set from 0 to 100. For example, if `sampleRate` is **100** and `sessionReplaySampleRate` is **50**, only 50% of all sessions is recorded for the session replay.

On the other hand, if `sampleRate` is **50** and `sessionReplaySampleRate` is **100**, 50% of all sessions is collected, and 100% of these is recorded for the session replay.

- **sessionReplayMaskAllTexts** Boolean

Default `true`

If you set the value to `false`, all text data is collected without masking.

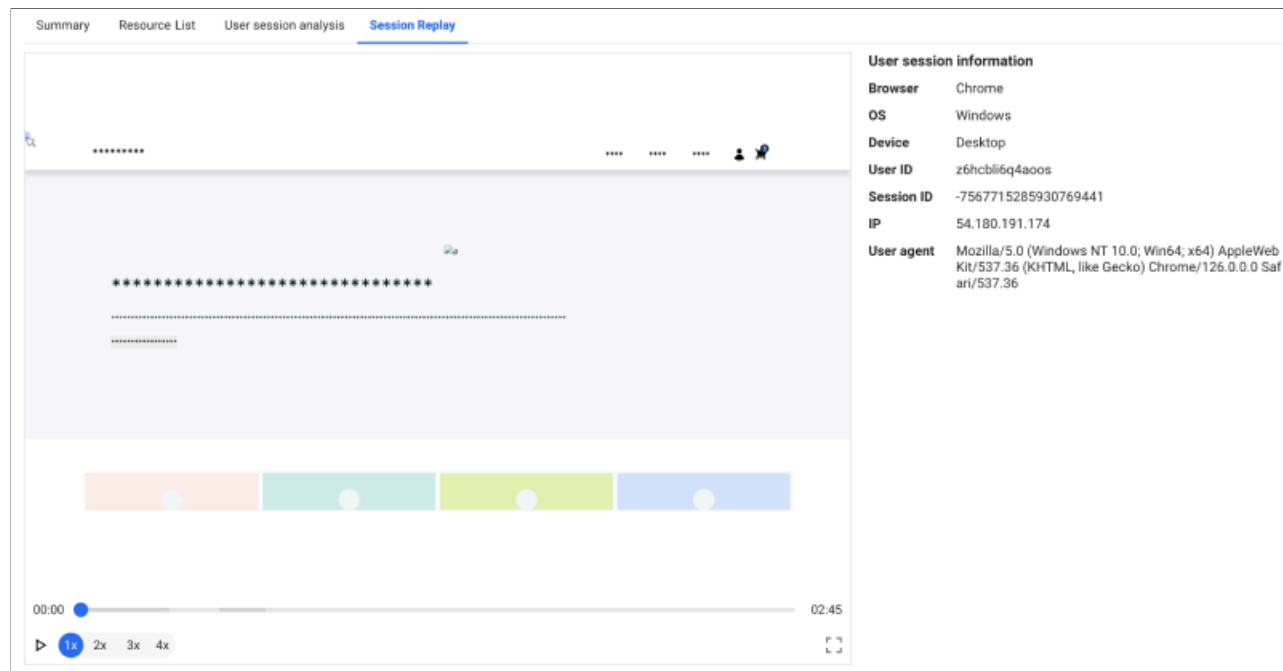
- **sessionReplayMaskAllInputs** Boolean

Default `true`

If you set the value to `false`, the data in all input fields is collected without masking.

Exclusion of sensitive data collection

The default method to exclude collection of sensitive data is provided. All text data on the screen is masked and collected as follows: However, the value of `sessionReplayMaskAllTexts` in the agent options must be set to `true`.



The screenshot displays the 'Session Replay' tab in a web analytics tool. The main area shows a web page with several redacted sections: a top navigation bar with asterisks, a large central content area with a long line of asterisks, and a bottom section with four colored boxes (orange, teal, green, blue) each containing a white circle. A timeline at the bottom indicates a duration from 00:00 to 02:45, with playback controls for 1x, 2x, 3x, and 4x speeds.

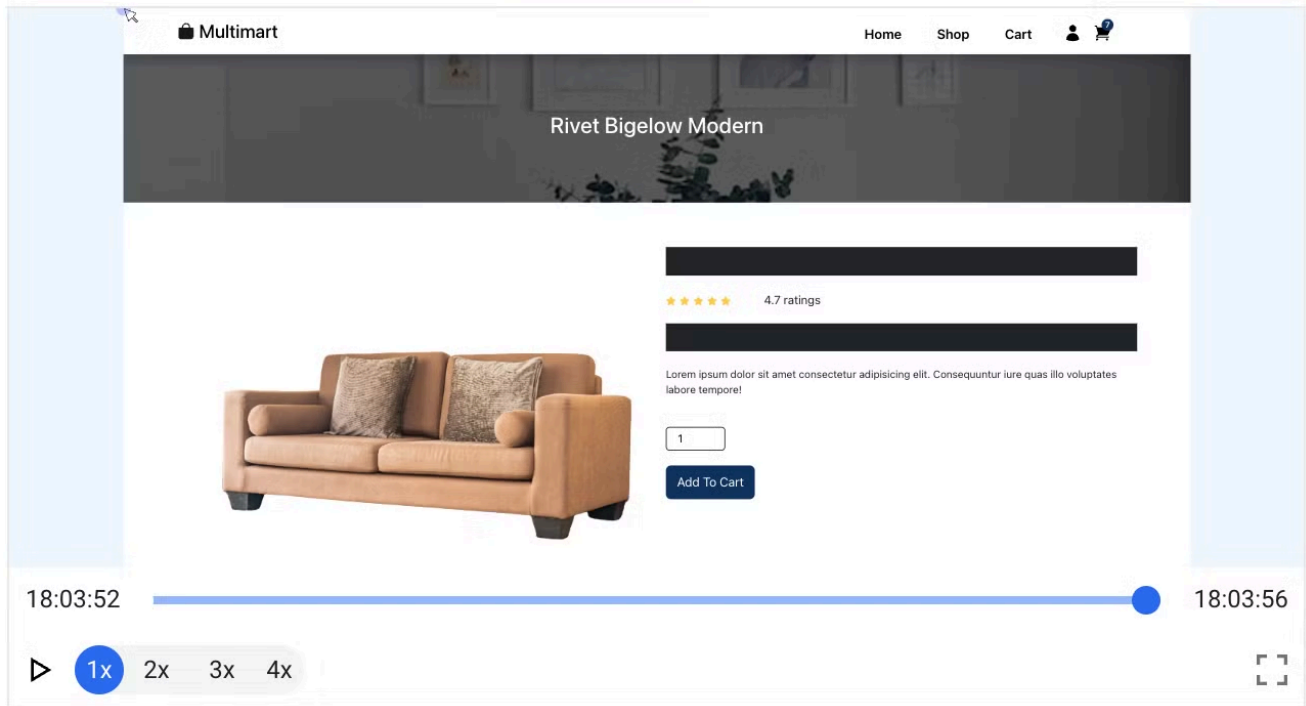
User session information

Browser	Chrome
OS	Windows
Device	Desktop
User ID	z6hcbli6q4aocs
Session ID	-7567715285930769441
IP	54.180.191.174
User agent	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/126.0.0.0 Safari/537.36

Masking specific areas

In addition to text, it may contain personally identifiable information or sensitive data, and the option to mask specific areas is provided for protection.

Masking for a specific area



Add `session-replay-block-class` to the class name of the html element for the area to be excluded from collection. See the following example:

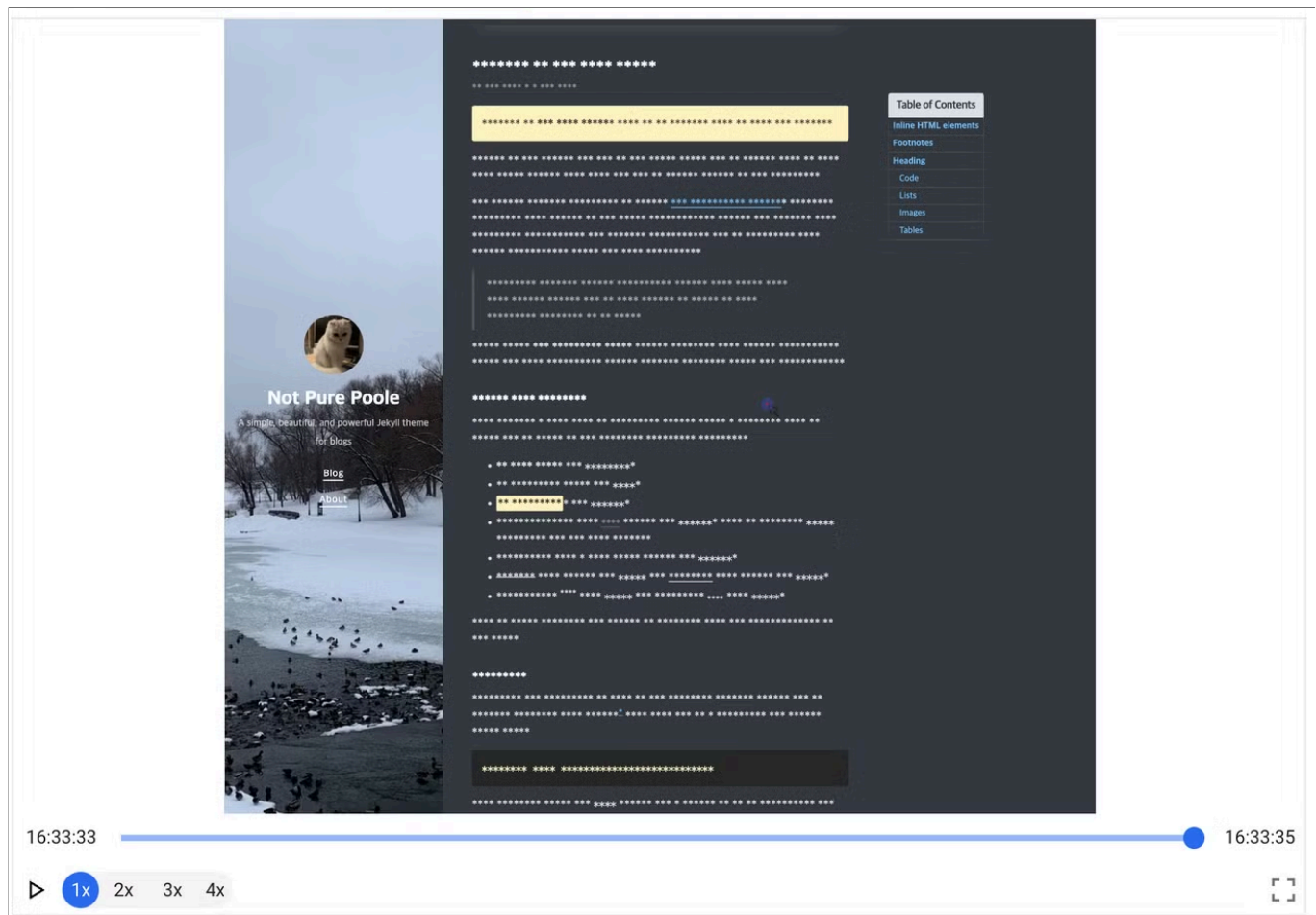
```

1 <div class="session-replay-block-class home-header pure-menu pure-menu-horizontal">
2   <div class="home-header-bar">
3     {% if site.data.archive %}
4     <ul class="home-header-menu pure-menu-list">
5       {% for item in site.data.archive %}
6         <li class="pure-menu-item">
7           <a href="{{ item.url | relative_url }}" class="pure-menu-link {% if page.url == item.url %}current-item{% endif
8             %}">{{ item.title }}</a>
9         </li>
10      {% endfor %}
11    </ul>
12  {% endif %}
13 </div>

```

Masking specific text

If you set `sessionReplayMaskAllTexts` to `false` among agent options, most of the text appears, but it provides the option to exclude specific text areas from collection.



To exclude a specific text area from collection, add `session-replay-mask-text-class` to the class name of the HTML element. See the following example:

```

1 <main class="session-replay-mask-text-class">
2   {{ content }}
3 </main>
4
5 <footer class="footer">
6   {%- assign year_from = site.posts[-1].date | date: '%Y' -%}
```

```

7  {%- assign year_to = site.time | date: '%Y' -%}
8  {%- if year_from == nil or year_from == year_to -%}
9    {%- assign year_string = year_from -%}
10  {%- else -%}
11    {%- assign year_string = year_from | append: '&nbsp;' | append: '-' | append: '&nbsp;' | append: year_to -%}
12  {%- endif -%}
13  </footer>

```

Using the session replay

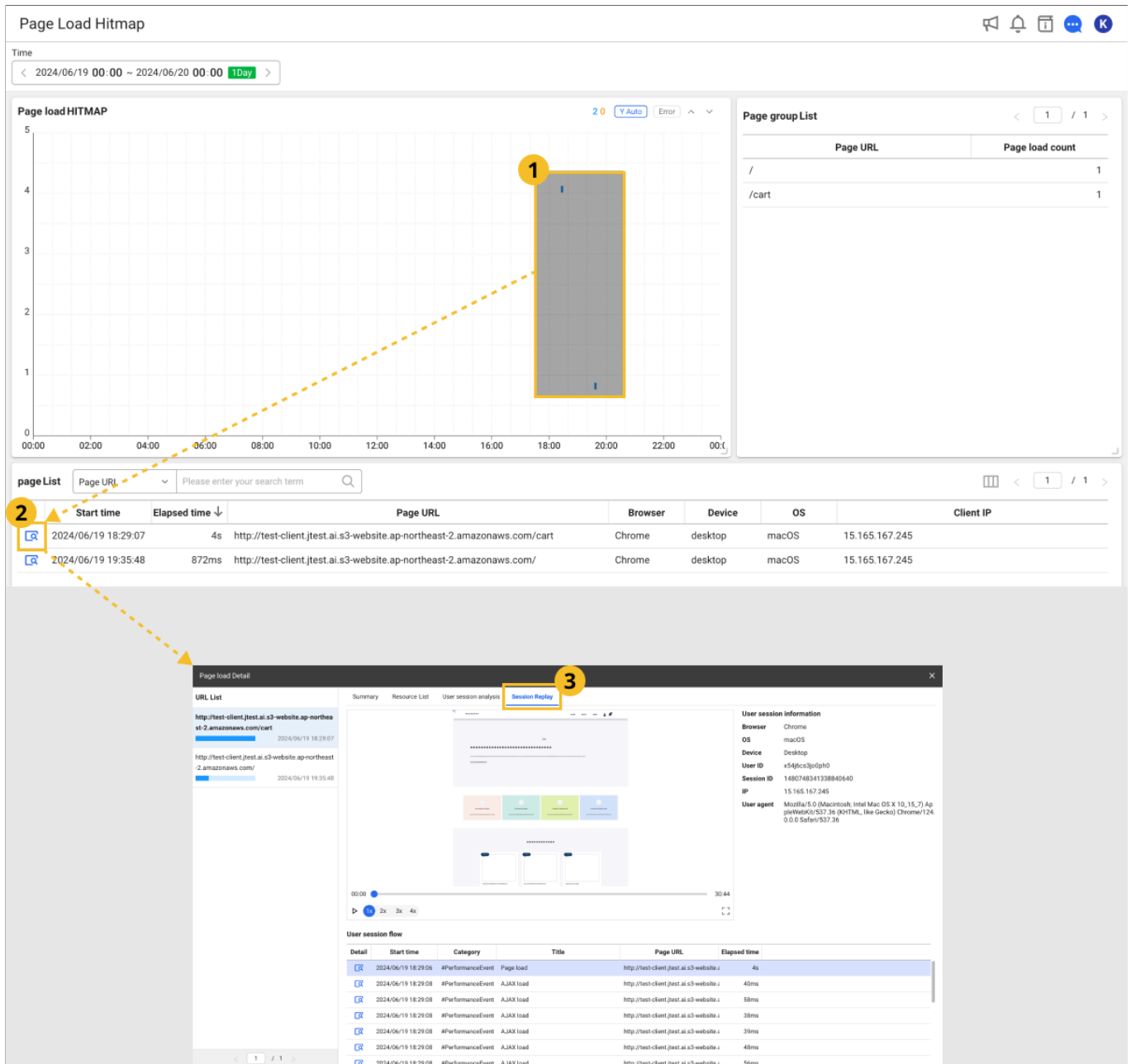
The **Session replay** feature can be used in the following menu path:

- **Analysis > Page Load Hitmap**
- **Analysis > AJAX Hitmap**
- **Analysis > User Session Log Search**

ⓘ The **Session replay** feature can be used by the members with the **View log** role. For more information about the member role scheme, see [the following](#).

Page Load Hitmap

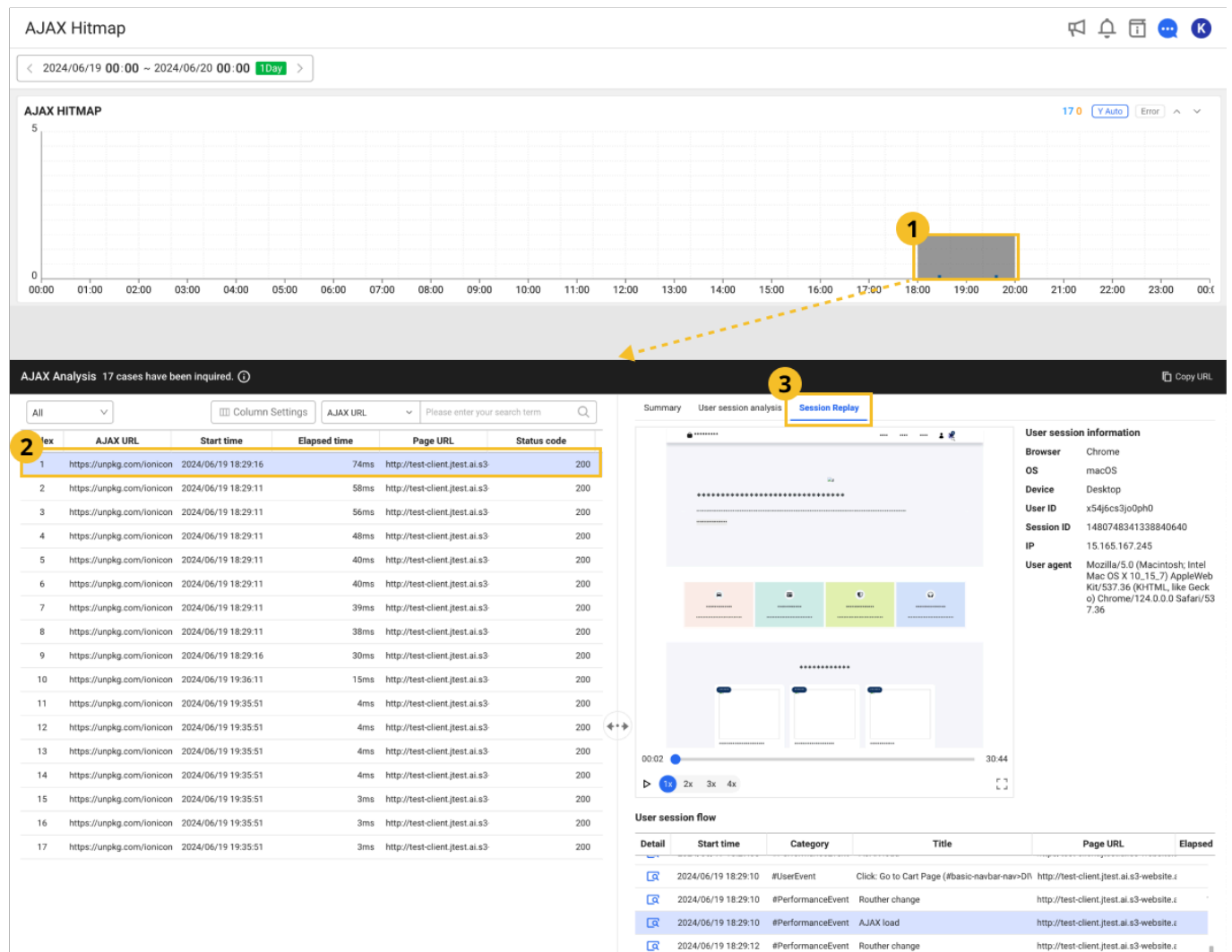
Analysis > Page Load Hitmap



1. In the **Page load Hitmap** chart, drag a specific area.
2. In **page List**, select  to check the Session replay feature.
3. If the **Page load Detail** window appears, select the **Session replay** tab.

AJAX Hitmap

Analysis > AJAX Hitmap



1. In the **AJAX Hitmap** chart, drag a specific area.
2. If the **AJAX Analysis** window appears, select an item from the list to check the Session replay feature.
3. On the screen to the right, select the **Session replay** tab.

User Session Log Search

Analysis > User Session Log Search

User Session Log Search

Time

< 2024/06/19 00:00 ~ 2024/06/20 00:00 1Day >

Filter

Enter Filter and press enter to add.

2

Search

#PerformanceEvent (26)

#PerformanceEvent IP browser

browserVersion browseragenttime.n

device os osVersion pageUri

pageUriPath screenHeight.n

screenWidth.n sessionID type

Of the total 0 results, 37+ match your search.

Oldest Newest

Please input the Keyword.

Search

Share

Settings

Timestamp

Log

2024-06-19 18:29:14.312 os macOS screenHeight.n 1294 pcode 39129 IP 15.165.167.245

{"total":40,"txid":"-1115284025731083035","mtid":"0","status":200}

2024-06-19 18:29:14.312 os macOS screenHeight.n 1294 pcode 39129 IP 15.165.167.245

{"total":58,"txid":"-5361159645338114134","mtid":"0","status":200}

User session analysis

Analysis

User session analysis

Session Replay

00:00

30:44

1x 2x 3x 4x

User session information

Browser

Chrome

OS

macOS

Device

Desktop

User ID

x54j6cs3jo0ph0

Session ID

1480748341338840640

IP

15.165.167.245

User agent

Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/124.0.0.0 Safari/537.36

User session flow

Detail	Start time	Category	Title	Page URL	Elapsed time
	2024/06/19 18:29:08	#PerformanceEvent	AJAX load	http://test-client.jtest.ai.s3-website.i	40ms
	2024/06/19 18:29:08	#PerformanceEvent	AJAX load	http://test-client.jtest.ai.s3-website.i	58ms
	2024/06/19 18:29:08	#PerformanceEvent	AJAX load	http://test-client.jtest.ai.s3-website.i	38ms
	2024/06/19 18:29:08	#PerformanceEvent	AJAX load	http://test-client.jtest.ai.s3-website.i	39ms
	2024/06/19 18:29:08	#PerformanceEvent	AJAX load	http://test-client.jtest.ai.s3-website.i	48ms
	2024/06/19 18:29:08	#PerformanceEvent	AJAX load	http://test-client.jtest.ai.s3-website.i	56ms
	2024/06/19 18:29:08	#PerformanceEvent	AJAX load	http://test-client.jtest.ai.s3-website.i	40ms
	2024/06/19 18:29:10	#UserEvent	Click: Go to Cart Page (#basic-navbar-nav>DI\	http://test-client.jtest.ai.s3-website.i	-

1. Set the time to search in **Time**.

2. On the right of the screen, select

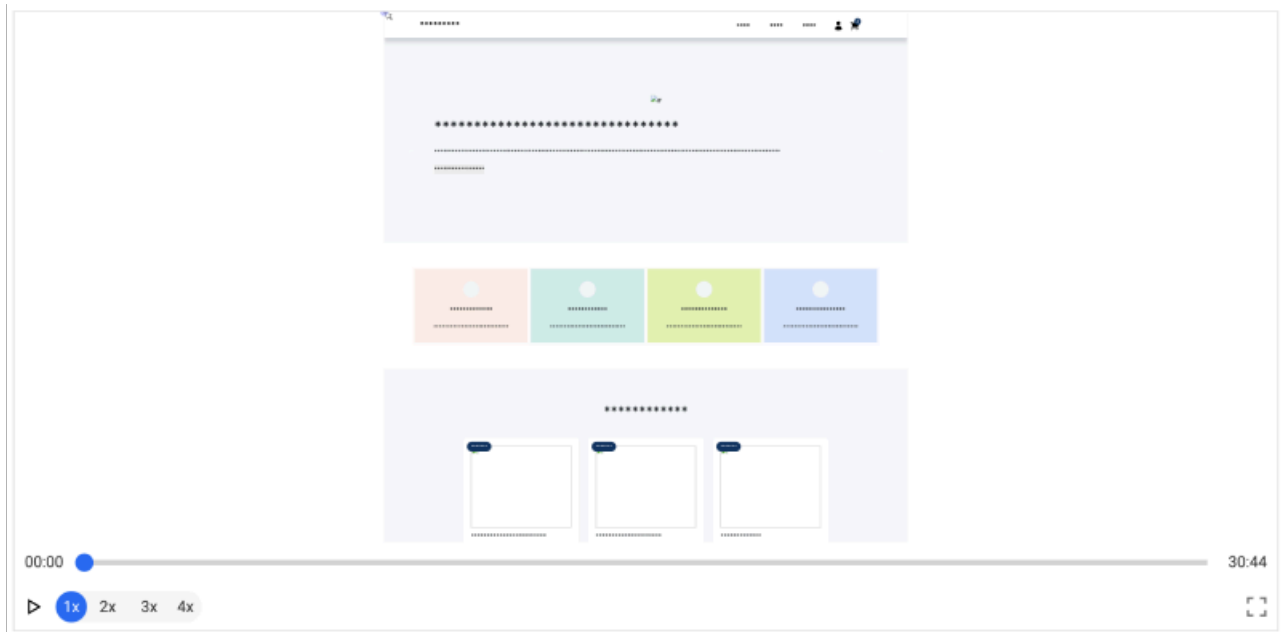
3. In the list of search results, select **Analysis** on the utmost right of the item to check the Session replay feature.

4. If the **User session analysis** window appears, select the **Session replay** tab.

Using the session replay

Session replay player

You can replay the recorded user behavior events to see how actual users interact with the screen.



- : You can play or pause recorded user behavior events.
- **1x / 2x / 3x / 4x**: You can adjust the playback speed.
- : The player screen can be enlarged to full screen. To end the full screen, press the **ESC** key.
- You can drag the timeline to move it to a desired location. The areas without mouse interaction are displayed in light gray.



User session information

On the right of the **Session replay** screen, you can see the actual user's session information.

User session information	
Browser	Chrome
OS	macOS
Device	Desktop
User ID	x54j6cs3jo0ph0
Session ID	1480748341338840640
IP	15.165.167.245
User agent	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/124.0.0.0 Safari/537.36

- **Browser:** Type of the browser of the end user.
- **OS:** OS type of the device used by the end user
- **Device:** Type of the device of the end user.
- **User ID:** Unique ID issued when an end user first accesses the web application
- **Session ID:** A unique ID issued when an end user of the web application begins browsing a web page.
- **IP:** IP address of the device used by the end user
- **User agent:** The user agent of the browser when an event occurred.

User session flow

You can check the event records collected based on the start time. It is the same as the **User session flow** information shown in **User session analysis**.

[illegible]

- If you select an individual event, the timeline of the Session Replay player is moved to the start point of the event.
- To check the detailed information about an individual event, select .

! For more information about **User session analysis**, see [the following](#).

AI Analysis

On the **AI Analysis** tab of the analysis panel, WhaTap AI reads the event flow of the whole session and tells you where the user got stuck. The result appears inside the panel instead of a drawer, so it does not cover the player.

1. Select the **AI Analysis** tab on the analysis panel.
 2. Click the **AI Analysis** button in the middle of the tab.
 3. The result is displayed as a stream.
- Timestamps in the result are shown as links. Click a link to move the player timeline to that point.

- Session logs are loaded progressively as playback advances, so the analysis runs against the logs loaded up to the moment you clicked the button.
- Query string values in URLs, IP addresses, and user IDs are masked before being used for the analysis.

If the following message appears instead of the button, check the stated reason.

Message	Description
Loading data	The session logs are still loading. The button appears once loading finishes.
No log collected.	No event was collected for the session.

ⓘ For how AI Analysis behaves in common and the data scope it uses, refer to [the following document](#).

Metrics

Home > Select Project > [Analysis](#) > [Metrics Chart](#)

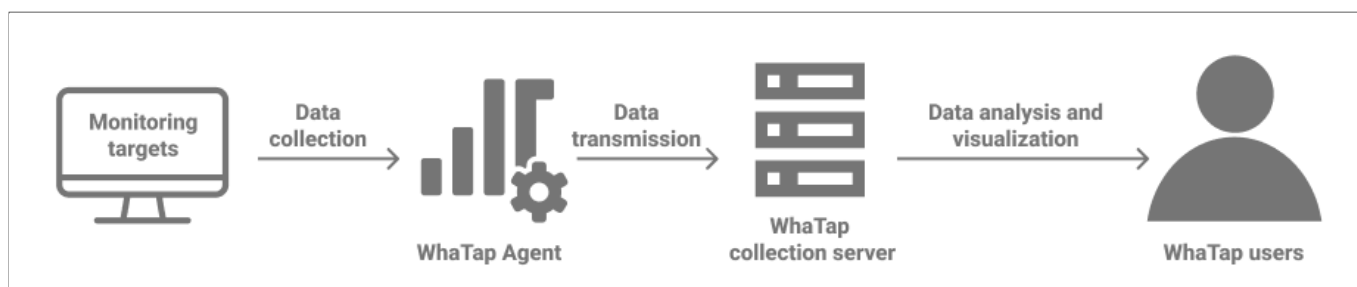
What is a metric?

A **metric** is time-series data that represents the state of a monitoring target as numerical values. It is the result of processing performance data collected by the WhaTap agent from monitoring targets.

Through metrics, you can view key performance indicators such as memory usage per server and DB connection time at a glance. After identifying a problem, you can perform detailed analysis through logs and traces.

Resource usage statistics help you estimate the required amount of resources, which can also be used for cost optimization.

How WhaTap collects metrics



To collect metrics, install the WhaTap agent on the monitoring target. Metrics sent by the agent are stored on the WhaTap collection server.

Metric components

WhaTap **metrics** consist of the following components:

- **Category:** A name that groups related metrics.

- **Tags:** Labels that identify collection targets. Stores rarely-changing unique information such as IP, Oname, and Host. A metric can contain multiple tags.
- **Fields:** Stores actual measured values. A metric can contain multiple fields.
- **Time:** The time when the metric was collected.
- **Oid:** The unique number of the agent that collected the metric.
- **Oname:** The name of the agent that collected the metric.

Viewing and visualizing metrics

WhaTap provides menus to view collected metrics in various ways.

- **Metrics Search:** View the original metric data.
- **Metrics Chart:** View metric data through visualized charts.

Browser Metrics

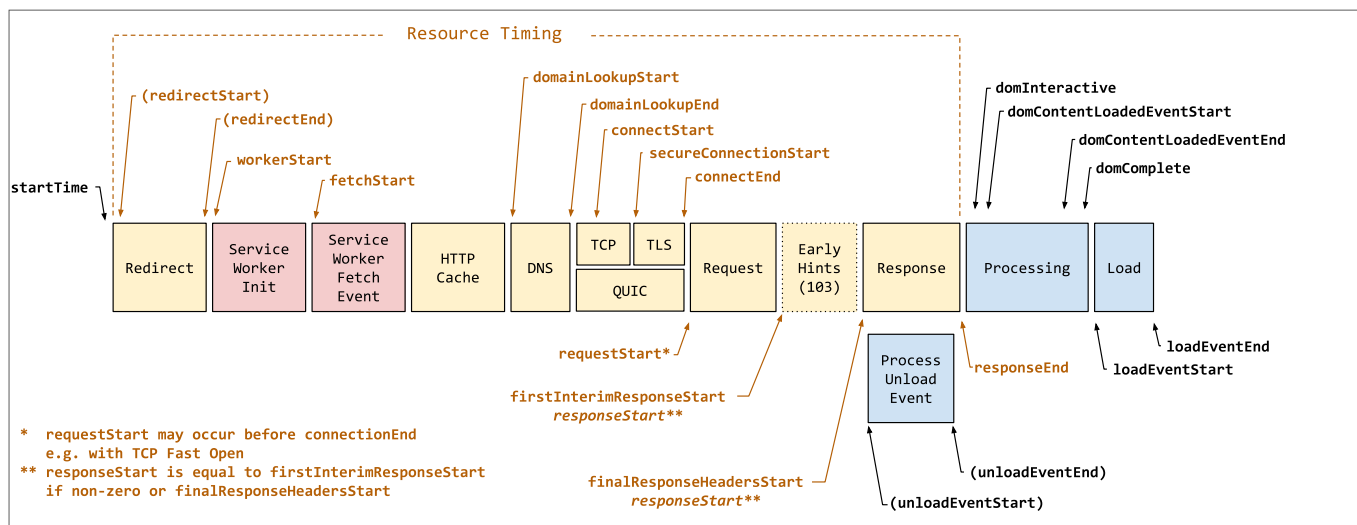
Home > Select Project > **Analysis** > **Metrics Search**

Metrics available in the **Metrics Search** menu. Each metric contains the following information. Some metrics may not have tag information.

- **Tags:** Information to identify the collection target (agent name, IP, etc.)
- **Fields:** Actual measured values (CPU usage, response time, etc.)

rum_page_load_each_page

- Collects page load time metrics (references **Navigation Timing** from the **Performance API**)
- Page load response time covers up to the **loadEventEnd** point
- Collection interval: 5 seconds
- Statistical data: 5 minutes, 1 hour



Tags

Tag	Description	Remarks
page_group	Page Group	-

Fields

Field	Unit	Description	Remarks
page_load_backend_time	millisecond	Average processing time in the backend area during page load	Time elapsed from startTime to responseEnd
page_load_frontend_time	millisecond	Average processing time in the frontend area during page load	Time elapsed from domInteractive to loadEventEnd
page_load_redirect_time	millisecond	Average amount of redirection time spent by the browser when loading web pages	Time elapsed from redirectStart to redirectEnd
page_load_cache_time	millisecond	Average time spent by a browser to search cached resources while loading web pages	Time elapsed from fetchStart to domainLookupStart
page_load_dns_time	millisecond	Average time spent by a browser to look up website domains while loading web pages	Time elapsed from domainLookupStart to domainLookupEnd
page_load_connect_time	millisecond	Average amount of network connection time	Time elapsed from connectStart to connectEnd

Field	Unit	Description	Remarks
		to the server spent by the browser	
<code>page_load_ssl_time</code>	millisecond	Average TCP handshake time spent by a browser when loading web pages	Time elapsed from secureConnectionStart to connectEnd
<code>page_load_firstbyte_time</code>	millisecond	Average time spent from the browser sending network requests while fetching web pages to the first byte receptions from the server	Time elapsed from requestStart to responseStart
<code>page_load_download_time</code>	millisecond	Average time spent for a browser to download resources from the server while loading web pages	Time elapsed from responseStart to responseEnd
<code>page_load_domcontentloaded_time</code>	millisecond	Average time for a web page to render its initial content and become interactive	Time elapsed from domContentLoadedEventStart to domContentLoadedEventEnd
<code>page_load_domload_time</code>	millisecond	Average time for a web page to load all resources, finish rendering, and become interactive	Time elapsed from loadEventStart to loadEventEnd
<code>page_load_render_time</code>	millisecond	Average time spent rendering resources downloaded from the server to the screen and completing page load	Time elapsed from domInteractive to domComplete

Field	Unit	Description	Remarks
		events	
page_load_event_time	millisecond	Average time for the load event to complete as measured by the browser	Time elapsed from the start to loadEventEnd
page_load_duration	millisecond	Average time taken by the browser to fully load a web page	Time to complete additional interactions after loadEventEnd , page load completion time measured by the browser agent
page_load_count	Count	Page Load Count	-

rum_page_load_browser_each_page

- Collects page load metrics by browser
- Collection interval: 5 seconds
- Statistical data: 5 minutes, 1 hour

Tags

Tag	Description	Remarks
page_group	Page Group	-
browser_name	Browser name	-

Fields

Field	Unit	Description	Remarks
page_load_count	Count	Page Load Count	-
page_load_duration	millisecond	Average time taken by the browser to fully load a web page	-

rum_page_load_device_each_page

- Collects page load metrics by device
- Collection interval: 5 seconds
- Statistical data: 5 minutes, 1 hour

Tags

Tag	Description	Remarks
page_group	Page Group	-
device_name	Device name	-

Fields

Field	Unit	Description	Remarks
page_load_count	Count	Page Load Count	-
page_load_duration	millisecond	Average time taken by the browser to fully load a web page	-

rum_page_load_os_each_page

- Collects page load metrics by OS
- Collection interval: 5 seconds
- Statistical data: 5 minutes, 1 hour

Tags

Tag	Description	Remarks
page_group	Page Group	-
device_name	Device name	-

Fields

Field	Unit	Description	Remarks
page_load_count	Count	Page Load Count	-
page_load_duration	millisecond	Average time taken by the browser to fully load a web page	-

rum_router_change_each_page

- Collects metrics related to page transition events in single-page applications (SPA)
- Collection interval: 5 seconds
- Statistical data: 5 minutes, 1 hour

Tags

Tag	Description	Remarks
page_group	Page Group	-
completed_loading	Whether a page change completes or not	Page change is not complete and appears as <code>false</code> when switching to another page

Fields

Field	Unit	Description	Remarks
router_change_count	Count	SPA page change count	-
router_change_time	millisecond	Average time taken to fully change pages	-

rum_resource_each_page

- Collects metrics related to resource response times from web pages
- Collection interval: 5 seconds
- Statistical data: 5 minutes, 1 hour

Tags

Tag	Description	Remarks
page_group	Page Group	-
is3rdParty	Whether the resource is called	If the domain of the web page is different from the domain of

Tag	Description	Remarks
	from a third party	the called resource, <code>true</code> is passed.
<code>request_host</code>	URL host of the resource	-
<code>request_path</code>	URL path of the resource	-
<code>type</code>	Resource type	css, image, script, etc.

Fields

Field	Unit	Description	Remarks
<code>resource_dns_time</code>	millisecond	Average time spent by a browser to look up website domains while loading web pages	Time elapsed from domainLookupStart to domainLookupEnd
<code>resource_connect_time</code>	millisecond	Average amount of network connection time to the server spent by the browser	Time elapsed from connectStart to connectEnd
<code>resource_ttfb_time</code>	millisecond	Average time spent from the browser sending network requests while fetching web pages to the first byte receptions from the server	Time elapsed from requestStart to responseStart
<code>resource_download_time</code>	millisecond	Average time spent for a browser to download resources from the server while loading web pages	Time elapsed from responseStart to responseEnd
<code>resource_duration</code>	millisecond	Average time taken by the browser to fully load a web page	Time to complete additional interactions after loadEventEnd , page load completion time measured

Field	Unit	Description	Remarks
			by the browser agent
page_load_count	Count	Page Load Count	-
resource_size	Byte	Resource size	-

rum_ajax_each_page

- Collects AJAX response time and status code metrics
- Collection interval: 5 seconds
- Statistical data: 5 minutes, 1 hour

Tags

Tag	Description	Remarks
page_group	Page Group	-
request_host	Host of the AJAX request URL	-
request_path	Path of the AJAX request URL	-

Fields

Field	Unit	Description	Remarks
ajax_1xx_count	Count	Number of AJAX cases with the response status code between 100 and 200	-

Field	Unit	Description	Remarks
<code>ajax_2xx_count</code>	Count	Number of AJAX cases with the response status code between 200 and 300	-
<code>ajax_3xx_count</code>	Count	Number of AJAX cases with the response status code between 300 and 400	-
<code>ajax_4xx_count</code>	Count	Number of AJAX cases with the response status code between 400 and 500	-
<code>ajax_5xx_count</code>	Count	Number of AJAX cases with the response status code of 500 or higher	-
<code>ajax_count</code>	Count	AJAX Count	-
<code>ajax_duration</code>	millisecond	Average response time for AJAX requests	-
<code>ajax_fail_count</code>	Count	Number of failed AJAX requests	-

rum_error_total_each_page

- Collects browser error metrics
- Collection interval: 5 seconds
- Statistical data: 5 minutes, 1 hour

Tags

Tag	Description	Remarks
page_group	Page Group	-
browser	Browser name	-
device	Device name	-
os	OS name	-
error_message	Error Message	-
error_type	Error Type	-

Fields

Field	Unit	Description	Remarks
error_count	Count	Error Count	-

rum_web_vitals_each_page

- Collects Core Web Vitals metrics
- Collection interval: 5 seconds
- Statistical data: 5 minutes, 1 hour

Tags

Tag	Description
page_group	Page Group

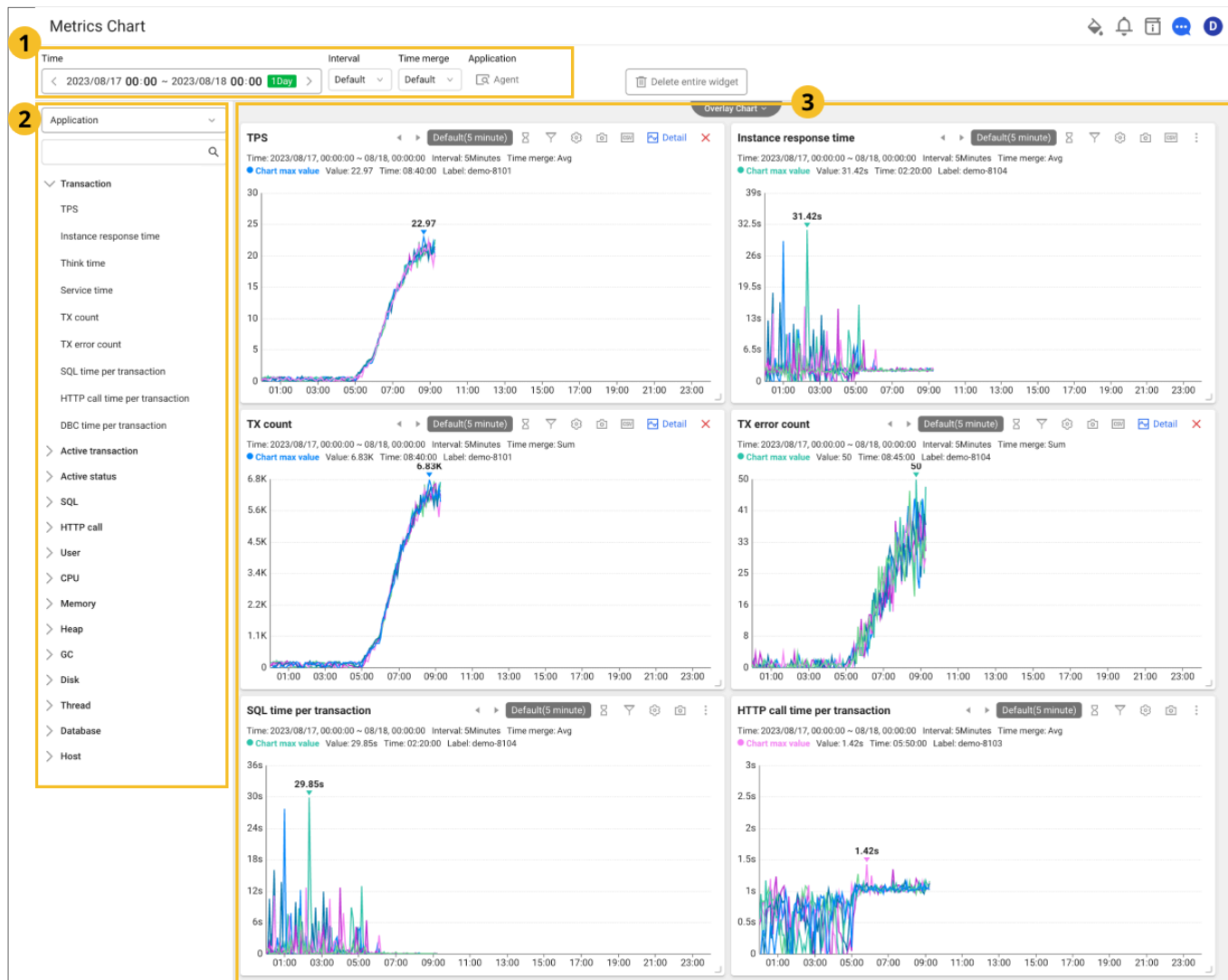
Fields

Field	Unit	Description
web_vitals_cls	Score	Score that indicates how much the layout shifts during page load (Cumulative Layout Shift)
web_vitals_fid	millisecond	Average time for the browser to respond after the user's first input (First Input Delay)
web_vitals_lcp	millisecond	Average time for the largest content element on the website to load (Largest Contentful Paint)
web_vitals_inp	millisecond	Average time from user interaction to the next screen paint (Interaction to Next Paint)

Metrics Chart

Home > Select Project > **Analysis** > **Metrics Chart**

In **Metrics Chart**, you can view the metrics data collected from the monitoring targets in a chart as follows. Specifying the time and metrics is required.



Options at the top

In the **1** area, you can specify the time range of the chart and the agent to be monitored through the options at the top of **Metrics Chart**.

- **Time:** You can specify the start and end of the X-axis as the total range in time.

① For more information on how to use the **Time** option.

- **Interval:** You can specify the X-axis data interval as the time interval.
- **Time merge:** As one of the data merging methods, you can merge data within the time specified by intervals.

For example, **Avg** indicates the average value of data within 1 hour.

- **Agent:** You can specify the agents to query. If not specified, all items are queried.

Metrics list

The **2** area is a list of metrics to query for options. First select a **Category**. After querying the metrics under the selected **Category**, select a desired metric. If you select the **Category** and metric, Based on the data within the time range specified in the top menu of the area **1**, you can see the chart widgets in the **3** area.

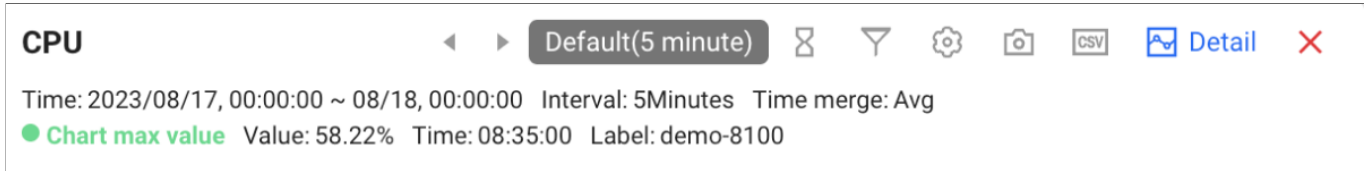
① Data Merge

Data merge provides the following methods: **Object Merge** and **Time merge**.

- **Time merge** merges data with the same field values at regular intervals in the original data.
- **Object Merge** merges data with matching tags among data with different field values.

Chart widget

On the top left of **3**, you can see the metric name. On the top right of the chart widget, you can see the following options:



- Time movement: ◀ ▶ Using the left arrow button or right arrow button, it can be moved by -1 or +1 as much as the selected time range.

For example, if the time range is between 00:00 on February 13 and 00:00 on February 14, when you select the ◀ left arrow button, you can see the data within the time range between 00:00 on February 12 and 00:00 on February 13.

- Interval/Time Merge: In the menu at the top of ❶, you can modify the specified interval and time merge.
- By selecting the monitoring target: ⌵, you can specify the monitoring targets. If not selected, it searches for all.
- Time comparison: If you select ⌲, you can compare the current trend of the same metric with that of the previous time zone.
- By selecting the snapshot: 📷 icon, you can snapshot charts except for widget options.
- By selecting the CSV: CSV icon, you can download the plot data on the chart as a CSV file.
- **Detail:** By selecting the 📊 icon, the detailed view can be seen. When there are many monitoring targets, you can see the metrics trends for each monitoring target individually.

❗ If you do not see any options at the top of the metrics chart widget, select the ⋮ icon.

Deleting a widget

To delete all widgets placed on the screen, select 🗑 **Delete All Widgets** on the screen.

Metrics Search

Home > Select Project > **Laboratory** > **Metrics Search**

You can query specific metrics based on tags in **Metrics Search**.

1

Metrics Search

Time selector

< 2023/08/17 07:57 ~ 2023/08/17 08:57 60Min >

Max count 100

Category app_counter

2

Tags All

container containerKey host_ip okindName oname onodeName pid type 8/8

Filter Filter by tag value

Field All

_clock_delta active_tx_0 active_tx_3 active_tx_8 active_tx_count apdex_satisfied apdex_tolerated apdex_total arrival_rate httpc_count httpc_error httpc_time meterlog resp90 resp95 resp_time sql_count

sql_error sql_fetch_count sql_fetch_time sql_time tps tx_count tx_doc_time tx_error tx_http_time tx_sql_time tx_time tx_time_sql_sum 5/29

3

	Time	Oid	Tags								Fields				
			container	containerKey	host_ip	okindName	oname	onodeName	pid	type	active_tx_3	arrival_rate	sql_time	tps	tx_error
1	2023-08-17 07:57:40	633280970	agent.service	1395943798	10.21.1.26	demo-okind-1	demo-8103	node-1	3285013	java	10	20.57942	7.822286	20.94554	0
2	2023-08-17 07:57:40	-857948929	agent.service	1395943798	10.21.1.26	demo-okind-1	demo-8105	node-1	3284721	java	10	17.974834	10.626543	20.609276	0

1 Time and category selection

In the **1** area, you can specify the **Time selector**, **Max count**, and **Category** for the collected metrics. **Time selector** and **Category** must have been specified.

- **Time selector:** You can query by specifying the time when metrics were collected. The default value is **1 hour**. In addition to the lookup time provided as the default option, you can specify the date and time after selecting the **Time selector** tab.

! For more information on how to use the **Time selector** option.

- **Max count:** You can specify the maximum number of metrics to search in the **3** list. You can set the count up to **10**, **50**, **100**, **200**, **300**, **1000**, **2000**, and **3000**.
- **Category:** It is a classification unit for the related metrics. You can set the desired **Category** after selecting the **Category** tab.
- **Refresh categories, tags, and field options:** **Category**, **Tags**, and **Field** options can be fetched.

2 Tag and field selection

In the 2 area, select **Tags** and **Field**. The default setting is "Select All," unless specified separately by user.

- **Tags:** Unique data that can identify the collected targets.
- **Field:** Metrics collected from the monitored targets.
- **Filter:** Select  **Filter by tag value** and then set the tag value for filtering.

For example, you can search filtered data by setting the value of `oname` to `demo-8101`.

-  **Search:** If you set the conditions and then select **Search**, you can search the source data of the metric in the 3 area.
-  : You can download the source data of the metric as a CSV file.

3 Metrics table

Because the metrics to be collected cannot be specified in advance, it is important to check the source data for all metrics being collected. After setting the desired conditions in the condition area above, you can search the original data of the metric in a table format in the 3 area. Columns in the table are changed when you specify the conditions for **Tags** and **Field**.

- To search metrics, **Time selector** and **Category** must have been specified.
- To search metrics, **Tags** and **Field** are optional.

Page Load Statistics

Home > Select Project > **Statistics** > **Page Load Statistics**

Page Load Statistics provides load performance information for each page of the website. You can trace the load process of each page to identify underperforming pages and analyze the results of improved pages. You can also gain insights needed to improve the user experience of web applications.

- It provides load performance statistics for each page URL path.
- You can check statistics based on average, maximum, minimum, and percentile values.
- You can perform in-depth performance analysis by filtering the time information in the page load process.
- It uses the Navigation Timing API to collect the time for each step of a page load.

! For more information about the data collected in **Page Load Statistics**, see [the following](#).

Checking the page load statistics

You can check various performance metrics such as **page load time**, **DOM load time**, **network latency**, and **resource load time** based on the URL path of the page. Using these metrics, you can evaluate the performance of each page and determine whether optimization is needed.

1. Go to **Statistics** > **Page Load Statistics** menu.
2. In **Time**, set the date and time to query the performance metrics.
 - **Configurable time range:** Minimum 5 minutes, maximum 3 days
 - **Data query unit:** 2 hours or less → 5-minute intervals, more than 2 hours → 1-hour intervals

✓ Recent search time

Previously queried time ranges are automatically saved in the recent search time list. Up to 10 search histories are saved, and you can quickly re-query by clicking the desired item.

3. In **Statistical criteria**, select the desired option. See the [Statistical criteria](#) document.
4. Click the  button on the upper right of the screen.
5. The performance metrics for the selected time range are displayed as charts and lists in each section.
 - If you hover the mouse over the graph, you can check the detailed metrics for that time zone.

Selecting the statistical criteria

Statistical criteria is the aggregation method used when analyzing performance metrics. Users can select the desired criteria as an option.

Statistical criteria	Description	
Avg	Provides statistics based on the average load time for each page. However, since intermittent performance degradation may be difficult to detect, you can use the maximum and percentile options together to check when performance degradation occurs and its severity.	
Max	Provides statistics based on the maximum load time for each page. That is, you can identify which page URL has the most inefficient page load time.	
Min	Provides statistics based on the minimum load time for each page. This can be used as a reference when evaluating the optimization status based on the fastest load time or setting performance goals. It is more effective to analyze together with the average or percentile option rather than using it alone.	
percentile	Allows analysis based on the load time of a subset of all users, making it useful for identifying the proportion of users experiencing performance degradation and improving user experience.	
	95quantile	The slowest 5% of page load times, representing the experience of the top 5% of users with the slowest page load times.
	90quantile	The page load time for the slowest 10% of users, primarily useful for identifying

Statistical criteria	Description	
		performance issues among slow user groups.
	75quantile	Shows the load time of the top 25% of users whose page load time is slower than the bottom 75%.
	50quantile	The median, meaning half of all users loaded the page faster than this time and half loaded it slower.

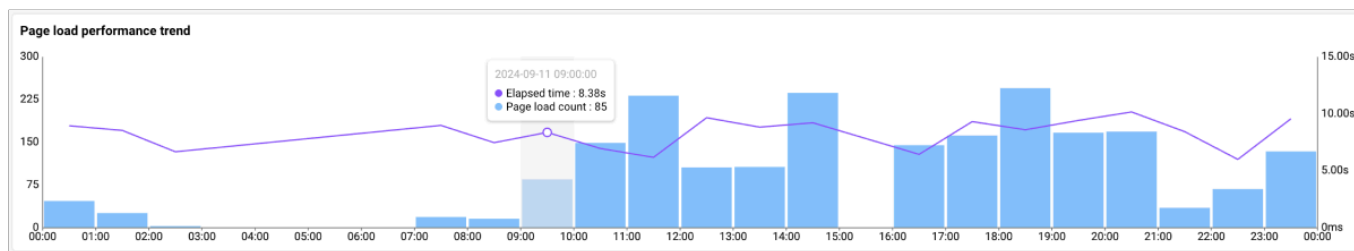
Page load performance trend

This chart visualizes the trend for page load counts and performance metrics (response time) over time. This graph allows you to check whether there were any performance fluctuations during a specific time zone. The items displayed in the chart vary depending on **Statistical criteria**.

- The **blue bar graph** shows the number of page loads for that time zone. If you click the blue bar graph for a specific time, you can check the data in the **Top 10 page load times**, **Top 10 page load counts**, and **Page group list** sections for that time range.

Statistical criteria: Avg/Max/Min

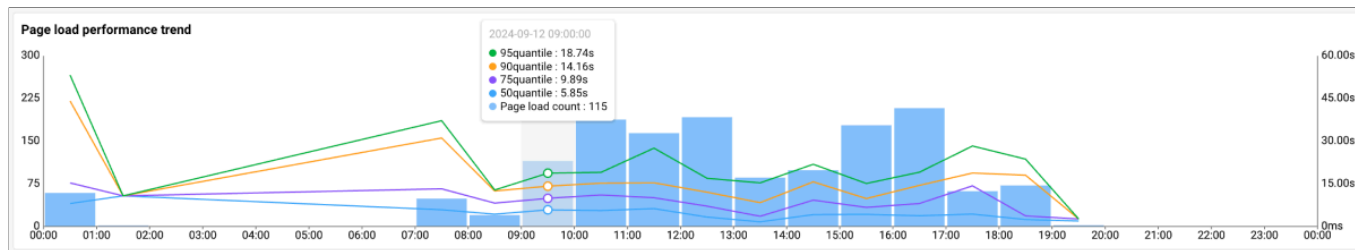
The purple line graph is the average page load time for that time zone.



Statistical criteria: percentile

Blue represents **50quantile**, purple represents **75quantile**, orange represents **90quantile**, and green represents

95quantile.



Top 10 page load counts - Elapsed time

Page groups are grouped at 5-minute or 1-hour intervals and the **elapsed time** of the top 10 page groups with the highest page load count per unit is displayed in a time series graph. The graph is sorted in descending order of page load time.

- The bar chart at the bottom of the section displays the **average value per page group** of the data represented in the time series graph.
- If data is not collected in continuous time intervals but exists only in specific intervals, only the bar graph is displayed.
- **How elapsed time is displayed based on the statistical criteria**
 - **Avg, Max, Min:** Displays elapsed time using the selected **reference value**.
 - **percentile:** Displays elapsed time using the **percentile value** selected in the upper right of the section.

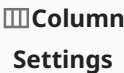
Top 10 page load counts - Count

Page groups are grouped at 5-minute or 1-hour intervals and the top 10 page groups with the highest page load count per unit are displayed in a time series graph. The graph is sorted in descending order of page load count.

- The bar chart at the bottom of the section displays the **sum value per page group** of the data represented in the time series graph.
- If data is not collected in continuous time intervals but exists only in specific intervals, only the bar graph is displayed.

Page group list

This is a list where you can check various performance metrics based on the page group. If you hover the mouse over a table column header, detailed information about that column is provided as a tooltip. For more information about the column items, see [the following](#).

Icon item	Description
	Click the CSV button to save the query results as a CSV file.
 Column Settings	You can hide, add, or change the order of columns. For more information, see the column settings document .
	Click the new window icon to view the page group list in a new window.

ⓘ If you select **Statistical criteria** as **percentile**, you can additionally check the count per page group and the elapsed time for **50quantile**, **75quantile**, **90quantile**, and **95quantile**.

Navigating from Page group list to User Session Log Search

Click the > icon in the **Page group** column to navigate to **User Session Log Search**. You can check the results by applying the selected page group URL and query time as filters.

When navigating from **Page Load Statistics** to **User Session Log Search**, data before and after the specified time range may also be included.

- **Query of 2 hours or more:** Data is queried in 1-hour units, and 1 hour is added before and after the query time range.

Ex. Query range: 11:00 - 14:00 (3-hour query) → Actual query range: 10:00 - 15:00 (5-hour total query)

- **Query of 2 hours or less:** Data is queried in 5-minute units, and 5 minutes are added before and after the query time range.

Ex. Query range: 11:00 - 11:30 (30-minute query) → Actual query range: 10:55 - 11:35 (40-minute total query)

Page load statistics details

Click the [🔍 Detail](#) icon in the **Page group list** list on the **Page Load Statistics** screen to navigate to the **Page load statistics details** window, which shows the statistics for the selected page group.

- **Page load timeline**

Provides the overall page load performance information for the web application. You can check the page load time and the process for each load step as average time. For more information, see [the following](#).

❗ If you select **Statistical criteria** as **percentile**, the page load performance is displayed as percentiles. Blue represents **50quantile**, purple represents **75quantile**, orange represents **90quantile**, and green represents **95quantile**.

- **Page load time**

Displays the page load time by time zone for the selected page group. You can select the desired performance metric from the dropdown list on the right of the page load time area. For more information, see [the following](#).

❗ If you select **Statistical criteria** as **percentile**, the page load time by percentile is provided.

- **Page load count**

Displays the page load count by time zone for the selected page group.

- **Top 10 page loads by Browser / Top 10 page loads by OS / Top 10 page loads by Device**

Provides a pie chart of the top 10 browsers, operating systems, and devices based on the number of page loads. This allows you to understand the environment of users accessing the web application.

- **Page load times by Browser / Page load times by OS / Page load times by Device**

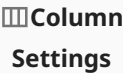
Provides the page load time by browser, operating system, and device. This allows you to identify the user environment where page load times are slow.

❗ Switching between graph/table view

Click the  **Graph** and  **Table** icons in the upper right to switch the view format. The table format provides additional elapsed time and detailed time information.

• Page URL Path list

This is a table that provides performance metrics based on URL paths (Path) that are not grouped by regular expressions. You can check the results by applying the selected URL and query time as filters. If you hover the mouse over a table column header, detailed information about that column is provided as a tooltip. For more information about the column items, see [the following](#).

Icon item	Description
	Click the CSV button to save the query results as a CSV file.
 Column Settings	You can hide, add, or change the order of columns. For more information, see the column settings document .
	Click the new window icon to view the page group list in a new window.
	Click the  icon in the Page URL Path column to navigate to User Session Log Search . You can check the results by applying the selected URL and query time as filters.

❗ When navigating from **Page URL Path** to **User Session Log Search**, data before and after the specified time range may also be included.

- **Query of 2 hours or more:** Data is queried in 1-hour units, and 1 hour is added before and after the query time range.
Ex. Query range: 11:00 - 14:00 (3-hour query) → Actual query range: 10:00 - 15:00 (5-hour total query)
- **Query of 2 hours or less:** Data is queried in 5-minute units, and 5 minutes are added before and after the query time range.



Ex. Query range: 11:00 - 11:30 (30-minute query) → Actual query range: 10:55 - 11:35 (40-minute total query)



Data collection restrictions

- Up to 5,000 page URL paths are generated every 5 minutes and sent to the server.
- If the number of unique URL paths exceeds 5,000 within 5 minutes, additional data is ignored.

Data accuracy

Some detailed time information from the **Navigation Timing API** may be collected as `0`. Certain performance metrics may not be displayed.

Chart interval

The chart interval is determined by the time unit of the queried data.

- If you query with 1-hour data, the chart interval is displayed as 1 hour.
- If you query with 5-minute data, the chart interval is displayed as 5 minutes.

Filtering the queried results

You can filter the retrieved statistical data based on **Page group** or **Performance metrics (ms)** to focus on analyzing the desired content.

Click the **filter** input field at the top of the screen and specify the filter under **Add filter**. Filter tags are entered in the order of `filter key`, `condition`, and `search value`.

Entering the filter key, condition, and search value

- When entering the `filter key`, options for page group and performance metrics (ms) are provided.

- When entering the **condition** , if the **filter key** is page group, **match** and **not match** options are provided; if it is a performance metric, **>** , **<** , **<=** , **>=** , **==** options are provided.
- When entering the **search value** , matching searches (**>** , **<** , **<=** , **>=** , **==**) are highlighted in blue, and exclusion searches (**not match**) are highlighted in red.

! For more information about the performance metrics that can be selected, see [the following](#).

Removing filter tags

- Click the X icon on a tag to delete the applied tag.
- Click the clear all ✕ icon in the input field to delete all tags.

Setting columns

You can hide table header columns or add desired items in **Page group list**. You can also change the column order. Click the  **Column Settings** button.

- You can sort the list on the left by clicking the  (ascending) /  (descending) icon.

Adding columns

1. In the **Column setting** window, select the column items to add from the list on the left.
 - To select all columns, click **Select all** at the top of the list on the left.
2. Check that the selected column items have been added to the **Displayed items** on the right.
3. Once you have added all the desired column items, click the **Confirm** button.

! For more information about column entries, see [the following](#).

Deleting columns

1. In the **Column setting** window, click the  on the right of the column item to delete from the **Displayed items**.
2. Check that the checkboxes for the deleted column items are unchecked in the list on the left.
3. Once you have deleted all the desired column items, click the **Confirm** button.

Changing the column order

You can change the order of columns.

1. In the **Column setting** window, drag items to reorder from the **Displayed items** list and move them to the desired position.
2. After all settings are finished, click the **Confirm** button.

Initializing the column settings

You can cancel all changes and reset them.

1. In the **Column setting** window, click the  **Initialized** button.
2. To reflect the initialized settings, click the **Confirm** button.

Interpreting the queried results with AI Analysis

Click the **AI Analysis** button on the option bar to have WhaTap AI read the performance trend, the top page groups, and the browser, OS, and device summaries currently on screen together, and report what it finds. The button shows an icon only; hover over it to see its name in a tooltip.

1. Set the query conditions and click the  button to query the data.
2. Click the **AI Analysis** button on the option bar.
3. The **AI Analysis** drawer opens on the right of the screen and the result is displayed as a stream.

- Page group names in the result are shown as links. Clicking a link opens the same detail window as clicking the detail view in the list.
- If you change the query conditions and query again, the result for the previous conditions is cleared.
- If the time input is invalid, the button is disabled. Hover over it to see the reason, `An invalid time is selected.`

ⓘ For how AI Analysis behaves in common and the data scope it uses, refer to [the following document](#).

Columns and terms

Column	Unit	Description
Backend time	millisecond	Time processed on the backend while loading the page. (Time from <code>startTime</code> to <code>responseEnd</code>)
Frontend time	millisecond	Time processed on the frontend area while loading the page. (Time from <code>domInteractive</code> to <code>loadEventEnd</code>)
Redirect time	millisecond	Redirection time taken for the browser while loading the web page. (Time from <code>redirectStart</code> to <code>redirectEnd</code>)
Caching time	millisecond	Time taken for the browser to retrieve cached resources while loading the web page. (Time from <code>fetchStart</code> to <code>domainLookupStart</code>)
DNS time	millisecond	Time taken for the browser to query the website domain while loading the web page. (Time from <code>domainLookupStart</code> to <code>domainLookupEnd</code>)
Server connection time	millisecond	Time taken to establish a network connection between the browser and the server while loading a page. It is the time taken for the client and server to establish a connection.
SSL time	millisecond	Time taken for TLS handshaking. (Time from <code>connectEnd</code> to <code>secureConnectionStart</code>) Measured only when the last request used HTTPS.

Column	Unit	Description
FirstByte time	millisecond	Time taken for the browser to receive the first byte from the server after sending a network request while loading the web page. (Time from <code>requestStart</code> to <code>responseStart</code>)
Download time	millisecond	Time taken for the browser to download resources from the server while loading the web page. (Time from <code>responseStart</code> to <code>responseEnd</code>)
DomContentLoaded time	millisecond	Time taken for the web page to render its initial content and become interactive. (Time from <code>domContentLoadedEventStart</code> to <code>domContentLoadedEventEnd</code>)
Dom load time	millisecond	Time until all resources on the web page are loaded, rendering is complete, and the user can interact. (Time from <code>loadEventStart</code> to <code>loadEventEnd</code>)
Rendering time	millisecond	Time taken to render resources downloaded from the server on the screen and complete the page load event. (Time from <code>domInteractive</code> to <code>domComplete</code>)
onLoad event time	millisecond	Time until the load event measured by the browser is completed. (Time from start to <code>loadEventEnd</code>)
Elapsed time	millisecond	Time taken for the browser to completely load the web page.
Count	Count	Number of page loads.
Page group	URL text	A statistical group that provides URL paths normalized using regular expressions. For more information, see the following .
Page URL Path	URL text	A URL path that has not been normalized using a regular expression.
Browser	Text	The type of browser used by the end user, such as Chrome, Edge, Safari, etc.
OS	Text	The type of OS used on the end user's device, such as macOS, Windows, Linux, etc.

Column	Unit	Description
Device	Text	The type of device used by the end user, such as Desktop, Mobile, Tablet, etc.

ⓘ For more information about **Navigation timing**, see [the following](#).

IP Load Statistics

Home > Select Project > **Statistics** > **IP Load Statistics**

IP Load Statistics displays the load performance for each IP address connected to the website. It provides statistics based on average, maximum, minimum, and percentile values, and allows you to identify IP addresses with performance degradation. It also allows you to filter and analyze the page load process for each IP address in detail based on Navigation Timing data.

IP load time is the time taken from loading all resources (CSS, JS, images, fonts, etc.) of a web page from a specific IP address to the complete rendering of all content on the screen. The browser agent uses the **Navigation Timing API** to collect this data.

! For more information about the data collected in **IP Load Statistics**, see [the following](#).

Checking the IP load statistics

It provides performance metrics for each page accessed from a specific IP address. You can check various performance metrics such as **IP load time**, **DOM load time**, **network delay time**, and **resource load time** for each IP address. This allows you to evaluate the performance status of each IP address and determine whether optimization is needed.

1. Go to **Statistics** > **IP Load Statistics** menu.
2. In **Time**, set the date and time to query the performance metrics.
 - **Configurable time range**: Minimum 5 minutes, maximum 3 days
 - **Data query unit**: 2 hours or less → 5-minute intervals, more than 2 hours → 1-hour intervals
3. In **Statistical criteria**, select the desired criteria.
4. Click the  button on the upper right of the screen.
5. You can check the performance metrics for the queried time range through graph charts and lists in each section. If you hover the mouse over the graph chart in each section, you can check the metric information for that time zone.

! For more information about **Statistical criteria**, see [the following](#).

Selecting the statistical criteria

Statistical criteria is an option that allows users to check the statistical metrics to analyze based on the selected criteria.

Statistical criteria	Description	
Avg	Provides statistics by calculating the average load time for each IP address. However, since intermittent performance degradation may be difficult to detect, you can use the maximum and percentile options together to check when performance degradation occurs and its severity.	
Max	Provides statistics based on the maximum load time for each IP address. That is, you can identify which IP address has the most inefficient page load time.	
Min	Provides statistics based on the minimum load time for each IP address. This allows you to identify which IP address has been loaded under the most optimal conditions. It is also useful when setting performance optimization goals or evaluating how effective a specific optimization is. It is more effective to analyze together with the average or percentile option rather than using it alone.	
percentile	Provides load time statistics based on percentiles for each IP address. It shows performance data more accurately than a simple average. You can identify the proportion of users experiencing performance degradation and select pages that need optimization.	
	95quantile	The slowest 5% of IP load times, representing the experience of the top 5% of users with the slowest IP load times.
	90quantile	The slowest 10% of IP load times, primarily useful for identifying performance issues among slow user groups.

Statistical criteria	Description	
	75quantile	Shows the load time of the top 25% of IP load times that are slower than the bottom 75%.
	50quantile	The median, meaning half of all IP addresses loaded the page faster than this time and half loaded it slower.

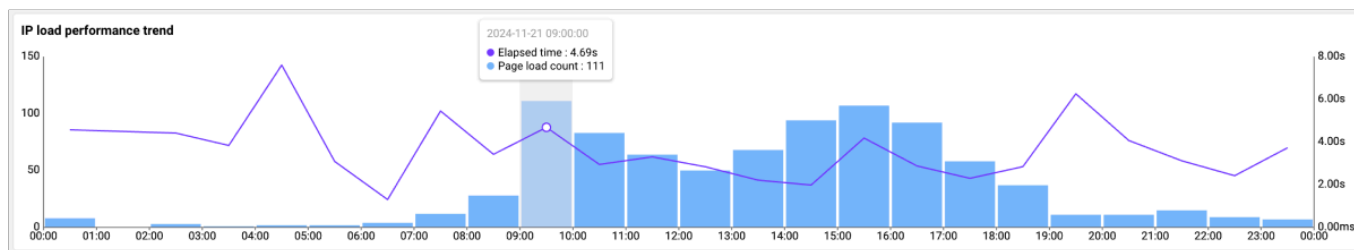
IP load performance trend

This chart visualizes the trend for IP load performance over time. This graph allows you to check whether there were any performance fluctuations during a specific time zone. The items displayed in the chart vary depending on **Statistical criteria**.

- The **blue bar graph** shows the number of pages loaded for a specific time zone. If you click the blue bar graph for a specific time, you can view the data in the **Top 10 page load times**, **Top 10 page load counts**, and **IP list** sections for the selected time range.

Statistical criteria: Avg/Max/Min

The purple line graph is the average page load time for that time zone.



Statistical criteria: percentile

Blue represents **50quantile**, purple represents **75quantile**, orange represents **90quantile**, and green represents **95quantile**.

Page load performance trend

Top 10 page load counts by IP - Elapsed time

IP addresses are grouped at 5-minute or 1-hour intervals and the top 10 IP groups with the longest page load time per unit are displayed in a time series graph. The graph is sorted in descending order of page load time by IP address.

- The bar chart at the bottom of the section displays the **average value per IP address** of the data represented in the time series graph.
- If data is not collected in continuous time intervals but exists only in specific intervals, only the bar graph is displayed.
- **How elapsed time is displayed based on the statistical criteria**
 - **Avg, Max, Min:** Displays elapsed time using the selected **reference value**.
 - **percentile:** Displays elapsed time as **95quantile**.

Top 10 page load counts by IP - Count

IP addresses are grouped at 5-minute or 1-hour intervals and the top 10 IP groups with the highest page load count per unit are displayed in a time series graph. The graph is sorted in descending order of page load count by IP address.

- The bar chart at the bottom of the section displays the **sum value per page group** of the data represented in the time series graph.
- If data is not collected in continuous time intervals but exists only in specific intervals, only the bar graph is displayed.

IP list

This is a list where you can check various performance metrics based on the IP group.

Icon item	Description
	Click the CSV button to save the query results as a CSV file.

Icon item	Description
 Column Settings	You can hide, add, or change the order of columns. For more information, see the column settings document .
	Click the new window icon to view the IP list in a new window.

ⓘ If you select **Statistical criteria** as **percentile**, you can additionally check the count per page group and the elapsed time for **50quantile**, **75quantile**, **90quantile**, and **95quantile**.

Navigating from IP Load Statistics to IP load statistics detail page

Click the > icon in the **IP list** list to navigate to **User Session Log Search**. You can check the results by applying the selected page group URL and query time as filters.

When navigating from **IP Load Statistics** to **User Session Log Search**, data before and after the specified time range may also be included.

- **Query of 2 hours or more:** Data is queried in 1-hour units, and 1 hour is added before and after the query time range.

Ex. Query range: 11:00 - 14:00 (3-hour query) → Actual query range: 10:00 - 15:00 (5-hour total query)

- **Query of 2 hours or less:** Data is queried in 5-minute units, and 5 minutes are added before and after the query time range.

Ex. Query range: 11:00 - 11:30 (30-minute query) → Actual query range: 10:55 - 11:35 (40-minute total query)

IP load statistics detail page

Click the  **Detail** icon in the **IP list** list on the **IP Load Statistics** screen to navigate to the IP load statistics detail page window, which shows detailed statistics for the selected IP group.

- **IP load timeline**

Provides the IP-based page load performance of the web application. It displays the page load time and the average time for each load step. For more information, see [the following](#).

- ❗ If **Statistical criteria** is set to **percentile**, the IP load performance is displayed as percentiles. Blue represents **50quantile**, purple represents **75quantile**, orange represents **90quantile**, and green represents **95quantile**.

- **Page load times by IP**

Displays the page load time by time zone for the selected IP group. You can select the desired performance metric from the dropdown list on the right of the **Page load times by IP** area. For more information, see [the following](#).

- ❗ If you select **Statistical criteria** as **percentile**, the IP load time by percentile is provided.

- **Page load counts by IP**

Displays the page load count by time zone for the selected IP group.

- **IP list**

This is a table that provides various performance metrics based on the IP address. If you hover the mouse over a table column header, detailed information about that column is provided as a tooltip. For more information about the column items, see [the following](#).

Icon item	Description
	Click the CSV button to save the query results as a CSV file.
 Column Settings	You can hide, add, or change the order of columns. For more information, see the column settings document .
	Click the new window icon to view the resource group list in a new window.
	Click the > icon in the IP list column to navigate to User Session Log Search . You can check the results by applying the selected IP address and query time as filters.

- ① When navigating from **IP list** to **User Session Log Search**, data before and after the specified time range may also be included.
- **Query of 2 hours or more:** Data is queried in 1-hour units, and 1 hour is added before and after the query time range.
Ex. Query range: 11:00 - 14:00 (3-hour query) → Actual query range: 10:00 - 15:00 (5-hour total query)
 - **Query of 2 hours or less:** Data is queried in 5-minute units, and 5 minutes are added before and after the query time range.
Ex. Query range: 11:00 - 11:30 (30-minute query) → Actual query range: 10:55 - 11:35 (40-minute total query)

① **Data collection restrictions**

- Up to 5,000 page URL path lists are generated every 5 minutes and sent to the server.
- If the number of unique URL paths exceeds 5,000 within 5 minutes, additional data is ignored.

Data accuracy

Some detailed time information from the **Navigation Timing API** may be collected as 0. This may cause certain performance metrics not to be displayed.

Chart interval

The chart interval is determined by the time unit of the queried data.

- If you query with 1-hour data, the chart interval is displayed as 1 hour.
- If you query with 5-minute data, the chart interval is displayed as 5 minutes.

Filtering the queried results

You can filter the retrieved statistical data based on **IP** or **Performance metrics (ms)** to focus on analyzing the

desired content.

Click the **filter** input field at the top of the screen and specify the filter under **Add filter**. Filter tags are entered in the order of **filter key** , **condition** , and **search value** .

Entering the filter key, condition, and search value

- When entering the **filter key** , options for page group and performance metrics (ms) are provided.
- When entering the **condition** , if the **filter key** is page group, **match** and **not match** options are provided; if it is a performance metric, **>** , **<** , **<=** , **>=** , **==** options are provided.
- When entering the **search value** , matching searches (**>** , **<** , **<=** , **>=** , **==**) are highlighted in blue, and exclusion searches (**not match**) are highlighted in red.

! For more information about the performance metrics that can be selected, see [the following](#).

Removing filter tags

- Click the X icon on a tag to delete the applied tag.
- Click the clear all  icon in the input field to delete all tags.

Setting columns

You can hide table header columns or add desired items in **Page group list**. You can also change the column order. Click the  **Column Settings** button.

- You can sort the list on the left by clicking the  (ascending) /  (descending) icon.

Adding columns

1. In the **Column setting** window, select the column items to add from the list on the left.
 - To select all columns, click **Select all** at the top of the list on the left.
2. Check that the selected column items have been added to the **Displayed items** on the right.

3. Once you have added all the desired column items, click the **Confirm** button.

! For more information about column entries, see [the following](#).

Deleting columns

1. In the **Column setting** window, click the **✗** on the right of the column item to delete from the **Displayed items**.
2. Check that the checkboxes for the deleted column items are unchecked in the list on the left.
3. Once you have deleted all the desired column items, click the **Confirm** button.

Changing the column order

You can change the order of columns.

1. In the **Column setting** window, drag items to reorder from the **Displayed items** list and move them to the desired position.
2. After all settings are finished, click the **Confirm** button.

Initializing the column settings

You can cancel all changes and reset them.

1. In the **Column setting** window, click the **↺ Initialized** button.
2. To reflect the initialized settings, click the **Confirm** button.

Interpreting the queried results with AI Analysis

Click the **AI Analysis** button on the option bar to have WhaTap AI read the per-IP counts and performance summary currently on screen and report what it finds. Use it to check whether a small number of IP addresses account for the traffic, or whether performance drops only for a particular IP address. The button shows an icon

only; hover over it to see its name in a tooltip.

1. Set the query conditions and click the  button to query the data.
 2. Click the **AI Analysis** button on the option bar.
 3. The **AI Analysis** drawer opens on the right of the screen and the result is displayed as a stream.
- IP addresses in the result are shown as links. Clicking a link opens the same detail window as clicking the detail view in the list.
 - IP addresses are used for the analysis with the last octet masked. User IDs are converted to hash values before being sent.
 - If the time input is invalid, the button is disabled. Hover over it to see the reason, `An invalid time is selected.`

! For how AI Analysis behaves in common and the data scope it uses, refer to [the following document](#).

Columns and terms

Column	Unit	Description
Backend time	millisecond	Time taken for the backend to process a page load from the IP address. (Time from <code>startTime</code> to <code>responseEnd</code>)
Frontend time	millisecond	Processing time of the frontend area while loading a page from the IP address. (Time from <code>domInteractive</code> to <code>loadEventEnd</code>)
Redirect time	millisecond	Time taken for redirection to load a web page from the IP address. (Time from <code>redirectStart</code> to <code>redirectEnd</code>)
Caching time	millisecond	Time taken to retrieve a cached resource from the IP address. (Time from <code>fetchStart</code> to <code>domainLookupStart</code>)
DNS time	millisecond	Time taken to query the website domain from the IP address. (Time from <code>domainLookupStart</code> to <code>domainLookupEnd</code>)

Column	Unit	Description
Server connection time	millisecond	Time taken for the network connection between the client requesting the IP and the server.
SSL time	millisecond	Time taken for TLS handshaking when connecting via HTTPS. (Time from <code>connectEnd</code> to <code>secureConnectionStart</code>)
FirstByte time	millisecond	Time taken to receive the first byte from the server after an IP request. (Time from <code>requestStart</code> to <code>responseStart</code>)
Download time	millisecond	Time taken to download IP-related resources from the server. (Time from <code>responseStart</code> to <code>responseEnd</code>)
DomContentLoaded time	millisecond	Time taken for the initial content rendering from the IP address.
Dom load time	millisecond	Time taken for all resources to load and fully render in the state where the end users can interact with them.
Rendering time	millisecond	Time taken to load a page from the IP address, render on screen, and complete the page event.
onLoad event time	millisecond	Time until the page load from the IP address is complete and the <code>onLoad</code> event fires. It indicates the time from retrieving the IP address and establishing a network connection until all resources are loaded.
Elapsed time	millisecond	Total time taken to fully load a page from the IP address.

ⓘ For more information about **Navigation timing**, see [the following](#).

Resource Load Statistics

Home > Select Project > **Statistics** > **Resource Load Statistics**

Resource Load Statistics provides the criteria needed to improve web page performance. It visualizes performance data for each resource on the web page to show performance trends, response times, and sizes at a glance. This allows you to understand the impact of each resource on page load and analyze performance bottlenecks. You can identify resources with long load times, optimize large-scale resources, and check the performance impact of 3rd party resources.

! For more information about the collected data, see [the following](#).

Checking the resource load statistics

It provides data needed to reduce the page load time and improve user experience. By clearly identifying the cause of performance degradation, improvement efforts can be carried out more effectively.

1. Go to **Statistics** > **Resource Load Statistics** menu.
2. In **Time**, set the date and time to query the performance metrics.
 - **Configurable time range:** Minimum 5 minutes, maximum 3 days
 - **Data query unit:** 2 hours or less → 5-minute intervals, more than 2 hours → 1-hour intervals
3. In **Statistical criteria**, select the desired option. For more information, see the [Statistical criteria](#) document.
4. Click the  button on the upper right of the screen.
5. The performance metrics for the selected time range are displayed as charts and lists in each section.
 - If you hover the mouse over the graph, you can check the detailed metrics for that time zone.

Selecting the statistical criteria

Statistical criteria is an option that allows users to check the statistical metrics to analyze based on the selected

criteria.

Statistical criteria	Description
Avg	Provides statistics by calculating the average load time for each resource. It is useful for understanding the overall performance.
Max	Provides statistics based on the maximum load time for each resource. You can identify which resource has the most inefficient page load time. It can be used for analyzing sections with longer response times or larger sizes.
Min	Provides statistics based on the minimum load time for each resource. You can identify which resource has the fastest page load time.

Load count trends by resource type

Visualizes the load count by resource type on an hourly basis. If the load ratio of a specific resource type is excessively high, it may cause a performance bottleneck. In that case, analyze the cause of the resource and proceed with optimization.

- The **pie chart** displays the ratio by resource type. If you hover the mouse over the chart, you can check the ratio of the resource as a tooltip.
- The **bar graph** displays the trend for the number of resource loads per time unit. If you hover the mouse over the graph, you can check the load count by resource type for that time zone as a tooltip. When the user selects a specific time in the bar graph, the data in the **Resource Load Count Top 10 - Elapsed time, Size** and **Resource group list** sections is updated for the selected time range.

ⓘ For more information about the resource types, see [the following](#).

Resource load count - Elapsed time

Provides the **response times of the top 10 resources with the highest load count** as a chart and list. You can

quickly identify resources causing performance degradation. You can also check the response time trend over time to analyze delays occurring in specific time zones. It can be used to identify resources with long response times and optimize server response time and network delays.

Resource load count - Count

Provides the **sizes of the top 10 resources with the highest load count** as a chart and list. You can identify large-scale resources causing poor web page loading performance. You can improve page performance through image optimization, JS bundling, and compression.

Resource group list

Provides a list of all resources loaded on the web page. You can check detailed performance metrics for each resource. You can analyze the load path and request count for a resource to reduce unnecessary repeated requests. You can check the performance status of third-party APIs or CDNs by checking the **Resource host** column of external resources (**3rd party**). For more information, see [the following](#).

- Click the header column in the table to sort the list based on the column.
- Hover the mouse over the header column to see a tooltip explaining the column.

Icon item	Description
	Click the CSV button to save the query results as a CSV file. - CSV file name: <code>Resource_LoadStatistics_YYYYMMDD_HHMMSS.csv</code> format
 Column Settings	You can hide, add, or change the order of columns. For more information, see the column settings document .
	Click the new window icon to view the resource group list in a new window.

Resource load statistics details

Click the  **Detail** icon in the **Resource group list** list on the **Resource Load Statistics** screen to navigate to the

Resource load statistics details window, which shows statistics for the selected resource group.

- **Resource load timeline**

Provides the overall load performance for the selected resource group. It displays the total resource load time and the average time for each load step, allowing you to understand the loading performance of the browser application at a glance. For more information, see [the following](#).

- **Resource page load time trend**

Checks the page load time by period for the selected resource group. You can select various performance metrics from the metric selection box. For more information about performance metrics, see [the following](#).

- **Resource page load size trend**

Checks the total size of page resources loaded for each time zone of the selected resource group. You can analyze the network load and page optimization status through the page load size trend.

- **Resource page load count trend**

Checks the number of resources loaded for each time zone of the selected resource group. If the number of resources suddenly increases or increases abnormally, you can check whether excessive resource requests were made from a specific page.

- **DNS time / Server connection time / FirstByte time / Download time**

Checks the average time taken for DNS query, server connection setup, first byte reception, and resource download for each time zone.

- **Resource group list**

This is a list where you can check various performance metrics for individual resources included in the selected resource group.

Filtering the queried results

You can filter the retrieved statistical data by domain, performance metrics (byte, ms), or resource path to focus on analyzing the desired items.

1. Click the **filter** input field at the top of the screen.
2. Specify the filter in the **Add filter** window. Filter tags are entered in the order of **filter key** , **condition** , and **search value** .

- Using the **Domain classification** option, you can distinguish which entity is providing the resource. For more information, see [Domain filter](#).

Domain filter

Using the **Domain classification** option, you can distinguish which entity is providing the resource and determine whether the factor affecting the performance is **internal (1st Party)** or **external (3rd Party)**. This allows you to effectively analyze where bottlenecks occur when optimizing the web page performance.

- **1st Party:** You can filter only the resources provided by the company. It includes web servers and self-hosted resources.

✓ Check **1st Party** resources to identify performance bottlenecks on internal servers.

- **3rd Party:** Resources provided by external services. It includes all resources loaded from external domains such as CDN resources, external APIs (e.g. Google Analytics), and third-party libraries.

✓ Analyze the response times of **3rd Party** resources to evaluate the impact of external services or CDNs on web page performance.

Entering the filter key, condition, and search value

- When entering the **filter key**, options for resource path, resource type, and performance metrics (byte, ms) are provided.
- When entering the **condition**, if the **filter key** is resource path or resource type, **match** and **not match** options are provided; if it is a performance metric, **>**, **<**, **<=**, **>=**, **==** options are provided.
- When entering the **search value**, matching searches (**>**, **<**, **<=**, **>=**, **==**) are highlighted in blue, and exclusion searches (**not match**) are highlighted in red.

! For more information about the performance metrics that can be selected, see [the following](#).

Removing filter tags

- Click the X icon on a tag to delete the applied tag.
- Click the clear all  icon in the input field to delete all tags.

Resource types

The following guides the resource types defined in WhaTap Browser Monitoring.

Resource type	Description
script	JavaScript file resource. For example, <code>.js</code> files are script files required for running front-end code and dynamic features.
image	Image file resource. For example, it includes various image formats such as <code>.png</code> , <code>.jpg</code> , <code>.svg</code> , etc.
fetch	An HTTP request to retrieve data from the browser. It is mainly used for API calls or JSON data requests.
css	Stylesheet file resource. It is the CSS file that defines the layout and design for web pages.
font	Web font resource. For example, it includes font files such as <code>.woff</code> , <code>.ttf</code> , <code>.otf</code> , etc.
xhr	Resource retrieved through XMLHttpRequest. It is mainly used for asynchronous data requests (e.g. AJAX calls).
media	Media file resource such as audio and video. For example, it includes various media formats such as <code>.mp4</code> , <code>.mp3</code> , <code>.wav</code> , etc.
beacon	A request to send asynchronous data to the server using <code>navigator.sendBeacon()</code> .
document	The HTML document itself, which is a major content resource that makes up the page.
other	Resource classified as Other. It includes resources that are not clearly classified by type.

Setting columns

You can hide table header columns or add desired items in **Resource group list**. You can also change the column order. Click the  **Column Settings** button.

Column Settings

Please select a column to display

Search by column name

☒ Select All

 Initialized 

☒ Count

☒ DNS time

☒ Download time

☒ Elapsed time

☒ Environment

☒ FirstByte time

☒ Redirect time

☒ resource host

☒ Resource path

☒ resource type

☒ server connection time

☒ Size

☒ SSL time

☒ Version

Display columns(14)

|| Resource path  

|| Environment 

|| Version 

|| resource host 

|| resource type 

|| Count 

|| Size 

|| Elapsed time 

|| Redirect time 

|| DNS time 

|| server connection time 

|| SSL time 

Cancel

Confirm

! You can sort the list on the left by clicking the  (ascending) /  (descending) icon.

Adding columns

1. In the **Column setting** window, select the column items to add from the list on the left.
 - To select all columns, click **Select all** at the top of the list on the left.
2. Check that the selected column items have been added to the **Displayed items** on the right.
3. Once you have added all the desired column items, click the **Confirm** button.

! For more information about column entries, see [the following](#).

Deleting columns

1. In the **Column setting** window, click the  on the right of the column item to delete from the **Displayed items**.
2. Check that the checkboxes for the deleted column items are unchecked in the list on the left.
3. Once you have deleted all the desired column items, click the **Confirm** button.

Changing the column order

You can change the order of columns.

1. In the **Column setting** window, drag items to reorder from the **Displayed items** list and move them to the desired position.
2. After all settings are finished, click the **Confirm** button.

Initializing the column settings

You can cancel all changes and reset them.

1. In the **Column setting** window, click the  **Initialized** button.
2. To reflect the initialized settings, click the **Confirm** button.

Interpreting the queried results with AI Analysis

Click the **AI Analysis** button on the option bar to have WhaTap AI read the per-resource counts, sizes, and load time summary currently on screen and report what it finds. Use it to narrow down how large resources or third-party resources affect page load. The button shows an icon only; hover over it to see its name in a tooltip.

1. Set the query conditions and click the  button to query the data.
 2. Click the **AI Analysis** button on the option bar.
 3. The **AI Analysis** drawer opens on the right of the screen and the result is displayed as a stream.
- Resource statistics treat resource size as an axis alongside latency, so the result is also presented by resource type and by whether the resource is third-party.
 - If the time input is invalid, the button is disabled. Hover over it to see the reason, An invalid time is selected.

 For how AI Analysis behaves in common and the data scope it uses, refer to [the following document](#).

Columns and terms

Column	Unit	Description
Count	Count	Total number of resource requests.
Elapsed time	millisecond	Time taken to fully load the resource.
Download time	millisecond	Time taken to download resources from the server. (Time from <code>responseStart</code> to <code>responseEnd</code>)
Redirect time	millisecond	Time taken for redirect while loading resources. (Time from <code>redirectStart</code>

Column	Unit	Description
		to <code>redirectEnd</code>)
Resource type	Text	Type of the requested resource. For more information, see the following .
Resource host	Text	Host name of the server providing the resource.
Resource path	Text	URL path or file path of the requested resource.
Version	Text	Version information of the resource.
Server connection time	millisecond	Time taken for the network connection between the client requesting the resource and the server.
Size	Byte	Data size of the requested resource.
Environment	Text	Operating environment where the resource is loaded. e.g. development environment, operating environment, etc.
DNS time	millisecond	Time taken to query the domain name. (Time from <code>domainLookupStart</code> to <code>domainLookupEnd</code>)
FirstByte time	millisecond	Time taken until the first byte is received from the server. (Time from <code>requestStart</code> to <code>responseStart</code>)
SSL time	millisecond	Time taken for TLS handshaking when connecting via HTTPS. (Time from <code>connectEnd</code> to <code>secureConnectionStart</code>)

ⓘ For more information about **Navigation timing**, see [the following](#).

Error Load Statistics

Home > Select Project > **Statistics** > **Browser Error Statistics**

In **Browser Error Statistics**, you can efficiently analyze errors that have occurred by visually providing the overall error status. With various filter options, you can group the desired data or retrieve information that meets specific conditions, enabling more detailed error analysis.

Checking the error load statistics

Through numerical error messages, you can monitor error occurrence trends in real time and analyze the occurrence count and trend for each error. Set the time range to query to check the error status of that time zone through a chart and list. When you select a specific section in the bar chart, you can see the error status of that section in the error table.

1. Go to **Statistics** > **Browser Error Statistics** menu.
2. In **Time**, set the date and time to query the error statistics.
 - **Configurable time range**: Minimum 5 minutes, maximum 3 days
 - **Data query unit**: 2 hours or less → 5-minute intervals, more than 2 hours → 1-hour intervals
3. Select the conditions to query through **Filter** and **Group by**.
4. Click the  button on the upper right of the screen.
5. Check the **load count trend by error message** chart and the **error message list**.
 - The error status for the selected time range is provided as a chart and list. When you select a specific section in the bar chart, you can see the error status of that section in the error table.

Checking user session logs

Click the > icon in the **error message list** to navigate to **User Session Log Search**. You can check the results by applying the selected error type and message as filters.

Selecting the grouping criteria

You can select a grouping option to group the error data based on specific criteria. The grouping criteria include **No grouping**, **Browser**, **Error type**, **Operating system**, and **Device**. Users can view grouped summary information for analysis purposes, or query individual error items as needed.

- **No grouping:** Data is listed as individual items, allowing you to directly check the number of occurrences for each item.
- **Browser:** Groups the data by the browser that caused the error.
- **Error type:** Groups the data by error type.
- **Operating system:** Groups errors by the user's operating system. It is useful for analyzing compatibility issues by operating system or errors that occur in specific environments.
- **Device:** Groups error data by device type.

Load count trend by error message

By visualizing the number of errors occurring for each time zone in the bar chart, you can easily check the frequency of errors in a specific time zone. When you select a specific section in the bar chart, you can see the error status of that section in the error table.

- **Grouping not applied**

You can visualize the total number of errors by time zone in a single graph to understand the overall error occurrence trend.

- **Grouping applied: Browser, Error Type, OS, Device**

You can visualize the number and proportion of errors by group using the stack chart and donut chart to analyze the number of occurrences by time zone in more detail.

Error message list

It provides detailed information such as error messages, number of occurrences, error types, OS, and browser in table format. This table allows you to easily check additional information about each error message.

- **Grouping not applied**

If you do not group the data, all data is listed as individual items. This allows you to directly check the number of occurrences for each item and error messages.

- **Grouping applied: Browser, Error Type, OS, Device**

You can organize the data into a tree structure by grouping it according to the selected criteria. You can also expand/collapse sub-items using the ▼ icon for more efficient navigation.

Icon item	Description
	You can save the queried data as a CSV file. Click the CSV button on the right of the error message list area. - CSV file name: <code>Error_LoadStatistics_YYYYMMDD_HHMMSS.csv</code> format
	Click the > icon in the error message list to navigate to User Session Log Search. You can check the results by applying the selected error type and message as filters.

Interpreting the queried results with AI Analysis

Click the **AI Analysis** button on the option bar to have WhaTap AI read the error trend and the **error message list** currently on screen and report what it finds. Use it to group similar error messages, or to find errors concentrated in a particular browser or OS version. The button shows an icon only; hover over it to see its name in a tooltip.

1. Set the query conditions and click the  button to query the data.
 2. Click the **AI Analysis** button on the option bar.
 3. The **AI Analysis** drawer opens on the right of the screen and the result is displayed as a stream.
- Browser error statistics collect occurrence counts only, so the result does not include elapsed time metrics either.
 - If the time input is invalid, the button is disabled. Hover over it to see the reason, `An invalid time is selected.`

! For how AI Analysis behaves in common and the data scope it uses, refer to [the following document](#).

Columns and terms

Column	Description
Error Message	Content of the error message that occurred
Count	Number of times the error message occurred
Error Type	Type of the error
OS	Operating system on which the error occurred
Browser	Browser on which the error occurred

Error types

Error types (Error Type) that occur in the browser are as follows.

- `console` : This developer-defined error appears on the browser console.
- `onError` : This error occurs when a specific resource fails to be loaded or a script error occurs.
- `fetchError` : This error occurs when sending HTTP requests through the fetch API.
- `xhrError` : This error occurs when sending HTTP requests through XMLHttpRequest (XHR).

Report

Home > Select Project > **Report**

If you click  in **Report** on the screen, you can see various analysis reports classified by daily, weekly, and monthly. Select a desired report format, and then set the date and time to view and filtering options. You can check the selected report by selecting .

Report is a document that reports the service usage status and failure records for individual projects. The persons in charge of service monitoring share the status report with the people of related departments.

The monitoring data analysis is important because it serves as a metric for determining the direction of improvement. However, it is cumbersome to aggregate and document data from multiple dashboards.

WhaTap's **Report** menu supports **automated report creation**. Through the **Report** menu, you can create weekly reports, share reports at the set times, and manage various templates.

Download Report

If you want to download the report or schedule the printing or mailing, click the  Download button. You can download it in HTML format.

Daily browser report

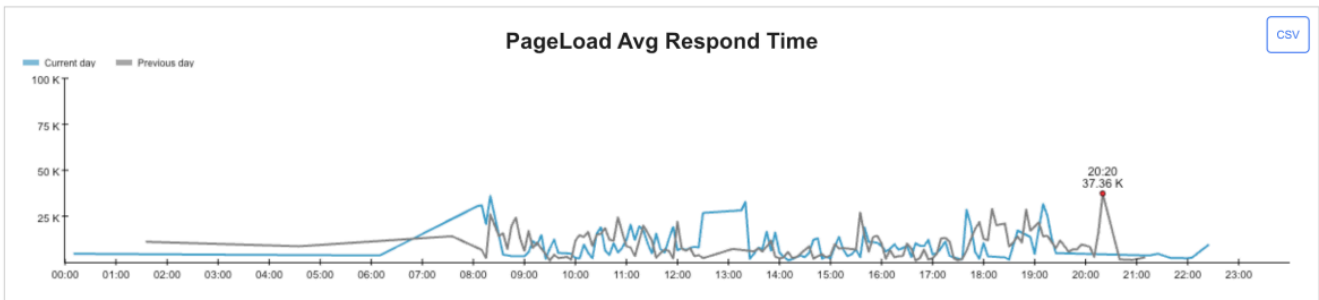
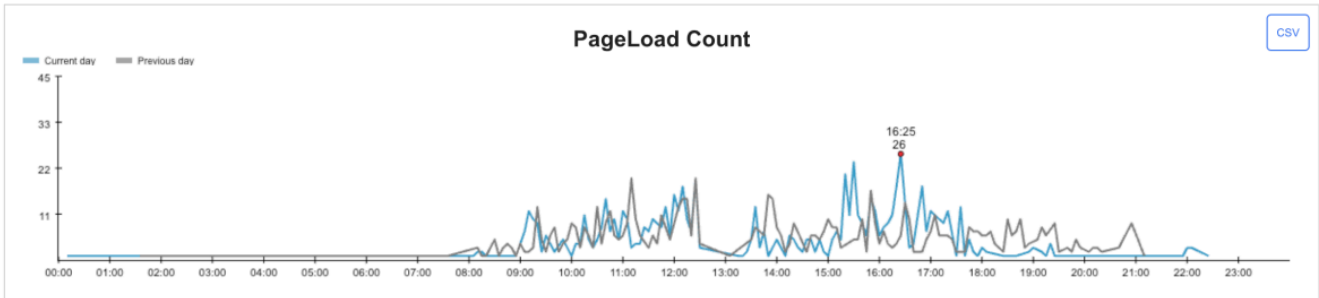
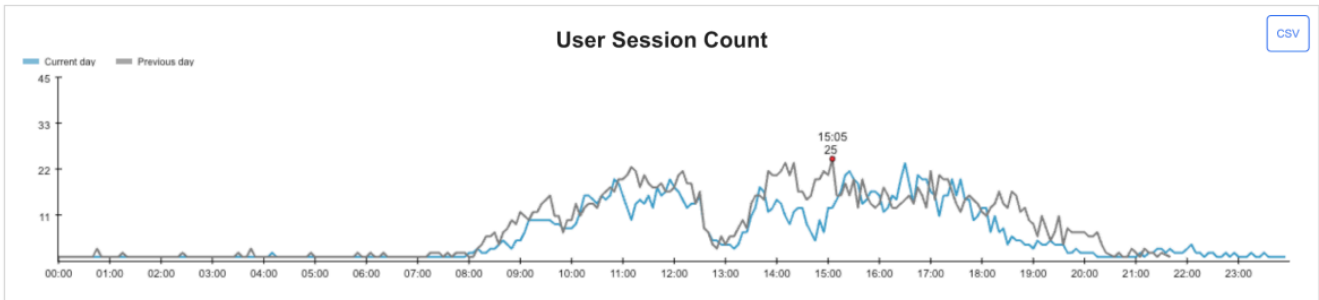
Using the daily browser report, you can see the web page usage of the day. Set the [start date](#), [start time](#), and [end time](#), and then select . A report is divided into the following.

Summary

The web page access performance during the inquiry period can be checked briefly through the figures and charts.

Overview

Daily User Session Number	PageLoad Request	PageLoad Avg Respond Time	AJAX request	AJAX Avg Respond Time	AJAX Error Ratio(%)
281	782	8,453ms	139041	320ms	0.73



- **Unique user sessions:** Number of user sessions during the inquiry period.
- **Page load count:** Number of page loads during the inquiry period.
- **Average page load response time:** Average of all the page load times during the inquiry period.
- **AJAX request count:** Number of AJAX requests during the inquiry period.
- **AJAX average response time:** Average of all AJAX response times during the inquiry period.
- **AJAX error rate:** Ratio of the total AJAX request count divided by the AJAX error count.
- **Chart:** unique user sessions, Page load count, Page load average response time

You can see the changes of key metrics over time through the graph and quickly find the time and figure at which each metric was the highest. On the upper right of the chart, select [CSV](#) to download the data in a CSV file format.

Top 10 page loads for each browser

You can see the top 10 most used browsers of the day. The page load count and average response time appear.

Top 10 page loads for each region

You can see the top 10 most accessed regions of the day. The page load count and average response time appear.

ⓘ Currently, only the regions accessed from South Korea are displayed.

Top 10 page loads

You can see the top 10 page groups that generated the highest page loads during the day. The page load count for each page and average response time appear.

Top 10 router changes

The router change event occurs when web pages are switched in the form of Client Side Rendering (CSR). You can see the top 10 page groups that generated the page loads frequently during the day. The page switching count to the page and average response time appear.

Top 10 AJAX calls

You can see the top 10 AJAX calls during the day based on the AJAX path. The AJAX call count and average response time appear.

Top 10 1st resource call counts

Among the resources called by hosts such as web pages, you can see the top 10 resource calls during the day based on the resource path. Excluding AJAX calls, it displays the resource call count together with the average response time.

Top 10 3rd resource call counts

Among the resources called from the hosts other than web pages, you can see the top 10 resource calls during the day based on the resource path. Excluding AJAX calls, it displays the resource call count together with the average response time.

Top 10 error counts by page

For the number of errors during the day, you can see the top 10 page groups classified by the page group.

Top 10 error counts by error message

The error messages are values that can identify errors in the browser. For the number of errors during the day, you can see the top 10 error messages classified by the error message.

Top 10 AJAX errors

For the AJAX error count during the day, you can see the top 10 AJAX paths classified by the AJAX path.

Recreating a report

Select [Reset](#) to refresh the data in the report. To search the data for a different date or time, set the [start date](#), [start time](#), and [end time](#) and then select [Reset](#) or .

Downloading or printing the report

- **Download:** You can download the report as an HTML file. The file name is saved in the format of `DailyBrowserReport_ yyyymmdd .html`.
- **Print:** You can save or print the report as a PDF file. To save as a PDF file, select **Save as PDF** for the target to print.

ⓘ The function to save as a PDF file may not be supported depending on the web browser.

Email delivery schedule

To receive regular reports by email, select **Reservation of Report Email**. You can replace various service checking tasks performed right after work with email confirmation.

Reservation of Report Email

ⓘ Report will be sent on 04~06 o'clock from tomorrow.

ⓘ Maximum 5 report mail settings can be registered.

Report type

Report time

00:00 ~ 23:59

Weekdays

☒ Mon

☒ Tue

☒ Wed

☒ Thu

☒ Fri

☐ Sat

☐ Sun

Emails

Input email and enter

+ Save

Alert Notifications

This feature delivers notifications to users based on predefined conditions when issues occur in monitored systems or applications. Notifications can be sent through various channels such as email, SMS, messengers, and mobile push notifications.

For example, you can configure alert thresholds for key performance indicators such as CPU usage, memory usage, disk I/O, network traffic, transaction count, error occurrences, and response time latency. When these thresholds are exceeded or abnormal behavior is detected, notifications are automatically sent.

With WhaTap's AI-based alerting, you do not need to define complex conditions manually. The AI learns normal application behavior patterns and automatically detects and notifies anomalies. Additionally, by using tag-count alerts, you can define conditions on values aggregated by specific tags, enabling precise detection even in complex scenarios.

Key Features

- **Event Configuration:** Configure threshold-based alert conditions and notification channels
- **Event Notification Settings:** Configure notification channels and time schedules per user
- **Event History:** View the history of alert occurrences

Permissions

To add or configure events, the **Alert Configuration** permission is required.

- **View:** To view metrics event configurations, the **View & Analyze** permission is required.
- **Add/Edit/Delete:** To modify configurations, the **Alert Configuration** permission is required.
- **Receive Notifications:** To receive notifications, the **Alert Receive** permission is required.

Reference

For more details, see [Member Permissions](#).

Event Configuration

Home screen > Select project  **Alert** > **Event Configuration**

In **Event setting**, you can configure and manage events based on anomaly conditions that may occur in monitored systems or applications. When the configured conditions are met, notifications can be delivered through your preferred channels such as email, SMS, messengers, or app push notifications. You can tailor event configurations according to your monitoring objectives.

Metrics

Metrics events allow you to monitor system and application performance indicators based on real-time data.

Composite Metrics

The composite metrics feature allows you to create events using complex rules and send alert notifications.

Hitmap Patterns

Hitmap patterns automatically detect anomalies based on changes in distributions such as response time and call density, and send alert notifications.

Home Screen > Select Project >  **Alert** > **Event Configuration**

Event setting provides features to configure and manage events based on anomaly signals that may occur in monitored systems or applications. When configured conditions are met, you can receive notifications through your preferred channels such as email, SMS, messengers, or app push notifications. You can design event configurations according to your monitoring objectives.

Metrics

Metrics events allow you to monitor system and application performance indicators based on real-time data.

Composite Metrics

The composite metrics feature allows you to create events using complex rules and send alert notifications.

Hitmap Patterns

Hitmap patterns automatically detect anomalies based on changes in distributions such as response time and call density, and send alert notifications.

Reference

- When configuring events, use appropriate threshold values to prevent unnecessary event generation.
- When setting up notifications, consider enabling recovery notifications (alerts sent when issues are resolved) to support effective operational response.

Common Features

All event configurations can be downloaded in **JSON or Excel format**. By sharing the same type of product configuration, you can reduce repetitive event setup work.

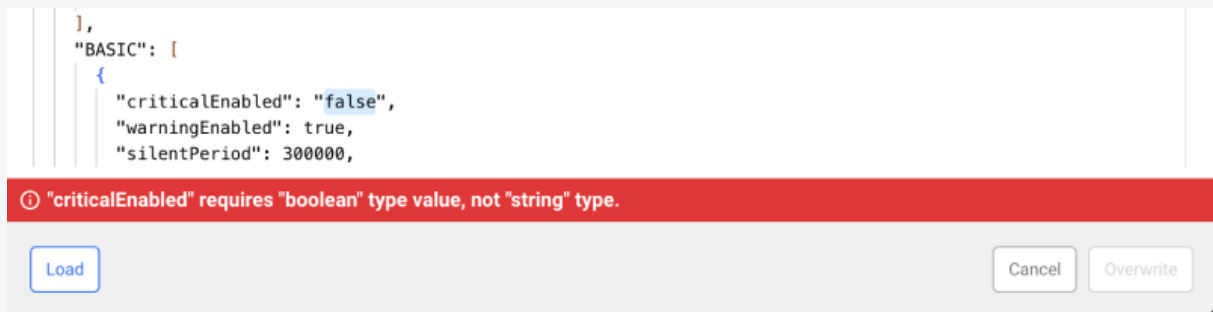
- **Batch JSON Edit:** Edit all event conditions in bulk using JSON format.
- **Batch JSON Download:** Save all currently registered event conditions as a JSON file.
- **Batch Excel Download:** Download the event list as an Excel file for external reporting or backup purposes.

Batch JSON Edit

1. Click the [Batch JSON Edit] button at the top of the screen.
2. Import a JSON file downloaded via the **Batch JSON Download** feature, or edit the content directly.
3. Click the [Overwrite] button to save.
4. When a confirmation message appears, click the [Overwrite] button.

⚠ After overwriting, you cannot restore the previous event configuration.

- Arbitrarily modifying the values of the `productType` or `platform` properties may cause events to malfunction in other projects.
- The JSON editor includes built-in validation. If the format or values are invalid, an error message appears below the editor and the [Overwrite] button is disabled.



JSON Data Structure



```
{
  "productType": "{productType}",
  "platform": "{platform}",
  "events": {
    "HITMAP": {
      ...
    },
    "LOG_REALTIME": [
      {...}
    ],
  },
}
```



```
"METRICS": [
  {...}
],
"COMPOSITE_LOG": [
  {...}
],
"BASIC": [
  {...}
],
"COMPOSITE_METRICS": [
  {...}
],
"ANOMALY": [
  {...}
]
}
}
```

JSON Field	Description
productType	Product type (e.g. APM, Infra)
platform	Platform information (e.g. java, node)
events	Event data
⬠ HITMAP	Hitmap pattern
⬠ LOG_REALTIME	Real-time log events
⬠ METRICS	Metrics
⬠ COMPOSITE_LOG	Composite log events
⬠ BASIC	Basic events
⬠ COMPOSITE_METRICS	Composite metrics

JSON Field	Description
◇ ANOMALY	Anomaly detection

Batch JSON Download

Click the [Batch JSON Download] button at the top of the screen.

- The JSON file name follows the format `integrated-event-YYYYMMDD.json`.
- The structure of the JSON data and the sub-properties under the `events` attribute may vary depending on the product (`platform`) or product type (`productType`).

⚠ If you arbitrarily modify the values of the `productType` or `platform` properties in the JSON data, events may not function correctly in other projects.

Batch Excel Download

Click the [Batch Excel Download] button at the top of the screen.

Metrics

Home > Select Project >  **Alert** > **Event Configuration** > **Metrics** tab

You can view the complete list of all metrics-based events configured for the project. You can review, edit, or delete existing metrics events, or add new events as needed.

Default Options

- **Event Level:** Indicates the severity of an event and is classified into three levels: Normal (**Info**), Warning (**Warning**), and Critical (**Critical**).
- **Duplicate Agent Alert Suppression:** If two or more agents with the same name run simultaneously, events are not triggered for the selected duration (30 seconds, 1 minute, 3 minutes, 5 minutes, 10 minutes).
- **Search:** Search events by event name.
- **JSON Download/Upload:** Edit event rules directly in the JSON editor, or download them as a file, modify them, and upload them again.
- **+ Add Event:** Add a new metrics event.
- **Resize Columns:** Adjust column widths by dragging column borders.

Table | Metrics Event List Structure

Item	Description
No.	Event sequence number
	Event activation status – Enabled: An event is triggered when conditions are met – Disabled: The event does not run and conditions are not evaluated
	Edit or delete event settings
Event Name	User-defined event name – Up to 255 characters

Item	Description
Rule	Event trigger conditions
Target	Monitoring targets to which the event applies
Trigger Count	The time window and number of occurrences required to trigger an event – An event is triggered when conditions are met the configured number of times within the selected duration
Pause	Temporarily suppresses the same event after an alert is triggered – Available durations: Disabled, 5 min, 10 min, 15 min, 20 min, 30 min, 1 hr, 2 hrs, 3 hrs, 6 hrs, 12 hrs, 1 day – If recovery notifications are enabled, alerts are not triggered for the selected duration after a RECOVERED notification is sent
Recovery Notification	Whether to receive a RECOVERED (normal) notification when Critical or Warning events are resolved – Enabled: Sends a RECOVERED notification when the status changes from in progress to normal and allows viewing active and resolved states in event history – Disabled: Triggers a one-time event each time the threshold is exceeded
Event Recipients	Users or groups that receive notifications for the event (recipient tags)

Event Configuration

Metrics events allow you to define events based on conditions that occur in monitored targets and configure notifications so you can respond efficiently.

Add Event

To add a metrics event, go to **Event Configuration > Metrics** tab, then click the [+ Add Event] button in the upper-right corner.

How to Add a Metrics Event

Step 1. Select Template: Choose the basic structure of the event.

Step 2. Define Event Conditions: Configure the conditions that determine when the event is triggered.

Step 3. Select Event Target: Select the monitoring targets where the event applies.

Step 4. Basic Information and Notification Settings: Configure the event name, message, and recipients.

Select Template

- Depending on how you want to configure the event, select one of the three templates on the left side of the screen.
 - Create New:** Freely configure categories and metrics from scratch. This option is suitable for advanced users who need complex conditions or fine-grained monitoring rules.
 - Quick Setup:** Provides predefined categories, metrics, and default thresholds. You can apply it quickly by adjusting only the threshold values, which makes it suitable for beginners.
 - Advanced Setup:** Add conditions or freely modify settings based on predefined categories and metrics.

ⓘ Metrics events support multiple configuration approaches. The available templates may vary depending on the monitoring platform.

Define Event Conditions

In the **Define Event Conditions** step, configure the metrics that determine when the event is triggered, along with the trigger count, pause settings, recovery notifications, and event active time.

Metric Settings

- Enter a category in the search field, or select a category (metric) from the category list. You must select a category to proceed with metric configuration.

- The category list shows the category name, data collection interval, and key. It displays metrics data collected in the project within the last 3 hours.
2. If the metric you need is not listed, select **Direct Input** and enter the category manually.

❗ **If you are not sure about categories or metrics, use the Metrics Finder.**

When configuring metrics for the first time, the Metrics Finder helps you locate metrics more quickly.

- Search for a metric in the Metrics Finder to view related categories and detailed information.
- Click the [Select] button for a metric in the search results to automatically apply the selected value to the event configuration screen.

The screenshot shows the 'Metrics finder' window with a search bar containing 'cpu'. Below the search bar, it indicates '13 cases/ Total 150 cases'. A table lists various metrics with columns for Category, Name, Unit, Type, Description, Merge, and a Select button. The metrics listed are related to CPU usage and system load.

Category	Name	Unit	Type	Description	Merge	
Application Process Counter	CpuTime	ms	field	CPU usage time	OM: avg TM: avg	Select
JavaOS	ProcessCpuTime	ms	field	Process CPU Time	OM: avg TM: avg	Select
JavaOS	SystemCpuLoad	pct	field	Number of Avg System CpuLoad Percent	OM: avg TM: avg	Select
JavaOS	ProcessCpuLoad	pct	field	Number of Avg Process CpuLoad Percent	OM: avg TM: avg	Select
JavaOS	CpuLoad	pct	field	Number of Avg CpuLoad Percent	OM: avg TM: avg	Select
Application Host Resource	CpuProc	pct	field	CPU utilization in the Proc area	OM: avg TM: avg	Select
Application Host Resource	CpuCores	count	field	Number of CPU cores	OM: avg TM: avg	Select
Application Host Resource	CpuWait	pct	field	CPU utilization taken to wait for disk IOs	OM: avg TM: avg	Select

At the bottom of the table, there is a pagination control showing '1' of 1 page and '50 / page'.

3. Select the event level (**Critical**, **Warning**, Info). You can configure multiple levels for a single metric.

❗ **Event Level Behavior**

Even when multiple conditions are satisfied, only the highest-priority level (**Critical** → **Warning** → Info) triggers an event. For details, see the [Level Action Guide](#).

- Event levels are classified as Critical (**Critical**), Warning (**Warning**), and Info (Info).



- It is recommended to configure multiple levels for the same metric with stepwise thresholds.

4. Enter the field, operator, and value for the selected event level.

a. Click + **Add** to add more conditions.

- **&&** : An event is triggered only when **all conditions are satisfied simultaneously**
- **||** : An event is triggered when **at least one condition is satisfied**

b. Click — to remove the condition.



For condition syntax and supported operators, refer to the [Condition Settings Guide](#).

Trigger Count

Configure how many times a condition must be met before an event is triggered. This is useful for detecting persistent issues only.

1. Select the trigger frequency.

- **Consecutive:** An event is triggered when the condition occurs consecutively for the configured number of times.
- **Recent:** An event is triggered when the condition occurs the configured number of times within the selected time window.
 - Available durations: 5 seconds , 10 seconds , 15 seconds , 30 seconds , 1 minute , 2 minutes , 3 minutes , 4 minutes , 5 minutes , 10 minutes , 30 minutes , 1 hour

2. Enter the trigger count.

- The **Interval** value corresponds to the data collection interval of the selected category. (Interval: 10s)

Pause

To prevent excessive event generation, select a pause duration for event execution. After an event occurs, the same event will not be triggered again during the configured time period.

- Available durations: Disabled , 5 minutes , 10 minutes , 15 minutes , 20 minutes , 30 minutes , 1 hour , 2 hours , 3 hours , 6 hours , 12 hours , 1 day

ⓘ When recovery notifications are enabled

Even if the same event condition is met again, no notification is sent until the configured time has passed since the **RECOVERED** notification was received.

Resolved Notification

When the resolved notification feature is enabled, you can track state changes after an event occurs (in progress → resolved). The event is shown as an ongoing event in the **Event History** menu.

- Resolved notifications apply only to **Critical** and **Warning** levels. Info-level events always operate as one-time events.
- When a **Critical** or **Warning** event is resolved, a notification with the RECOVERED (normal) state is sent.
- Click the toggle button to **enable** or **disable** this feature.
 -  **Enabled:** Events operate based on state. They enter an in-progress state when the threshold is exceeded and transition to a resolved state when the value recovers below the threshold.
 -  **Disabled:** A one-time event is generated each time the threshold is exceeded.

Event Operation Time

Configure tags so that events operate only during specific time periods (such as business hours, non-business hours, or maintenance windows). If no tag is configured, events operate continuously for 24 hours while enabled.

1. Click **+ Add**.
2. In Event Operation Time, click **+ Create New Tag**.

3. In **Create Event Operation Time Tag**, select the **tag name, days, time, and color**, then click the [Apply] button.
4. You can view created tags in the tag list, and apply them by selecting the checkbox.
5. To modify or delete a tag, click the  icon and update or delete it in **Edit Event Operation Time Tag**.

 When a tag is deleted, it is removed from all users to whom it was applied. However, tags that are currently in use by event rules cannot be deleted.

Simulation

The simulation feature lets you pretest configured event conditions using historical data. If the expected number of events is too high or too low, you can adjust the conditions to find optimal threshold values.

- **Rule Validation:** Verify how newly created event rules would behave when applied to historical data.
- **Threshold Tuning:** Adjust thresholds to an appropriate level when events occur too frequently or not at all.

Start Simulation

1. After completing event condition settings, click the [Simulation] button.
2. Review the expected number of events and the visualized results.

Understanding Simulation Results

1. Time-Series Data

A graph displays metric changes over time along with threshold lines for each level, allowing you to intuitively identify when events are triggered.

2. Event Counts by Level

The number of events for the Critical, Warning, and Info levels is shown on the right side of the screen. This helps you understand overall event distribution and frequency.

3. Target-Level Details

Hover over a specific point on the chart to see the exact metric value and monitoring target at that moment. When multiple targets exist, they are distinguished by color. If patterns differ by target, you can use filtering to analyze them individually.

4. Peak Interval Identification

Identify periods where metrics spike sharply to understand when issues occur most frequently or to recognize recurring patterns.

✓ Simulation Tips

- **Simulate Different Time Ranges**

Run simulations for weekdays vs. weekends and peak vs. off-peak hours to set optimal thresholds for each time period.

- **Include Actual Incident Periods**

Include past outage periods in simulations to verify whether current rules accurately detect real incidents.

- **Add Target Filtering**

When many targets are monitored, simulation results can become complex. Use filters to focus on specific targets for clearer insights.

- **Adjust Thresholds Gradually**

Instead of making large changes at once, adjust thresholds incrementally, such as in 10 percent steps, to find optimal values.

Select Event Target

If no event target is specified, events are triggered for all targets in the project, which can result in excessive events. It is therefore recommended to clearly define the targets to which the event applies.

ⓘ If you change the event targets, the number of events may change. Click the **Simulation** button and run it again to check the updated expected event count.

Target Selection

You can specify the targets to which the event applies using either **Select Input** or **Direct Input**.

- **Select Input** **Direct Input**

1. Select or enter a tag, operator, and value.
 - a. If it is difficult to find a tag, you can search for it using the **Metrics Finder**.
2. Click + **Add** to add event targets using conditions (`&&` , `||`).

Enter the tag, operator, and value of the target to monitor directly.

- When you choose direct input, a list of available tags (**Tag Select Input**) is provided below the input field.

```
ex. endsWith(okindName, 'example_name') && container == 'prod.billing'
ex. ${4xxErrorType} == '401'
```

 If you change the input method, any content you have entered will be reset.

Basic Information and Notification Settings

Configure the **event name**, **message**, and **recipients**. Because this information is displayed as is in event history and actual notification messages, it should be clear and easy to identify.

Event Activation

Configure whether the event is triggered when the defined conditions are met.

-  **Enabled:** The event is triggered when the configured conditions are satisfied.
-  **Disabled:** Conditions are not evaluated.

Event Name

Enter the event name to be used as the event title. The configured event name is displayed in the **Event List** and **Event History** menus and can be used as a search keyword.

Message

Enter the event message. The message you enter is used in event history and notifications. Click the  icon to view previously written message history. You can include variables in the message to insert actual values at the time the event occurs.

- **Variable Usage Rules**

- Variable format: `${metric_name}`
- Only metrics within the **same category as the configured category** can be used as variables.
- You can combine multiple variables to create a more detailed message.
- The list of available variables can be found in the **Metrics Query** menu.

- **Variable Usage Example**

	Event Configuration	When Event Occurs
Title	Disk Usage Increase	Disk Usage Increase
Message	Disk = <code>\${disk}%</code>	Disk = 89.9%

Event Recipients

Specify the members who will receive event notifications.

- **Receive All Receive by Tag Selection**

Notifications are sent to all targets (members and channels) in the project that have event reception enabled.

Notifications are sent to project members who have the selected tag.

1. Select **Receive by Tag Selection**.
2. In the **Reception Tag** section, click the [+ Add Tag] or [+] button.
3. Under the **Event Reception Tag** window, click + **Create New Tag**.
4. In the **Create Tag** window, enter a tag name, select a color, and click the [Create Tag] button.
5. You can edit or delete created tags by clicking the  icon in the tag list.

6. In the **Event Reception Tag** window, select the desired tag from the **Tag List** to apply it.

ⓘ Reception Tags

Reception tags are a feature for managing users or channels that receive notifications **at a group level**. When a reception tag is assigned to an event, notifications are delivered only to users associated with that tag. If no reception tag is set, notifications are sent to all users in the project.

Edit/Delete Event

1. Go to **Alert Notifications > Event Settings** and open the **Metrics** tab.
2. In the event list, click the  icon for the event you want to edit or delete.
3. In the **Edit Event Rule** window, update the options and click the [Save] button.
 - a. To delete the selected event, click the [Delete] button in the upper-right corner of the **Edit Event Rule** window.

Event Sharing

You can save metric event settings as a JSON file to share configurations with other users or import settings created by others.

- JSON file name: `event-rules-YYYYMMDD.json`

Export

1. Click the [**JSON** ] button at the top right of the screen.
2. When the JSON editor opens, click the [ Export] button.
 - If you use the export function after searching for events, only the searched list will be downloaded as a JSON file.
3. After the JSON file is downloaded, share it with other users.

Import

1. Click the [] button at the top right of the screen.
2. Select the JSON file downloaded using the **Export** function.
3. When the JSON editor opens, choose either [Add to List] or [Overwrite].

 It is recommended to use this feature between projects of the same product type. You can import event settings from projects of different products, but they may not function correctly.

Edit in JSON Format

1. Click the [**JSON** ] button at the top right of the screen.
2. When the editor opens, modify the content according to the JSON format.
3. After completing the edits, click the [Save] button at the bottom of the screen.

 If the modified content does not conform to the JSON format, an error message will be displayed at the bottom of the screen and the content cannot be saved. The error message may vary depending on the type of formatting issue.

JSON error

JSON Data Structure

```
{
  "version": 2,
  "eventId": "zcef7s7f27rum1",
  "enabled": true,
  "stateful": false,
  "title": "Active Transaction",
  "message": "Active Transaction = ${active_tx_count}",
  "category": "app_counter",
  "alertLabel": [
    "oid"
  ],
  "repeatCount": 1,
  "repeatDuration": 0,
```

```

"silent": 300000,
"receiver": [],
"timeTag": [],
"selectString": "",
"conditions": [
  {
    "level": 20,
    "enabled": true,
    "rule": "active_tx_count > 100"
  }
],
"createTime": 1763632674345,
"lastModifiedTime": 1763632674347,
"lastModifiedUser": "support@whatap.io",
"basic": true,
"metaId": "java004",
"selectCondition": {
  "oid": [
    "-1010758404",
    "-250906941"
  ]
}
}

```

Table | JSON Data Structure

JSON Field	Type	Description	Notes
version	Integer	Version of the event rule	Restricted
eventId	String	Unique identifier of the event rule	
enabled	boolean	Whether the event is enabled	
stateful	boolean	Whether the event is state-based	
title	String	Event rule name defined by the user	
message	String	Event rule message defined by the	

JSON Field	Type	Description	Notes
		user	
category	String	Data category	
alertLabel	List<String>	Default identifier (primary key) value assigned per category for event state management	Restricted
repeatCount	Integer	Number of repetitions	
repeatDuration	Long	Duration for repetition	
silent	Integer	Pause duration	
receiver	List<String>	List of recipient tag keys	
timeTag	List<String>	List of operation-time tag keys	
selectString	String	Target selection	
conditions	List<Object>	List of trigger conditions by level	
◇ level	byte	Level value	Critical : 30, Warning : 20, Info : 10
◇ enabled	boolean	Whether the level is enabled	
◇ rule	String	Trigger condition for the level	
createTime	Long	Initial creation time of the event rule	
lastModifiedTime	Long	Last modification time of the event rule	
lastModifiedUser	String	Account that last modified the event rule	

JSON Field	Type	Description	Notes
basic	boolean	Whether the event was created using Quick Setup	Restricted
metaId	String	Unique identifier of the Quick Setup template	Restricted: This value exists only for events created via Quick Setup (basic=true).
selectCondition	Object	Target selection value for Quick Setup	This value exists only for events created via Quick Setup (basic=true).

 Fields marked as “Restricted” may affect event behavior, so it is recommended not to modify their values.

Appendix

Trigger Conditions and Target Selection Guide

Metric alert trigger conditions and event target selection use the same syntax. However, trigger conditions use the field key as the variable, while target selection uses the tag key as the variable.

Level Behavior Guide

Event levels are categorized into Critical, Warning, and Info.

1. Priority-based Event Triggering

When multiple level conditions are met at the same time, only the event with the highest priority is triggered.

Example

Configuration:

- Warning: CPU > 70%
- Critical: CPU > 90%

Current state: CPU 95%

→ Result: Only the Critical event is triggered (Warning is suppressed)

2. Level Escalation

For events that maintain an in-progress state, when a higher-level condition is met while a lower-level event is already in progress, both levels enter the in-progress state.

Example

Configuration:

- Warning: CPU > 70%
- Critical: CPU > 90%

Scenario:

- 1) CPU 80% → Warning triggered (in progress)
- 2) CPU 95% → Warning (in progress), Critical (in progress)

3. Level De-escalation

For events with an in-progress state, when the level transitions from a higher level to a lower level, the higher-level event is resolved while the lower-level event remains active.

Example

Configuration:

- Warning: CPU > 70%
- Critical: CPU > 90%

Scenario:

- 1) CPU 97% → Critical triggered
- 2) CPU 85% → Critical resolved, Warning remains
- 3) CPU 65% → Warning resolved

4. Execution Flow

- Warning: CPU > 70%

- Critical: CPU > 90%

Metric value: 60% → 75% → 92% → 85% → 60%

State: Info Warning Critical Warning Info
 (triggered) (triggered) (maintained) (resolved)
 Warning
 (maintained)

- ⓘ Events that have an in-progress state remain active as long as their conditions are met and continue until those conditions are cleared.

Condition Configuration Guide

- ⓘ If a metric name starts with a number or contains special characters, you must wrap it in the `${metric_name}` format.

- Brackets `()`, `[]`
- Operators `+`, `-`, `*`, `/`, `%`
- Delimiters `:`, `@`, `#`, `,`, `,`, (space)
- Other special characters `!`, `^`, `&`, `|`, `~`, `\"`, `=`

Examples using `${}` :

```
${cpu(xos)} > 50
${mem[0]} >= 100
${cpu-usage} > 80
${namespace:cpu} > 70
${metric name} > 60
${4xx_error} > 10
```

Usable without `${}` : Metric names that contain only alphabetic characters, underscores (`_`), or dots (`.`)



```
cpu > 80
cpu_usage > 50
CPUUtilization.Average > 0.8
```

1. Comparison Operators

Numeric Comparison

```
Greater than:      cpu > 80
Greater than or equal:  cpu >= 80
Less than:         memory < 1000
Less than or equal:  memory <= 1000
Equal to:          status == 200
Not equal to:       error != 0
```

String Comparison

```
status == 'OK'
region == "us-east-1"
```

2. Arithmetic Operators

- Basic arithmetic operations

```
Addition:      cpu + 10 >= 90
Subtraction:    memory - 100 >= 500
Multiplication: cpu * 2 >= 100
Division:       disk / 1024 >= 100
Modulo:         value % 10 == 0
```

- Controlling precedence with parentheses

```
(cpu + memory) * 2 >= 200
(disk - used) / total >= 0.2
((cpu + memory) / 2) >= 50
```

- Negative values

```
cpu > -100
```

3. Logical Operators

- AND operator (&&)

```
cpu > 80 && memory > 1000  
${cpu(xos)} > 50 && ${mem(xos)} > 60
```

- OR operator (| |)

```
cpu > 90 | | memory > 90  
disk < 10 | | network > 1000
```

- Combined logical operations

```
(cpu > 50 && memory > 50) | | disk < 20  
cpu > 80 && (memory > 1000 | | disk > 500)
```

4. Pattern Matching Operators

- LIKE operator

```
oname like 'prod-*'  
url like '*error*'  
message like 'WARN%'
```

- NOT LIKE operator

```
oname not like 'test-*'  
url not like '*debug*'
```

5. Built-in Functions

- Null check functions

```

isNull(value)      # Checks whether the value is null
isNotNull(value)   # Checks whether the value is not null
nvl(value, 0)      # Returns a default value if null
isEmpty(str)       # Checks whether the string is empty
isNotEmpty(str)    # Checks whether the string is not empty

```

- Aggregate functions

```

sum(cpu, memory, disk)    # Sum
avg(cpu, memory)          # Average
max(cpu, memory, disk)    # Maximum
min(cpu, memory, disk)    # Minimum
count(value1, value2, value3) # Count

```

- Math functions

```

round(cpu, 2)    # Rounds to 2 decimal places

```

- String functions

```

length(str)      # String length
startsWith(str, 'prefix') # Checks prefix
endsWith(str, 'suffix') # Checks suffix
indexOf(str, 'search') # String index
substring(str, 0, 10) # Substring
trim(str)        # Trims whitespace
replace(str, 'old', 'new') # String replacement
hasStr(str, 'search') # Checks string containment

```

- Conditional functions

- **if:** A conditional function that returns the `trueValue` if the condition evaluates to `true`, or the `falseValue` if it evaluates to `false`

Syntax

```

if(condition, trueValue, falseValue)

```

Example

```
if(value >= 90, 'High', 'Medium') == 'High'
```

- **decode:** Compares a value against conditions and returns the result of the matching condition, or a default value if no match is found

Syntax

```
decode(value, condition1, result1, condition2, result2, ..., conditionN, resultN, defaultValue)
```

Example

```
decode(value, 1, 'Low', 2, 'Medium', 3, 'High', 'Unknown') == 'Unknown'
```

- **in:** Returns `true` if the value matches any of the specified candidates; otherwise returns `false`

Syntax

```
in(value, candidate1, candidate2, ..., candidateN)
```

Example

```
in(status, 200, 201, 204) == false
```

! Writing Guidelines

- Recommended
 - Use simple variable names without `${}` (e.g. `cpu`, `memory`)
 - Always use `${}` when special characters are included
 - Use parentheses to clearly separate complex expressions
 - Enter function names exactly as defined (case-sensitive)
- Avoid



- Incomplete expressions (e.g. `cpu > , memory && )`)
- Mismatched parentheses (e.g. `(cpu > 80 && cpu > 70) )`)
- Consecutive or invalid operators (e.g. `cpu >> 80 , cpu >< 80 )`)

Composite Metrics

Home > Select Project >  **Alert** > **Event Configuration** > **Composite metrics** tab

You can create events and send alert notifications using complex rules.

Default Options

- **Search:** Search for specific events by selecting an event name or an event recipient tag
- **JSON Download/Upload:** Edit event rules directly in the JSON editor, or download them as a file, modify them, and upload them again
- **Alert Message Settings:** Select detailed fields for alert messages and edit the content for each field
- **+ Add Event:** Add a new composite metrics event

Permissions

To configure **Composite metrics** events, you must have **Alert Settings** permission. You can grant this permission by clicking the  icon for the member in **Home > Select Project > Management > Integrated Member Management**.

Table | Composite Metrics Event List Layout

Item	Description
Event Title	User-defined event name - Up to 255 bytes
Condition	Event trigger conditions
Notify on Event Recovery	Whether to receive RECOVERED (normal) notifications when Critical or Warning events are resolved – Enabled: Sends a RECOVERED notification when the state changes from in-progress to normal, and allows you to distinguish in-progress and resolved states in event history – Disabled: Triggers a one-time event each time the threshold is exceeded

Item	Description
Event Recipient Tag	Users or groups that receive notifications for the event
	Edit or delete the event
	Event enablement status – Enabled: Event is triggered when conditions are met – Disabled: Event does not run and conditions are not evaluated

Event Configuration

Composite metrics events define event conditions based on **MXQL**, the metric data query language. The **Create from Chart** feature provides a combo box to support MXQL auto-completion.

Add Event

1. Click the **Composite metrics** tab from  **Alert > Event Configuration**.
2. Click the [+ Add Event] button on the right side of the screen.
3. In the **Composite metrics** window, click **Create from Chart**.
 - a. You can use the provided templates.
4. In the **Event Configuration** screen, first query the event data, then configure the event conditions.

Query Event Data

Query event data to build charts. In the **Event Configuration** screen, select either the **Widget** or **Text** option under **Query event data** on the right to configure events.

- Widget Text

Using the options for configuring a time series chart, you can automatically complete the **MXQL** used for event configuration.

1. Select the date range for querying data.
2. Select the **Category** (target) for querying data.
3. Click + **Add Filter** under **Filter** to create event conditions.
 - a. In the filter window, select an expression, tag, and filter value.
 - b. Click + **Add** to add additional conditions.
 - c. To delete a condition, click the – icon next to it.
4. Under **Grouping**, select the grouped metric data. Multiple selections are supported.
5. Under **Time Unit**, set the time interval for grouped data by selecting `seconds` , `minutes` , or `hours` .
6. Under **Field**, select the fields to be used for event trigger conditions. Multiple selections are supported.

 MXQL entered in text mode is not compatible with widget mode.

An editor appears where you can directly modify **MXQL**.

1. Edit **MXQL** directly and click the **Query** button.
2. After selecting a date, you can view the results as a chart or graph.

Notification

Enter the basic information for alert notification settings.

- **Enable Event:** When the toggle is enabled, the event becomes active.
- **Level:** Select the event level (Critical, Warning, Info).
 - **Notify on Event Recovery:** When the toggle is enabled, a notification is sent when the event is resolved and enters the normal (RECOVERED) state.
- **Title:** Enter a title for the event rule.
- **Message:** Enter the content of the alert message delivered when the event is triggered. You can use variables in the message to include actual values at the time the event occurs.

Message example

✔ Variable Usage Rules

- Variable format: `${metricName}`
- Only metrics within the **same category as the configured category** can be used as variables.
- You can combine multiple variables to compose a detailed message.
- The list of available variables can be checked in **Metric Query**.

Event Trigger Conditions

- **Data Query Range:** Select the real-time data query window.
 - Available ranges: 30 seconds , 1 minute , 2 minutes , 5 minutes , 10 minutes , 30 minutes , 1 hour , 2 hours , 3 hours
- **Condition:** Enter the event trigger condition. Only fields included in the event data query can be used.

Additional Information

- **Interval:** The alert condition is evaluated at the selected interval (1 minute , 5 minutes , 10 minutes , 30 minutes , 1 hour , 12 hours).
- **Silence:** After an event is triggered, the same event is suppressed for the selected duration (unset , 1 minute , 5 minutes , 10 minutes , 30 minutes , 1 hour , 12 hours).
- **Event Active Time:** Configure tags to control when the event runs, such as working hours, non-working hours, or maintenance windows. If no tag is set, the event runs continuously for 24 hours while it is enabled.
 1. Click + **Add**.
 2. Under **Event Active Time**, click + **Create New Tag**.
 3. In **Create Event Active Time Tag**, select a **Tag Name**, **Days**, **Time**, and **Color**, then click the [Apply] button.
 4. Created tags appear in the tag list and can be applied by selecting the checkbox.
 5. To edit or delete a tag, click the  icon and modify or remove it in **Edit Event Active Time Tag**.

 When a tag is deleted, it is removed from all users to whom it is applied. However, tags that are

 currently in use by event rules cannot be deleted.

- **Event Recipient Tags:** When recipient tags are configured, notifications are sent to project members who have those tags. If no recipient tag is set, notifications are sent to all users in the project.

1. Click + **Add Tag** or the + icon.
2. Click + **Create New Tag** at the bottom of the **Event Recipient Tags** window.
3. In the **Create Tag** window, enter a tag name, select a color, and click the [Create Tag] button.
4. To edit or delete a created tag, click the  icon in the tag list.
5. Select the desired tag from the **Tag List** in the **Event Recipient Tags** window to apply it.

 When configuring events, selecting event recipient tags allows notifications to be sent to project members and third-party plugins associated with those tags. You can assign tags to project members and third-party plugins separately via **Alert > Notifications**.

Test Event Rules

You can test the configured conditions against historical data to estimate how many events would have been triggered.

1. Select the date for testing the event rules and click the [Confirm] button.
2. Click the [Run] button to check the number of triggered events.

Edit/Delete Event

1. Go to the **Alert > Notifications > Composite metrics** tab.
2. In the composite metrics list, click the  icon of the event you want to edit or delete.
3. In the **Notifications** screen, update the options and click the [Save] button.
 - a. To delete the selected event, click the [Delete] button in the upper right corner of the **Notifications** screen.

Event Sharing

You can save metric event settings as a JSON file to share configurations with other users or import settings created by others.

- JSON file name: `event-rules- YYYYMMDD .json`

Export

1. Click the [**JSON** ] button at the top right of the screen.
2. When the JSON editor opens, click the [ Export] button.
 - If you use the export function after searching for events, only the searched list will be downloaded as a JSON file.
3. After the JSON file is downloaded, share it with other users.

Import

1. Click the [] button at the top right of the screen.
2. Select the JSON file downloaded using the **Export** function.
3. When the JSON editor opens, choose either [Add to List] or [Overwrite].

 It is recommended to use this feature between projects of the same product type. You can import event settings from projects of different products, but they may not function correctly.

Edit in JSON Format

1. Click the [**JSON** ] button at the top right of the screen.
2. When the editor opens, modify the content according to the JSON format.
3. After completing the edits, click the [Save] button at the bottom of the screen.

 If the modified content does not conform to the JSON format, an error message will be displayed at the

! bottom of the screen and the content cannot be saved. The error message may vary depending on the type of formatting issue.

JSON error

Alert Message Settings

You can select detailed fields for the messages to be received and edit the content for each field.

Appendix

MXQL Syntax Guide

Composite metrics operates based on MXQL queries. For detailed syntax, refer to the [MXQL Syntax Guide](#).

Hitmap Patterns

Home > Select Project >  **Alert** > **Event Configuration** > **AJAX Hitmap pattern** tab

AJAX Hitmap Pattern automatically detects time-based anomalies by analyzing Hitmap data that visually represents the distribution and variation of AJAX requests in a web application.

Without manually configuring complex conditions or thresholds, the system analyzes AJAX request patterns collected in real time and automatically sends alert notifications when abnormal patterns that deviate from the normal range are detected.

AJAX Hitmap pattern alerts are useful in the following situations.

- When AJAX requests suddenly increase or become concentrated during specific time periods compared to normal behavior
- When the response time or frequency of AJAX requests changes in an unusual way
- When you need to detect pattern-based anomalies that are difficult to capture with fixed conditions

Event Configuration

Configure the options for the event patterns to be detected.

Activation

When the toggle is turned on, an alert notification is sent when the corresponding pattern is detected.

-  **Enabled:** An event is triggered when conditions are met.
-  **Disabled:** No event is triggered, and conditions are not evaluated.

Silence

After an event is triggered, the event is suppressed for the selected duration and is not recorded in **Event History**.

Event Recipient Tags

When recipient tags are configured, notifications are sent to project members who have those tags. If no recipient tag is set, notifications are sent to all users in the project.

1. Click + **Add Tag** or the + icon.
2. Click + **Create New Tag** at the bottom of the **Event Recipient Tags** window.
3. In the **Create Tag** window, enter a tag name, select a color, and click the [Create Tag] button.
4. To edit or delete a created tag, click the  icon in the tag list.
5. Select the desired tag from the **Tag List** in the **Event Recipient Tags** window to apply it.

Event Active Time

Configure tags to control when events run, such as working hours, non-working hours, or maintenance windows. If no tag is set, events run continuously for 24 hours while they are enabled.

1. Click + **Add**.
2. Under **Event Active Time**, click + **Create New Tag**.
3. In **Create Event Active Time Tag**, select a **Tag Name**, **Days**, **Time**, and **Color**, then click the [Apply] button.
4. Created tags appear in the tag list and can be applied by selecting the checkbox.
5. To edit or delete a tag, click the  icon and modify or remove it in **Edit Event Active Time Tag**.

 When a tag is deleted, it is removed from all users to whom it is applied. However, tags that are currently in use by event rules cannot be deleted.

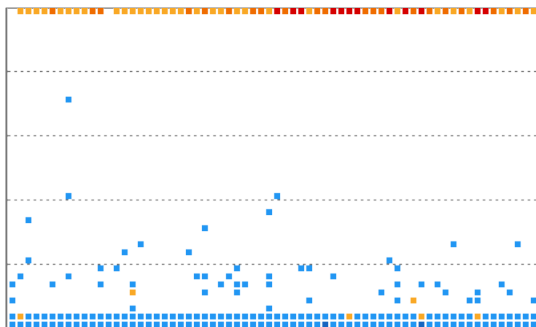
Hitmap Pattern Types

When you turn on the toggle for each pattern, an event is triggered when the corresponding pattern is detected.

Hitmap Horizontal Line Pattern

This pattern appears when multiple transactions complete within a similar response time range. Requests cluster around a specific latency band, which indicates increasing service delay or unstable responses from external resources.

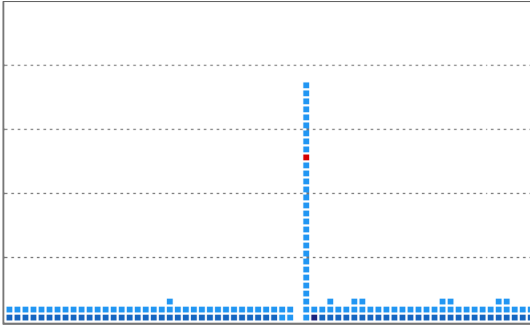
- Timeouts or delays occurring in external API, DB, or HTTPS calls
- Requests waiting for a certain duration before being processed all at once
- Repeated latency in shared I/O or network paths



Hitmap Vertical Line Pattern

In this pattern, transactions start at different times but finish simultaneously. Multiple requests are processed at the same moment, which indicates a bottleneck in a shared resource.

- Increased wait time due to shared resources such as DB locks or file locks
- Concurrent processing delays in queues or thread pools
- Temporary request suspension followed by simultaneous resumption due to system pauses such as GC pauses

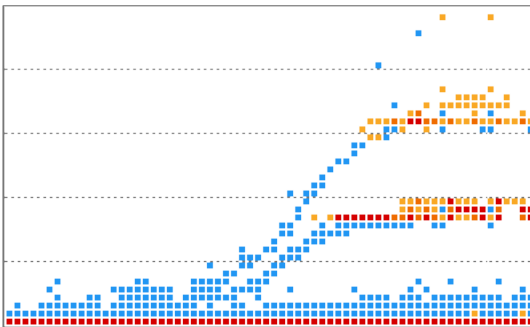


- ✓ A Hitmap vertical line pattern is a signal that lock contention or bottleneck resources should be inspected.

Hitmap Flying Pattern

This pattern appears when response times rise and fall in waves at regular intervals due to shortages in shared resources such as specific system components or logs.

- Shortages in shared resources such as GC, connection pools, or file handles
- Periodic latency factors such as log flushing or storage I/O
- Batch jobs or background jobs that periodically consume resources

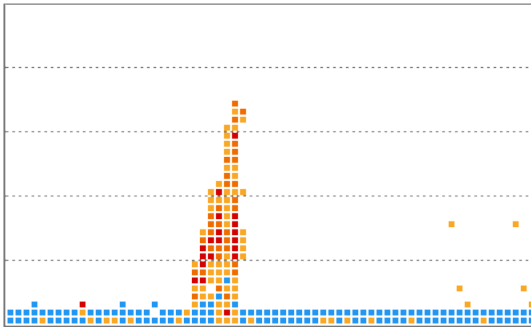


- ✓ A Hitmap flying pattern strongly indicates periodic bottlenecks or resource shortages.

Hitmap Overload Pattern

This pattern occurs when transactions become abnormally concentrated within a specific response time range. It appears when requests fail to distribute normally and instead pile up in a narrow band due to temporary load spikes or early signs of failure.

- Transactions accumulate in a queue due to temporary failures or delays and are then processed in bulk
- Backend service response latency
- Insufficient processing capacity caused by traffic concentration during specific time periods

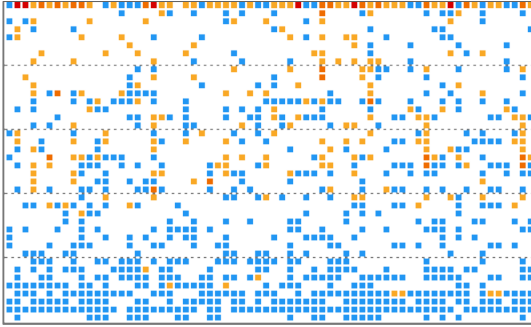


✓ An overload pattern indicates temporary service quality degradation or load balancing failure.

Hitmap Flood Pattern

This pattern indicates an overall increase in response times caused by excessive requests or load. It reflects a broad performance degradation across the service.

- Sudden spikes in user traffic or automated traffic
- System resource shortages such as CPU, memory, or GC pressure
- Overall response quality degradation due to backend bottlenecks



- ✓ A flood pattern requires immediate investigation.

Event Notification Settings

Home > Select Project >  **Alert** > **Notifications**

Event Notification Settings per User

 Information about other members in **Event notification settings per-user** can only be viewed by members with the **Member Management** permission. For all other members, required information is masked.

- Mobile device management for bulk event notification receipt settings and access control can be managed in **Account Information**.
- For detailed information about member permissions, refer to the [following document](#).

Notification Methods

Alerts are available via [Email](#), [SMS](#), [WhatsApp](#), and [Mobile](#).

In the **Event notification settings per-user** section, select the checkbox for the desired notification method to receive alerts. If you clear the checkbox for a notification method, alerts will not be sent through that channel.

 All users except the project super administrator can change only their own notification settings.

Email Alerts

Email alerts are sent to the email address provided during sign-up.

SMS Alerts

You can receive alerts via SMS.

ⓘ The SMS alert feature is available only for paid projects.

1. Click your **profile icon** at the top right of the screen, then go to **Manage Accounts**.
2. In **Account Information**, click the [General Mobile Phone] button in the **User Phone Number** section.
3. Enter the **phone number** that will receive the verification code.
 - Only **Korean mobile phone numbers** can be registered to receive SMS alerts.
4. Click the [Save] button.

✔ Change or delete the registered phone number

- To change the phone number, click the [Change] button. SMS verification is required.
- To delete the phone number, click the [Delete] button. After deletion, you will no longer receive alerts via SMS.

WhatsApp Alerts

You can receive alerts via WhatsApp.

ⓘ The WhatsApp alert feature is available only for paid projects.

1. Click your **profile icon** at the top right of the screen and go to **Manage Accounts**.
2. In **Account Information**, click the [WhatsApp] button in the **User Phone Number** section.
3. Enter the **phone number** that will receive the verification code.
4. Click the [Send Verification Code] button.
5. Enter the 6-digit verification code sent to your WhatsApp application.
 - If you do not receive the verification code within 10 minutes, click the [Resend Verification Code] button.
6. Click the [Verify] button.

Mobile Alerts

Users who need to receive mobile alerts must install the WhaTap application on their mobile device and log in.

- Download the WhaTap application – [Android](#)
- Download the WhaTap application – [iOS](#)

Repeated Alerts (Escalation) Settings

If the alert condition is not resolved after an alert is triggered, you can configure repeated alert intervals based on the time of the initial alert. This setting applies only to **Critical** level alerts.

For example, to repeat alerts immediately (0 minutes), 1 hour later, and 1 day later after the initial alert, enter `0,1H,1D` in the **Repeated Alerts (Escalation)** column.

1. Go to **Alert > Notifications** and navigate to the **Event notification settings per-user** section.
2. Configure the settings in the **Repeated Alerts (Escalation)** field.
 - **M**: minutes, **H**: hours, **D**: days. If the unit is omitted, the value is interpreted as minutes.
 - Enter a number or a number with a unit (**M**, **H**, **D**). If the input format is invalid, an error message is displayed.
3. Click the [Save] button. The settings will not be applied unless they are saved.

Event Reception Tags

When configuring events, you can select **event reception tags** to send alerts to project members and 3rd-party plugins associated with those tags. If no event reception tag is set, alert notifications are sent to **all project members**.

Add / Edit / Delete Event Reception Tags

 Event reception tags that are **currently applied to an event** cannot be deleted.

1. Go to **Alert > Notifications**.

2. In the user list under the **Event notification settings per-user** section, click + **Add Tag** or +.
3. In the **Event reception tag** window, click + **Create New Tag**.
4. In the **Create Tag** dialog, enter a tag name, select a color, and click the [Create Tag] button.
5. To edit or delete a created tag, click the  icon in the tag list.
6. Select the desired tag from the **Tag List** in the **Event reception tag** window to apply it.

Remove Event Reception Tags

1. Go to **Alert > Notifications** and open the **Event notification settings per-user** section.
2. Click + in the **Event reception tag** field for the user whose event reception tag you want to remove.
3. In the **Event reception tag** window, click the **X** on the tag you want to remove. The tag will be removed.

Receive Alerts Without Reception Tags

When configuring events, you can select event reception tags to send alerts to project members and 3rd-party plugins associated with those tags.

- To receive alerts **without event reception tags**, enable the **Receive alerts without reception tags** option.
- If you want to receive **only alerts that have event reception tags**, disable this option.

 To stop receiving all alerts, disable this option and make sure that no **Event reception tag** is selected.

Selecting the event reception mode

In the **Event notification settings per-user** section, you can select the event alert reception scope per member or third-party channel from the following three modes. The reception mode is selected in the **Event reception tags** column of each member or channel row.

Mode	Description
Receive all alerts	Receive all alerts regardless of the event reception tags
Receive alerts for selected tags	Receive only alerts that have the selected event reception tags. Select at least

Mode	Description
only	one tag
Receive no alerts	Receive no event alerts. The selected tags are preserved without being deleted

- In the **Receive alerts for selected tags only** mode, select at least one tag to receive alerts in the tag selection area.
- To also receive alerts that have no event reception tag specified, select the **Unclassified** virtual tag in the tag selection area. The **Unclassified** tag can be selected or deselected like a normal tag, but the tag itself cannot be edited or deleted.

ⓘ In the **Receive no alerts** mode, the corresponding member or channel receives no event alerts at all. The selected tags are preserved without being deleted, so if you switch back to the **Receive alerts for selected tags only** mode, you can use the previously selected tag settings as is.

Select Notification Language

You can choose the language in which you receive alert notifications.

1. Go to **Alert > Notifications**.
2. In the user list under the **Event notification settings per-user** section, click the ► button at the far left of the user whose language you want to configure.
3. Select the **Notification Language**.

Day and Time-based Notification Settings

You can configure whether a user receives notifications by day of the week and time of day.

1. Go to **Alert > Notifications**.
2. In the user list under the **Event notification settings per-user** section, click the ► button at the far left of the user whose day and time settings you want to configure.

3. Select the days and/or enter the time ranges during which you want to receive alerts.

- Settings can be configured separately for each notification method.

3rd-party Plugins

You can receive alert notifications through external applications.

- **Supported applications:** Slack, Telegram, Teams, Teams Workflows, Jandi, AlertNow, Webhook, Webhook JSON, Opsgenie, PagerDuty, LINE, ilert

ⓘ Internal messengers that are not included in WhaTap Labs' supported scope can be integrated via standard **Webhook** or **Webhook JSON**.

Table | 3rd-party Plugin List

Item	Description
Plugin Name	Name of the plugin
Auth Key	Authentication key
Auth Value	Authentication value
Receive Level	Notification level: Warning, All, or Critical
Repeated Alerts (Escalation)	Re-notification interval from the initial alert time if the issue is not resolved - Applies only to Critical alerts
Smart Message (AI)	AI-powered notifications available when adding the Telegram plugin - Supported languages: Korean, English, Japanese
Event Reception Tag	Whether to receive alerts for specific members or teams in the project
Delete	Delete the 3rd-party plugin

Add a 3rd-party Plugin

1. Go to **Alert > Notifications**.
2. In the **3rd-party Plugins** section, click **Add**.
3. In the **Add 3rd-party Plugin** window, select the desired service tab.
4. Follow the instructions for the selected service (see [How to add 3rd-party plugin notifications](#)).
5. Click **Test Receive** to verify the configuration.
6. When finished, click **Add** or **Register**.

How to Add 3rd-party Plugin Notifications

Slack

- ⓘ In the **Add 3rd-party Plugin** window, click the [**Add To Slack**] button on the **Slack** tab to register **WhaTapNotiBot** to your Slack team and channel.

Telegram

- ⓘ
1. In **Telegram**, search for **@WhaTapApmNotiBot**.
 2. Select the bot and click the [**Start**] button.
 3. In the chat window, enter the `/license` command followed by the access key of the project for which you want to receive alerts.

Telegram

ⓘ Available commands

- `/help` : View the list of commands
- `/license` : Enter the access key of the project to receive alerts

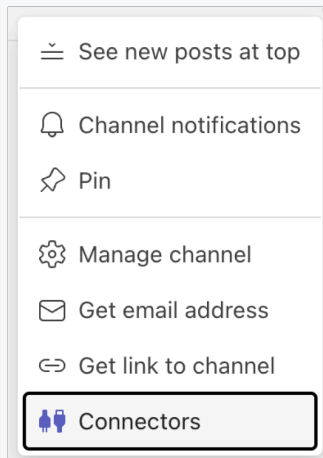


- `/remove` : Enter the project code for which you want to stop receiving alerts
- `/list` : View the list of project codes currently receiving alerts

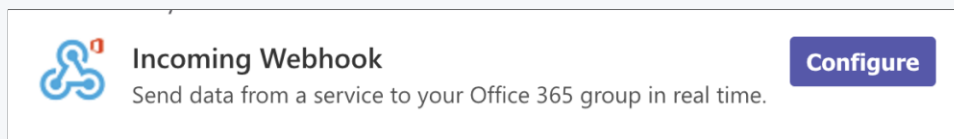
Teams



1. Go to the **connector** of the channel where you want to receive alerts.



2. Add a **Webhook** and click the [Configure] button.



3. Enter a webhook name and click the [Create] button.



Incoming Webhook

[Send feedback](#)

The Incoming Webhook connector enables external services to notify you about activities that you want to track. To use this connector, you'll need to create certain settings on the other service, which needs to support a webhook that's compatible with the [Office 365 connector format](#).

Fields marked with * are mandatory

To set up an Incoming Webhook, provide a name and select Create. *

Customize the image to associate with the data from this Incoming Webhook.

Upload Image



Default Image

Create

Cancel

4. Copy the generated **Webhook URL**.

Copy the URL below to save it to the clipboard, then select Save. You'll need this URL when you go to the service that you want to send data to your group.

<https://whatap.webhook.office.com/webhook>



Url is up-to-date.

Done

Remove

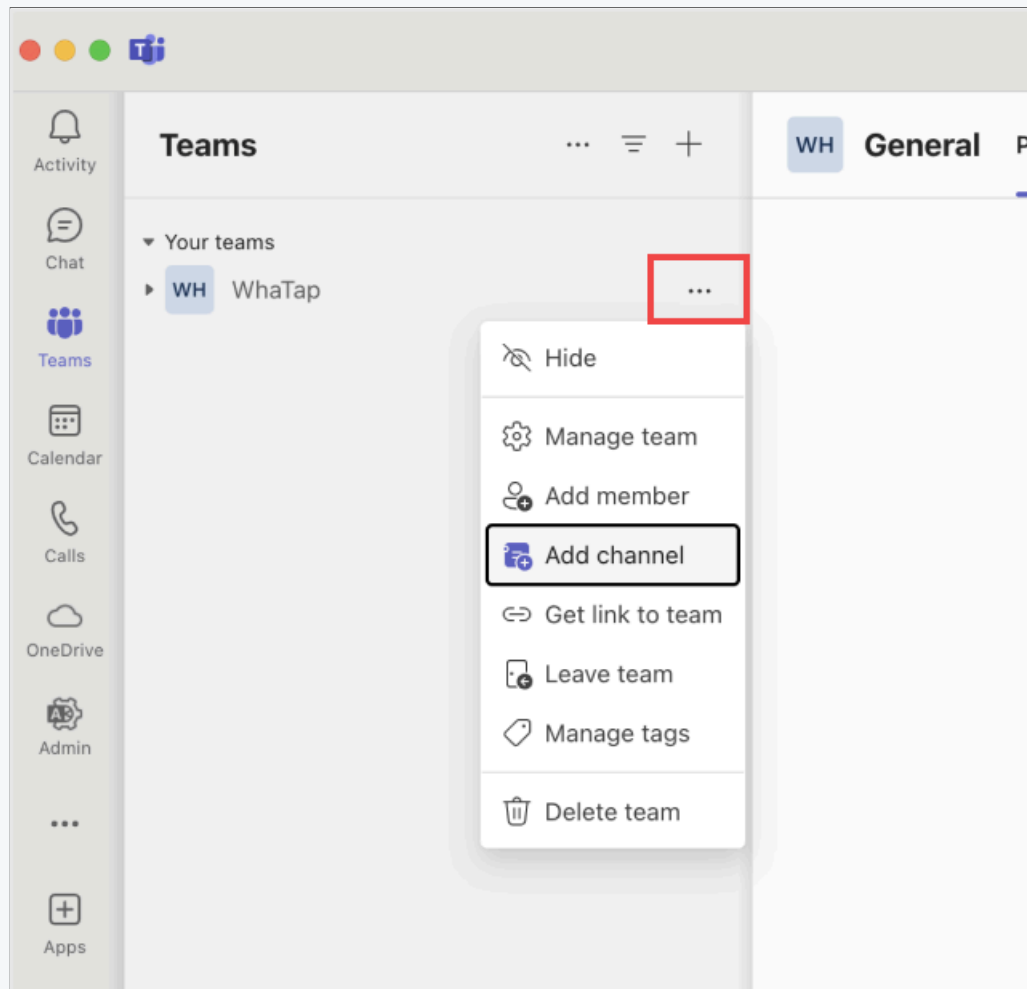
5. Return to the **Add a third-party plug-in** window in the WhaTap Monitoring service, and enter values for **Channel Name** and **Webhook URL** at the bottom of the **Teams** tab.
6. Click the [**Registration**] button.

Teams Workflows



- ⓘ To use workflows, you need an account that can access **Microsoft Teams** and a **Business plan** subscription.

1. In the **Teams** menu, click **More actions (...)** > **Add channel**.



2. In the **Create a channel** window, enter the **Channel name** and **Description**. For **Channel type**, select **Standard**, then click the [Create] button.



Create a channel

Channel name *

WhaTap Event Channel

Description

Help others find the right channel by providing a description

Choose a channel type * ⓘ

Select

Standard
Everyone on the team has access.

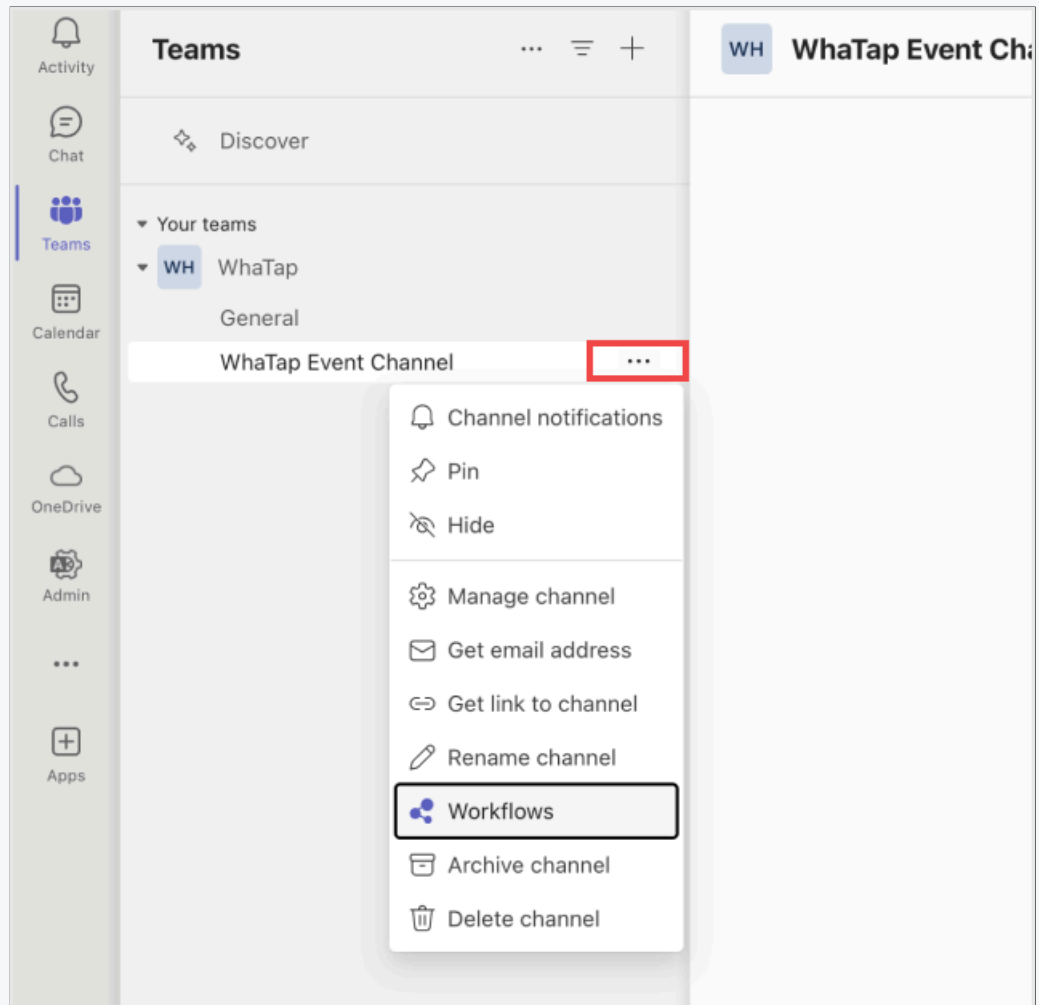
Shared
People or teams in your org or outside your org can have access.

Private
Specific people on the team have access.

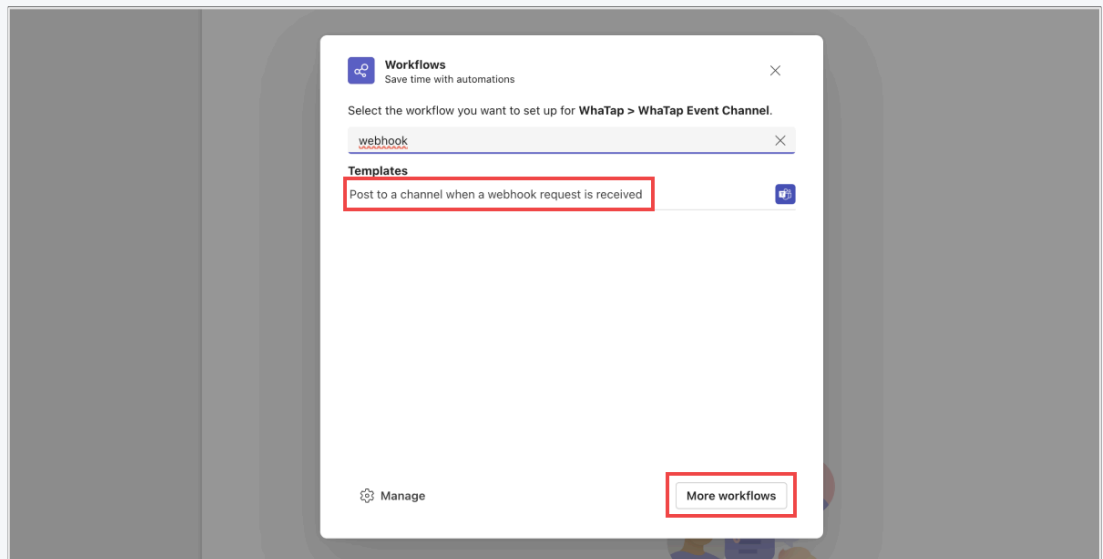
Cancel Create

ⓘ If you select **Shared** as the channel type, workflows cannot be used. If you select **Private**, workflows can be added but notifications cannot be received.

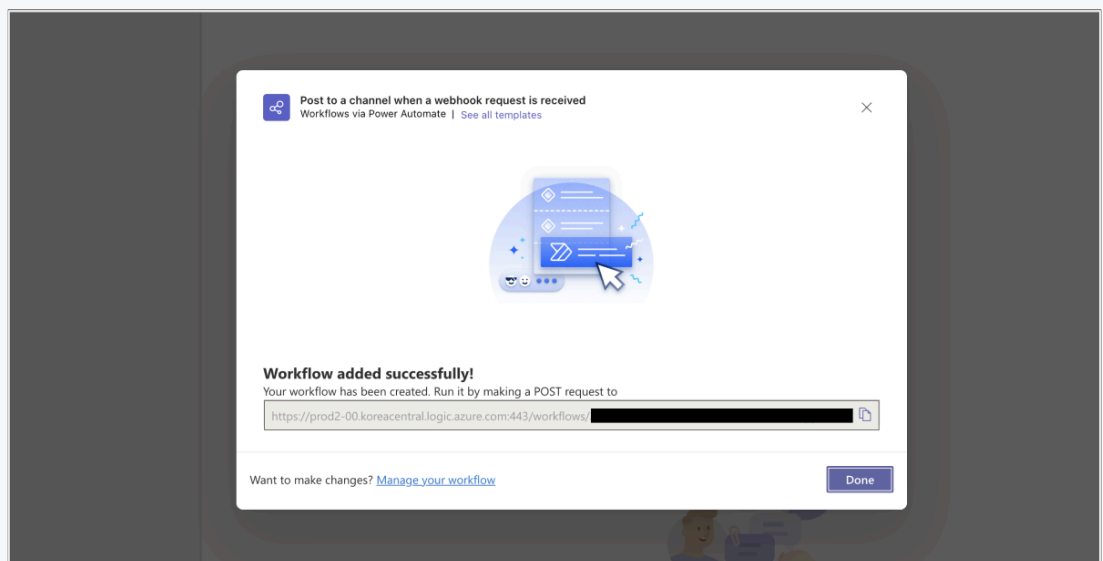
3. Click **More actions (...)** of the created channel, then select **Workflows**.



4. In the **Workflows** dialog, select **Post to a channel when a webhook request is received** from the workflow list, then click the [Add workflow] button.



5. Check the **Name** and **Connection** status, then click the [Next] button.
6. Review the **Details**, then click the [Add workflow] button.
7. The workflow has been added. Copy the generated URL.



8. Return to the **Add a third-party plug-in** window in the WhaTap Monitoring service, and enter values for **Channel Name** and **Webhook URL** at the bottom of the **Teams Workflows** tab.



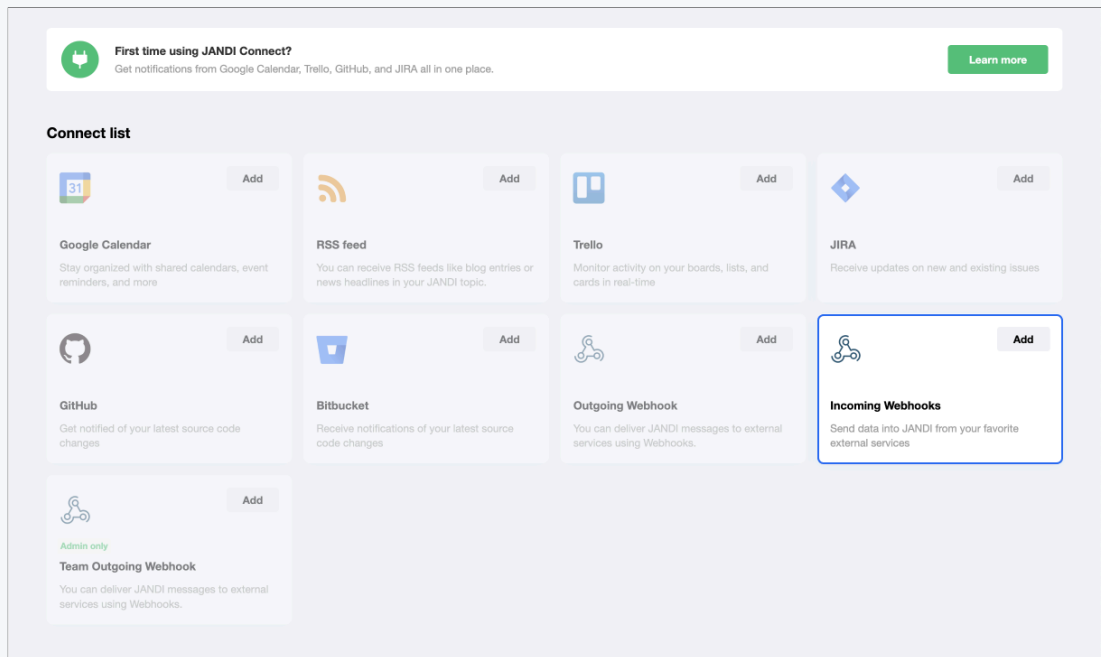
9. Click the [**Registration**] button.



Jandi



1. Go to **Jandi Connect settings** in the topic or chat where you want to receive notifications.
2. In the **Connect list**, find **Webhook Receiver (Incoming Webhook)** and click the [**Connect**] button.



3. Click the [**Add Webhook Receiver**] button.
4. Copy the Webhook URL from the **Register Webhook URL** section.



Connect settings

Profile settings

Change the name or profile image that appears for messages from this integration

 Incoming Webhooks

Topic / JANDI selection

Please select a topic or DM to receive integration messages. You change it after saving it.

JANDI

Webhook URL Registration

Webhook URL

Copy the generated Webhook URL and register with the service.

 <https://wh.jandi.com/connect-api/webhook/31257583/d3...>

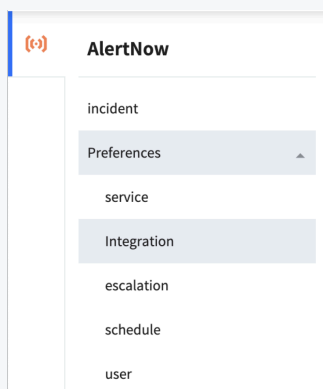
Copy

- Return to the **Add a third-party plug-in** window in the WhaTap Monitoring service, and enter values for **Channel Name** and **Webhook URL** at the bottom of the **Jandi** tab.
- Click the [**Registration**] button.

AlertNow



- Access the **AlertNow** service, then go to **Settings > Integration**.



- On the **Integration** screen, click the [Create Integration] button.
- From the list of integration types, select **WhaTap**.
- On the **Create Integration** screen, configure the **Name** and **Naming Rules**.

5. Click the [Confirm] button to create the integration.
6. Check the list of created integrations and select the newly created integration.

Integration name	Integration type	Integration key	service name
whatap-test	WhaTap		whatap-test

7. On the details screen, click the copy icon on the right side of the URL to copy it.

Basic Information	
Integration type	WhaTap
Integration key	fda527a5140
URL	https://alertnowitgr.opsnow.com/integration/whatap/v1/
constructor	Lisa

8. Return to the **Add a third-party plug-in** window in the WhaTap Monitoring service, and enter values for **Channel Name** and **Webhook URL** at the bottom of the **AlertNow** tab.
9. Click the [**Registration**] button.

Webhook



1. Register the Webhook that will receive notifications. The alert `title` and `text` are sent to the registered URL via the **POST** method.

```
{
  "title": "Alert Title",
  "text": "Sample warning message"
}
```

2. Enter values for **Webhook Name** and **Webhook URL**.
3. Click the [**Registration**] button.

Webhook JSON



1. Register the Webhook that will receive notifications. Alert data is sent to the registered URL in **JSON format** via the **POST** method. For substitution variables that can be included in the JSON payload, see **Available Variables**.
 - `${something}` represents a substitution value provided by WhaTap. Refer to **Available Variables** for the full list and descriptions.
 - The event status (`status`) is categorized as **event triggered** (`on`) or **event recovered** (`off`).
 - The `level` is categorized as `None` , `Info` , `Warning` , or `Critical` .

```
{
  "pcode": "${pcode}",
  "projectName": "${projectName}",
  "time": "${time}",
  "oid": "${oid}",
  "oname": "${oname}",
  "title": "${title}",
  "message": "${message}",
  "level": "${level}",
  "status": "${status}",
}
```



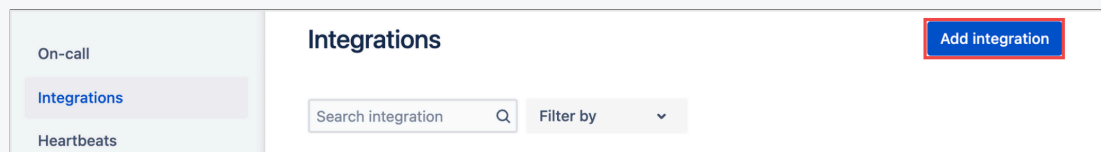
```
"metricName": "${metricName}",
"metricValue": "${metricValue}",
"metricThreshold": "${metricThreshold}",
"uuid": "${uuid}",
"okind": "${okind}",
"onode": "${onode}"
}
```

2. Enter values for the **Webhook Name** and **Webhook URL** fields.
3. To add a header value, click + **Add** in **Add user header value**, then enter the key and value.
4. Click the [**Registration**] button.

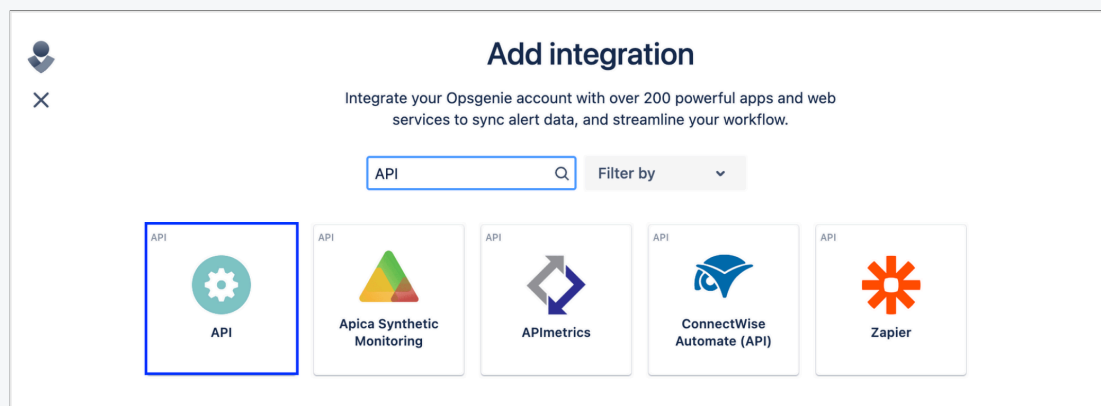
Opsgenie



1. In the **Teams** menu, select the team that will receive notifications.
2. Go to the **Integration** menu and click the [Add Integration] button.



3. In the integration list, search for and select **API**.





4. Set the API name, assign a team, configure the required permissions, and then click the [Save Integration] button.

5. Return to the **Add a third-party plug-in** window in the WhaTap monitoring service, and enter values for **Name** and **API Key** at the bottom of the **Opsgenie** tab.
6. Click the [**Registration**] button.

PagerDuty



1. Sign up for a [PagerDuty](#) account and log in.
2. In PagerDuty, go to **Services > Service Directory**, then click the [+ New Service] button.

PagerDuty

3. Enter the service information (**Name**) and click the [Next] button.

PagerDuty

4. Configure the **Escalation Policy**, then click the [Next] button.



PagerDuty

5. Configure **Alert Grouping** and **Transient Alerts**, then click the [Next] button.

PagerDuty

6. In **Integrations**, select **Event API V2**, then click the [Create Service] button.

PagerDuty

7. On the created service page, copy the **Integration Name** and **Integration Key** values.

PagerDuty

8. Return to the **Add a third-party plug-in** screen in the WhaTap monitoring service and enter the **Integration Name** and **Integration Key** values.
9. Click the [Register] button.

ⓘ If you do not receive messages or the plugin is not registered, check whether the **Integration Name** or **Integration Key** values are correct.

LINE



1. Install the LINE app.
2. Scan the QR code below to add the official **WhaTapLabs** channel as a friend.

Add LINE 3rd-party plugin

3. Use the following commands to link your project with LINE and configure notifications.

Commands

```
/help # Show available commands
/list # View the list of linked projects
/license [ACCESS_KEY] # Add a new project linkage
/remove [PROJECT_CODE] # Remove a linked project
/language [ko|ja|en] # Set notification language (Korean/Japanese/English)
```




- You can find the project access key under **Management > Project Management**.
- Using the `/language` command in LINE applies **only to the current LINE chat room**. It does not change the notification language for other 3rd-party plugins or projects.

Examples

```
/license XXXXXXXXXX # Link a project
/language ko # Set language to Korean
/list # Check linked projects
```



- **Project access key:** Available under **Management > Project Management** in the WhaTap project.
- **Language settings:** The `/language` command applies only within the LINE chat room and does not affect other notification channels.

ilert



1. In the ilert console, click [+ Create alert source] at the top right of the **Alert sources** page.
2. In the **Integration type** step, enter `WhaTap` in the **Filter integrations** search field at the bottom to search.
3. From the search results, select the **WhaTap** tile and click the [Next] button.
4. Enter the alert source name and assign a team, then click the [Next] button.
 - You can create a new escalation policy or select an existing one.
5. Select the alert grouping method and click the [Continue setup] button.
 - The initially selected **Do not group alerts** option can be changed later.
6. On the next screen, you can configure additional settings such as **alert templates and priorities**. For now, click the [Finish setup] button to complete the setup.
7. When the setup is complete, an **API key** or **Webhook URL** will be generated on the completion screen.



8. Copy the generated **Webhook URL** from the **Webhook JSON** tab and paste it into the **Webhook URL** input field below.
9. Click the [Register] button.

Preventing Alert Storms

If a large number of alerts occur within a short period, alert notifications are temporarily paused for a specified duration. For example, if 20 events occur within 5 minutes, alert notifications will be suspended for the next 5 minutes. Once the configured **Pause Duration** has elapsed, the alert-storm prevention feature is automatically released.

1. Go to **Alert > Notifications > the Avoids a large number of alerts** section.
2. Click the toggle button to **Enable** or **Disable** the feature.
 -  **Enabled:** Temporarily pause alerts when an alert storm is detected
 -  **Disabled:** Resume alert notifications
3. Configure the **Detection Time**, **Detection Count**, and **Pause Duration**.
 - If the number of events exceeds the **Detection Count** within the **Detection Time**, alert notifications are paused for the configured **Pause Duration**.



When alerts are blocked due to an alert storm, a notification banner is displayed. To immediately release the alert-storm block, click the **Stop Alert Storm Prevention** button in the banner.

Event Notification Format

Common Alert Template

The common template can be applied in most cases. It allows you to receive alerts in a consistent format across various environments.

- Product Type: Application, Database, Kubernetes
- Event Type: Application Alerts, Database Alerts, Metrics Alerts
- Event Channel: SMS, Mobile, 3rd-party plugins, Plugins

Event Title

Event title format

```
[Level][Platform][ProjectName][ApplicationName][EventTitle]
```

Event title example

```
[Info][JAVA][Application Project][TC-0-1-8081][CRITICAL_HIGH_MEMORY]
```

❗ If ApplicationName is not set, it is omitted from the event title.

- Platform is displayed as one of the following.
 - JAVA
 - NODEJS
 - PYTHON
 - PHP
 - DOTNET
 - GO
 - POSTGRESQL



- ORACLE
- MYSQL
- MSSQL
- BSM_JAVA
- CLOUDWATCH
- TIBERO
- KUBERNETES
- KUBE_NS
- URLCHECK
- URLCHECK_ADMIN
- CUBRID
- ALTIBASE
- CLUSTER
- REDIS
- MONGODB
- VR
- RUM

Event Message

This section describes the information that can be included in the event message. If `optional` is set to `false`, the item is always included in the message. If `optional` is set to `true`, the item is displayed only when the data is available.

Event message example

```
Project Name : Application Project
Project Code : 3
Agent Name : TC-0-1-8081
Message : RECOVERED: Memory is too high. less than 10%
```

Event Time : 2022-04-12 18:53:24 +0900
 Event OFF Time : 2022-04-12 18:53:24 +0900
 Alert Type : APPLICATION_MEMORY
 Metric Name : memory
 Metric Value : 20
 Metric Threshold : 10
 Stateful : true

Table | Event Message Components

En	Ko	Supported Alert Types	Description
Project Name	Project name	All	-
Project Code	Project code	All	-
Agent Name	Agent name	All (optional)	oname
Message	Event message	All	-
Alert Type	Alert type	All	See the AlertType table below
Event Time	Event occurrence time	All	In 2022-04-13 10:40:49 +0900, +0900 indicates the GMT offset
Event Off Time	Event recovery time	All (optional)	In 2022-04-13 10:40:49 +0900, +0900 indicates the GMT offset
Metric Name	Metric name	All (optional)	Name of the metric used to evaluate the event condition
Metric Value	Metric value	All (optional)	Included when the metric value exceeds the threshold
Metric Threshold	Metric threshold	All	Threshold value used to determine the event

En	Ko	Supported Alert Types	Description
		(optional)	condition
Stateful	Recovered-event notification	All (optional)	<code>true</code> if recovered-event notification is enabled; otherwise <code>false</code>
Event Rule	Event trigger condition	Metrics alerts	-
Target Filter	Event target filter	Metrics alerts	Evaluates event conditions only for metrics collected from specific targets
Repeat Count	Event repeat count	Metrics alerts	The event is triggered when the condition is met the specified number of times within the repeat duration
Repeat Duration	Event repeat duration	Metrics alerts	Time window used to evaluate the repeat count
Receiver	Receiver	Metrics alerts	-
Query	MXQL query	Composite metrics alerts	-
Rule	Event trigger condition	Composite metrics alerts	-
Query Period	Query period	Composite metrics alerts	-
Query Interval	Query interval	Composite	-

En	Ko	Supported Alert Types	Description
		metrics alerts	
Silent Time	Silent time	Composite metrics alerts	-
Query	URL	Exception alerts	URL of the request that caused the exception
TXID	Transaction ID	Exception alerts	-
Class	Error class name	Exception alerts	-
Log Message	Log message	Server – file log alerts	-
Log File	Log file path	Server – file log alerts	-
IP	IP address	All server alerts	-
CPU	CPU usage	All server alerts	Snapshot at the time the event occurred
CPU_load1	CPU load (1 min)	All server alerts	Snapshot at the time the event occurred
CPU_loadPerCore	CPU load per core	All server	Snapshot at the time the event occurred

En	Ko	Supported Alert Types	Description
		alerts	
Memory	Memory usage	All server alerts	Snapshot at the time the event occurred
Swap	Swap usage	All server alerts	Snapshot at the time the event occurred
Disk Name Used Percent Free Size IO Percent	Disk performance	All server alerts	Snapshot at the time the event occurred
Name Bps Pps	Network traffic performance	All server alerts	Snapshot at the time the event occurred
Message Time Name	Acknowledgement information	All server alerts	-

ⓘ This section displays as much information as possible that can be provided for the event.

- **AlertType** is displayed as one of the following values.

AlertType	Description
APPLICATION_CPU	Application CPU alert
APPLICATION_MEMORY	Application memory alert
APPLICATION_DISK	Application disk alert
APPLICATION_ACTIVE_TRANSACTION	Application active transaction alert
APPLICATION_ERROR_TRANSACTION	Application error transaction alert



AlertType	Description
APPLICATION_SLOW_TRANSACTION	Application slow transaction (response time) alert
METRICS	Metrics alert
COMPOSITE_METRICS	Composite metrics alert
ANOMALY	Anomaly detection alert
LOG_REALTIME	Real-time log alert
COMPOSITE_LOG	Composite log alert
SERVER_REBOOT	Server reboot alert
SERVER_NO_DATA	Server no-data alert
SERVER_PORT	Server port alert
SERVER_NETWORK_IOPS	Server network IOPS alert
SERVER_NETWORK_BPS	Server network BPS alert
SERVER_DISK_IO	Server disk I/O alert
SERVER_DISK_QUOTA	Server disk usage alert
SERVER_DISK_INODE	Server inode alert
SERVER_CPU	Server CPU alert
SERVER_MEMORY	Server memory alert
SERVER_CPU_STEAL	Server CPU steal alert



AlertType	Description
SERVER_MEMORY_SWAP	Server swap memory alert
SERVER_LOG_FILE	Server log file alert
SERVER_WINDOW_EVENT	Server Windows event alert
SERVER_OFF	Server alert OFF notification
SERVER_ACKNOWLEDGE	Server acknowledge notification
SERVER_PROCESS_COUNT	Server process count alert
SERVER_PROCESS_CPU	Server process CPU alert
SERVER_PROCESS_MEMORY	Server process memory alert
SERVER_PROCESS_OFF	Server process alert OFF notification
AGENT_ACTIVE	Agent activated alert
AGENT_INACTIVE	Agent deactivated alert
AGENT_REACTIVATED	Agent reactivated alert
URL	URL alert
TOO_MANY_EVENT	Too many events alert
CLOUD_WATCH	CloudWatch alert
EXCEPTION	Exception alert

Event History And Monitoring

Home > Select Project >  **Alert** > **Event History**

Event History allows you to view events that have occurred in the project and analyze their detailed information. It supports Elasticsearch DSL queries, enabling you to find events that match specific conditions quickly and accurately.

Viewing Event History

Click the **event title** in the search results to move to a screen where you can analyze detailed data for the time period when the event occurred. This allows you to comprehensively understand the situation before and after the event.

1. Go to **Alert > Event History**.
2. In the **Event History** screen, use **Time Selection** to set the period for viewing event records.
3. Enter a query in the filter input field and click the [Magnifying Glass] button.
 - If no query is entered, all events are displayed.

Column Settings

You can choose which columns to display or hide.

1. In the **Event History** screen, click the [ Column Settings] button at the top left.
2. In the **Column Settings** window, select the columns you want to show or hide.
 - You can change the column order in **Displayed Items**.
 - You can search for specific columns using the **Column Name Search** field.

Column Item	Description
Event Name	The title of the event

Column Item	Description
Message	The event message or snapshot information
Event Status	The current status of the event
Agent Name	The name (Oname) of the agent where the event occurred - Displayed as blank if the event is not associated with a specific agent
Agent Group	The agent type (OkindName) where the event occurred - Displayed as blank if the event is not associated with a specific agent
Agent Node	The server (OnodeName) of the agent where the event occurred - Displayed as blank if the event is not associated with a specific agent
Event Occurred Time	The time when the event occurred
Event Resolved Time	The time when the event was resolved - Displayed as - for events without a resolved state

3. After completing the settings, click the [Confirm] button to apply the column configuration.

↓ CSV Download

You can download the query results as a CSV file.

1. Go to **Alert > Event History** and click the [CSV] button.
2. The list of events that match the current filter conditions is saved as a CSV file.

View Only Events in Progress

You can view only events that are currently in progress.

1. Go to **Alert > Event History** and click the [View Only Events in Progress] button.
2. The following condition is automatically added to the filter query and applied to the search.

Stateful: **true** and Status : **"ON"**

Event Search

- **Precise Search:** Specify exact conditions to retrieve only the events you need
- **Composite Conditions:** Search across multiple fields at once to handle complex conditions in a single query
- **Flexible Patterns:** Supports various search methods such as wildcards and partial matches
- **High Performance:** Fast searches powered by indexed data

Searchable Fields

Field names and values are not case sensitive.

- **Field Name Tag:** Event types that support the field
- **Field Type:** The data type of the field
 - **#:** Number
 - **T:** String
 - **🔵:** Boolean
 - **📅:** Date

Event Information Fields

Field Name	Type	Description	Example Value
Title	String	Event title	"Database Connection Error"
OffTitle	String	Event recovery title	"RECOVERED: Database Connection Error"
Message	String	Event message	"Connection timeout occurred"
OffMessage	String	Event recovery message	"RECOVERED: Connection timeout occurred"

Field Name	Type	Description	Example Value
Level	String	Current event level	Critical, Warning, Info
OriginLevel	String	Original event level	Critical, Warning, Info
Status	String	Event status	ON, OFF, CANCEL, ACKNOWLEDGE, MAINTENANCE, DISABLED
ActCount	Number	Number of occurrences	2
MetricName	String	Metric name	"memory"
MetricValue	String	Metric value	"85.5"
OffValue	String	Recovery value	"72.8"
MetricThreshold	String	Threshold value	"80"
alertType	String	Event type	"BASIC", "METRICS", "TRANSACTION", etc.
alertId	String	Unique identifier of the event rule	"zf3uojer0fv4v7"

Type-specific Supported Fields

ⓘ These fields are supported only for specific event types. If the event type does not match, the value may be empty.

Field Name	Type	Event Type	Description	Example Value
eventRule	String	Basic, Metrics	Event trigger rule	"memory ≥ 80"
field	String	Real-	Log search	"content"

Field Name	Type	Event Type	Description	Example Value
		time Log	key	
keyword	String	Real-time Log	Log search value	"Error"
logCategory	String	Real-time Log	Log category	"AppLog"
logContent	String	Real-time Log	Log content	"00:00000:00009:2025/04/06 23:03:52.55 server Error: 1601, Severity: 17, State: 3\n00:00000:00009:2025/04/06 23:03:52.55 server There are not enough 'user connections' available to start a new process."

Target Fields

Field Name	Type	Description	Example Value
Oid	Number	Agent unique identifier	-98765432
Oname	String	Agent name	"web-server-01"
Okind	Number	Agent type unique identifier	867318026
OkindName	String	Agent type name	"web-server"
Onode	Number	Agent node unique identifier	334634079
OnodeName	String	Agent node name	"production-node-1"

Unique Event Fields

Field Name	Type	Description	Example Value
Id	Number	Unique identifier of the event	5768121
UUID	String	Unique event identifier	"550e8400-e29b-41d4-a716"

State / Flag Fields

Field Name	Type	Description	Example Value
Stateful	Boolean	Whether the event is state based	true, false
Disabled	Boolean	Whether the event is disabled	true, false
Escalation	Boolean	Whether escalation is applied	true, false
SystemEvent	Boolean	Whether it is a system event	true, false

Query Syntax by Search Type

ⓘ For the basic search syntax, refer to the [WhaTap log search query syntax](#) documentation.

1. Keyword Search

Events that contain the string "Database Connection"

```
title: "Database Connection"
```

2. Multiple Value Search

Retrieves events that include any one of the specified values.


```
title: Database Connection
```

Events whose current event level is Info or Warning

```
level: info warning
```

3. Pattern Search

You can search using patterns with wildcards.

Find events that occurred on specific agents using a name pattern

```
oname: web-*-prod
```

タイトルが Connection で終わるイベントを検索

```
title: *Connection
```

4. Complex Condition Search

You can combine multiple conditions to perform precise searches.

Events with a Critical level and a title containing Database

```
level: critical and title: Database
```

Warning or Critical events from a specific agent

```
oname: web-server-01 and level: Warning Critical
```

5. OR Condition Search

Retrieves events that satisfy **at least one** of the specified conditions.

Events where the title or message contains a specific keyword

```
title: Connection or message: Connection
```

6. Exclusion Conditions

You can exclude specific conditions from the search.

All events except those with an Info level

```
not level: info
```

Real-time Notifications

The **Alert status** feature displays pop-up notifications when events configured in **Event Configuration** occur. You can receive alerts immediately when an event is triggered, allowing you to quickly recognize issues and respond. You can also review notifications that have already occurred or were missed, helping you understand and resolve situations where real-time response was difficult.

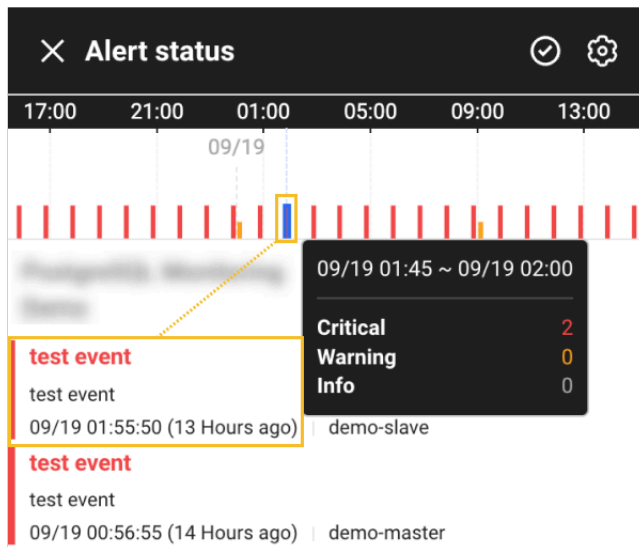
Viewing Real-time Notifications

Real-time notifications appear as pop-ups in the upper-right corner of the screen. You can check the time, title, and message of each notification.

- To close an individual notification, click the  button in the upper-right corner of the message.
- To close all pop-up notifications displayed on the screen, click **Close All Notifications**.

Viewing Notification History

1. Click the  icon at the top of the screen after selecting a project.
2. The **Alert status** window appears.
 - You can view up to 100 notifications from the past 24 hours for the current project.
 - Selecting an individual event item navigates you to the related menu.
 - In the time-based chart within the **Alert status** window, selecting a specific time range takes you to the corresponding event items.



- To view the full notification history, click **View All Notification History**. This navigates to **Alert > Event History**.

ⓘ If the **Alert status** list is empty, the **View All Notification History** button is displayed. Disable the **Receive notifications without reception tags** option.

Resolving Events in Bulk

You can resolve all ongoing events at once from the **Alert status** list. The bulk resolution process may take approximately 30 seconds to 1 minute to complete.

- When using the **Add Acknowledge** feature, a large number of recovery (resolved) notifications may be generated for currently ongoing events.
- Even after using the **Add Acknowledge** feature, events may occur again if the event rules remain enabled and the data still exceeds the threshold.
- Recovery notifications can be disabled by editing the event in **Alert > Event Configuration**. Disable **Send additional notification when the event is resolved** or **Recovered notification**.

How to use:

1. Click the  icon at the top of the screen after selecting a project.
2. When the **Real-time Notifications** window appears, click the  icon in the upper-right corner.
3. When the message `Resolve all ongoing events in the real-time notification list.` appears, click the **Resolve** button.

Configuring Real-time Notifications

1. Click the  icon at the top of the screen after selecting a project.
2. When the **Alert status** window appears, click the  icon in the upper-right corner.
3. Configure each option in the real-time notification settings window.
 - **Popup notifications:** Displays notifications as pop-ups even when the **Alert status** window is closed.
 - **Show resolved events:** Allows resolved events to be displayed in the **Alert status** list.
 - **Preview notification sound:** Lets you preview browser notification sounds. Click the  button next to **Critical** or **Warning**. If you cannot hear the sound, check your browser settings.

System Events

WhaTap Monitoring provides common system events that are automatically detected and generated across the platform. This document explains the types of events generated by the system, their trigger conditions, and recommended response actions.

EVENT_BLOCKED_BY_SYSTEM

If an excessive number of events occurs within a short period of time in a project, the WhaTap system detects this situation and blocks further events. When events are blocked, the system delivers the following message to the user. If you encounter this event, review and adjust your event configuration settings.

 **Critical**



Event Message

Excessive events detected, event blocked for stability. name: server agent

Project No	8
Project Name	new server
Target	server agent
Event Title	EVENT_BLOCKED_BY_SYSTEM
Event Time	(GMT+0900) 2024-11-18 17:58:58

Log In to WhaTap

Events are blocked as a result of excessive events in the server agent of new server project. If no events occur in server agent for a period of time, it is automatically unblocked.

Blocking Criteria

Events are blocked based on the following criteria:

1. Agent-level blocking

If **10 events** occur within **1 minute** from a single agent, all subsequent events from that agent are blocked.

2. Project-wide blocking triggered by multiple agents

If **3 or more agents** are blocked within **1 minute** according to condition 1, events from the entire project are blocked.

3. Project-level blocking

If **100 events** occur within **1 minute** in a single project, all events from that project are blocked.

How to Release the Block

When events are blocked by the system, no events must occur from the affected agent or project for **1 minute**.

- Disable all configured events from **Alert > Event Configuration**. **Review your event settings and adjust thresholds to prevent excessive event generation.**
- Before configuring events, use the **Simulation** feature to estimate how many events may occur.

Laboratory

This menu provides new or experimental features to users. Through  **Laboratory** or **Sitemap** in the side menu, you can access it. The provided features may differ depending on the product.

 The functionality of the **Laboratory** menu is currently in Beta version and may cause unexpected errors. Be careful when using the **Laboratory** menu. It is not recommended to use the menu in handling important data or in the operating environment. If you have any feedback or issues, feel free to contact the support team at support@whatap.io.

- **MXQL Data**

MXQL is a query language to flexibly query WhaTap's performance data (metrics). It is used to comprehensively search and use the metrics collected from multiple agents in a project. For more information about **MXQL**, see [the following](#).

- **Metrics Search**

You can search for specific metrics according to the categories based on fields and tags. For more information, see [the following](#).

 In the side menu, if the  **Laboratory** menu does not appear, you can access it through **Sitemap**.
Under the side menu, select . If the **Sitemap** window appears, select a desired menu in the **Laboratory** section on the lower right of the window.

Browser Agent Collection Data

The WhaTap browser agent collects events that occur from each user's browser, reprocesses the collected data, and uses the data for monitoring. All collected data is user session dependent.

Types

The following lists the types of data that the WhaTap browser agent collects.

- **Long**: A unique ID data type in the long to string32 format.
- **TimeStamp**: Time stamp upon measurement, in milliseconds (ms).
- **HighResolutionTime**: Elapsed time in milliseconds (ms)
- **TimeSet**: The start time stamp and the elapsed time are displayed for display of a timeline.
- It is any of **ResourceType** : `document` , `xhr` , `fetch` , `beacon` , `css` , `script` , `image` , `font` , `media` , and `other` .
- **ErrorType** : It is any of `console` , `onError` , `promiseRejection` , `fetchError` , `xhrError` , and `messageError` .

Common collection data

When any event occurs, the collected basic data is as follows:

Data name	Type	Description	Example
sessionID	Long	A unique ID issued when a web application user starts navigating to a web page. The maximum duration of a session is 4 hours, and the ID expires after 15 minutes of inactivity. IDs are stored in browser cookies.	x7dbvq64ga80sd
userID	Long	It is a unique ID issued when a web application user first connects and	z7eqnnitlc19j9

Data name	Type	Description	Example
		has no expiration time. IDs are stored in browser cookies.	
sendEventID	Long	This is a unique request ID issued when data is transmitted to the collection server after an event occurs.	x4rp4tnuq25i0h
page.url	String	The URL of the page where an event occurred.	https://site.whatap.io/8/dashboard?dataKey=abcd
page.host	String	The URL domain name of the page where an event occurred.	site.whatap.io
page.path	String	The URL path of the page where an event occurred.	/8/dashboard
page.query	String	The URL parameter of the page where an event occurred.	dataKey=abcd
page.protocol	String	The URL protocol of the page where an event occurred.	https
page.title	String	The document title of the page where an event occurred.	SaaS Monitoring No.1 WhaTap Monitoring WhaTap
userAgent	String	The useragent of the browser where an event occurred.	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/103.0.0.0 Safari/537.36

Page Load Event

This event occurs when you access the web application and the page content is fully loaded. The basic collection data is as follows:

Data name	Type	Description	Example
navigationTiming.startTimeStamp	TimeStamp	It is the time stamp of <code>navigationStart</code> . Time stamp of the page that starts loading	1658057382311
navigationTiming.eventID	Long	This is a unique ID issued when a page loading event occurs.	x7mkg24tags307
navigationTiming.redirect	TimeSet	Time taken for HTTP redirection.	{duration: 10, start: 315}
navigationTiming.cache	TimeSet	Time taken to load the app cache resources.	{duration: 5, start: 325}
navigationTiming.dns	TimeSet	Time taken for browsing the website domain.	{duration: 5, start: 330}
navigationTiming.connect	TimeSet	Time taken for TCP connection setting.	{duration: 6, start: 335}
navigationTiming.ssl	TimeSet	Time taken for the ssl handshake process.	{duration: 5, start: 336}
navigationTiming.download	TimeSet	Time taken for document downloading.	{duration: 100, start: 341}
navigationTiming.firstByte	TimeSet	It is the time taken for the browser to request a document from the server and receive the first byte.	{duration: 10, start: 341}
navigationTiming.domInt	HighResolutio	It is the time when the browser	441

Data name	Type	Description	Example
eractive	nTime	finished parsing the document from <code>navigationStart</code> . DOM is ready.	
navigationTiming.domContentLoaded	TimeSet	Time taken to build the DOM tree.	{duration: 50, start: 441}
navigationTiming.domComplete	HighResolutionTime	This is the time between <code>navigationStart</code> and completion of loading documents and resources.	491
navigationTiming.domLoad	TimeSet	Time taken for the <code>onload</code> event.	{duration: 1, start: 491}
navigationTiming.backEndTime	HighResolutionTime	Time from <code>navigationStart</code> to resource downloading.	126
navigationTiming.frontEndTime	TimeSet	Time from browser parsing completion to load event completion.	{duration: 51, start: 441}
navigationTiming.loadTime	HighResolutionTime	Time from <code>navigationStart</code> to <code>loadEventEnd</code> .	492
totalDuration	HighResolutionTime	This is the time between <code>navigationStart</code> and completion of importing the final page.	550

Visually Complete

Visually Complete (VC) is the measured point in time at which the content in the area visible to the user (the viewport) is fully rendered on screen.

While page load time is the point at which all resources have been loaded, Visually Complete is closer to the point at which the screen appears fully rendered to the user's eyes. Even if there are still resources loading in the background, the user perceives the page as ready once the screen looks stable. It is a metric that complements actual perceived performance.

Name	Type	Description	Example
visuallyComplete	HighResolutionTime	Time of viewport content rendering completion, relative to <code>navigationStart</code> . In ms	1820

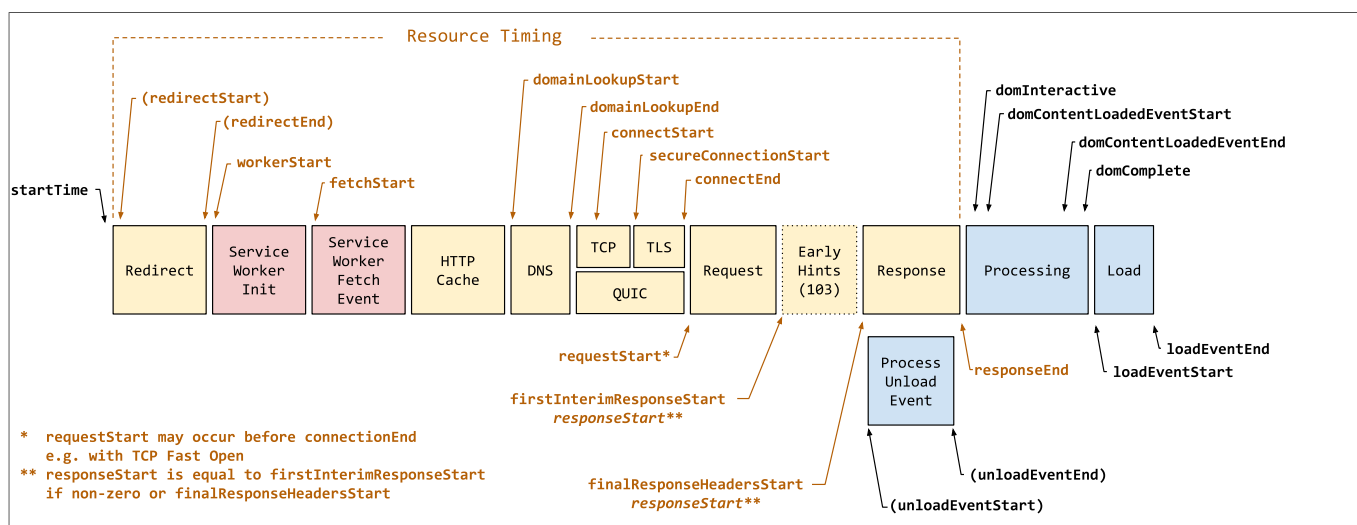
Page load time

Page load time is the total time from the moment the web browser requests the URL of a page until all of the page's resources (HTML, CSS, JavaScript, images, etc.) are downloaded and processed, and the browser completely renders the page so that it is available.

To collect page load times, browser agents collect them through the **Navigation Timing API**, one of the web APIs provided by the browser. This API provides measurements of the times for each step (e.g. request start, response, rendering start and completion).

Navigation timing

There are measurement events accessible through the **PerformanceTiming** interface that are measured in milliseconds. The following is a list of those events in the order in which they occur.



Event	Description	Remarks
<code>navigationStart</code>	Starting point of page navigation.	-
<code>redirectStart</code>	First HTTP redirection time point.	If there is no redirection, the value is <code>0</code> .
<code>redirectEnd</code>	Time point when the response to the last redirection has been received.	If there is no redirection, the value is <code>0</code> .
<code>fetchStart</code>	Time point when the browser begins the process of fetching resources.	Time point when cache lookup has begun.
<code>domainLookupStart</code>	Starting point for DNS lookups for domain names.	The browser begins converting the host name in the URL into an IP address. It can have the same value as <code>fetchStart</code> if there is cached DNS information or information in the local <code>hosts</code> file.
<code>domainLookupEnd</code>	Time point when the DNS lookup has been completed.	The browser successfully obtains the IP address for the domain. Even if the DNS lookup fails, it is logged with this time point.
<code>connectStart</code>	Time point when the connection to the server has begun.	TCP connection begins. If you are reusing the persistent connection, you can have the same value as <code>domainLookupEnd</code> .
<code>connectEnd</code>	Time point when the connection to the server has been established completely.	The TCP handshake is completed. For HTTPS, the SSL/TLS handshake is also completed by this time point. Even if the connection fails, it is logged with this time point.
<code>secureConnectionStart</code>	Time point when the HTTPS connection has	It is recorded only when applicable.

Event	Description	Remarks
	begun.	
requestStart	Time point when requesting resources to the server has begun.	-
responseStart	Time point when receiving the first byte of the response from the server has begun.	It is used to calculate the server's initial response time (Time to First Byte, TTFB) starting from the time the first byte of the HTTP response header was received.
responseEnd	Time point at which the last byte has been received from the server.	Time point when the entire response (header and body) has been received. If you have a redirection response, it is the time point when the last byte of the final response has been received.
domInteractive	It is the time point when DOM parsing has been finished and the script is ready.	-
domContentLoadedEventStart	Time point when the <code>DOMContentLoaded</code> event has occurred.	At this time, the loading and parsing of the HTML document have been completed and the DOM tree has been completely built.
domContentLoadedEventEnd	Time point at which all <code>DOMContentLoaded</code> event handlers have completed execution.	-
domComplete	Time point at which both DOM loading and subresource loading have been completed.	-
loadEventStart	Time point just before	Time point immediately after the DOM is fully

Event	Description	Remarks
	the load event.	configured and all resources have been loaded. Moment just before the <code>window.onload</code> event handler has been executed.
<code>loadEventEnd</code>	Time point at which all load event handlers have completed execution.	The page and all subresources are fully loaded.

Router change event

This event occurs when you move to a page in a single page application (SPA) type web application. The basic collection data is as follows:

Data name	Type	Description	Example
loadTime	HighResolutionTime	Time taken to change the router.	500
startTimeStamp	TimeStamp	Timestamp of the router change start time.	1658057382311
endTimeStamp	TimeStamp	Timestamp of the router change completion time.	1658057382811
pageLocation	String	The URL of the page where a router change event occurred.	https://site.whatap.io/8/dashboard?dataKey=abcd
host	String	URL domain name of the page URL where a router change event occurred.	site.whatap.io
path	String	The URL path of the page where a router change event occurred.	/8/dashboard

Data name	Type	Description	Example
query	String	The URL parameter of the page where a router change event occurred.	dataKey=abcd
protocol	String	The URL protocol of the page where a router change event occurred.	https
isComplete	Boolean	This value is used to distinguish when a router is changed to another page while the router change is not completed.	true

Resource Load Event

This event occurs when you call a resource while using the web application. The basic collection data is as follows:

Data name	Type	Description	Example
startTime	HighResolutionTime	Time from <code>navigationStart</code> to the resource request start.	1000
startTimeStamp	TimeStamp	Timestamp of the resource request start.	1658057382321
eventID	Long	A unique ID issued each time a resource is fetched.	x280or4ok0kqo0
type	ResourceType	Resource type.	image
url	String	Resource URL.	https://unpkg.com/purecss@2.0.3/build/pure-min.css?dataKey=abcd

Data name	Type	Description	Example
urlHost	String	Domain name of the resource URL.	unpkg.com
urlPath	String	Path of the resource URL.	/purecss@2.0.3/build/pure-min.css
urlQuery	String	Parameter of the resource URL.	dataKey=abcd
urlProtocol	String	Protocol of the resource URL.	https
is3rdParty	Boolean	This value is used to identify resources requested by the 3rd party.	false
timing.duration	HighResolutionTime	Time taken to complete the resource download.	61
timing.size	Number	Resource encoding size.	20
timing.redirect	TimeSet	Time taken for HTTP redirection.	{duration: 10, start: 315}
timing.cache	TimeSet	Time taken to load the app cache resources.	{duration: 5, start: 325}
timing.dns	TimeSet	Time taken to search the resource domain.	{duration: 5, start: 330}
timing.connect	TimeSet	Time taken for TCP connection setting.	{duration: 6, start: 335}
timing.ssl	TimeSet	Time taken for the ssl handshake process.	{duration: 5, start: 336}
timing.firstByte	TimeSet	It is the time taken for the browser to request resources from the server and receive the first byte.	{duration: 10, start: 341}
timing.download	TimeSet	Time taken to complete the resource	{duration: 100, start: 341}

Data name	Type	Description	Example
		download.	
resourceInfo.method	String	Resource request method.	POST
resourceInfo.status	Number	Resource request state.	200

Error event

It is an error that occurred from the browser while you are using the web application. The basic collection data is as follows:

Data name	Type	Description	Example
message	String	Message displayed when an error occurs.	SyntaxError: Unexpected end of JSON input
stack	String	Stack information displayed when an error occurs.	at printWarning (webpack-internal:///./node_modules/react-dom/cjs/react-dom.development.js:86:30) at error (webpack-internal:///./node_modules/react-dom/cjs/react-dom.development.js:60:7)
timestamp	TimeStamp	Timestamp of an error occurred.	1658057382321
type	ErrorType	Error type.	onError
status	Number	This is the error status value to be collected when the error type is <code>xhr</code> or <code>fetch</code> .	501

Data name	Type	Description	Example
url	String	This is the AJAX request URL when the error type is <code>xhr</code> or <code>fetch</code> .	<code>https://site.whatap.io/yard/api?pcode=8&type=summary</code>

Core Web Vitals Event

Core Web Vitals are three metrics that Google considers important, among various measurement values that affect the user experience of web content. The basic collection data is as follows:

Data name	Type	Description	Example
lcp	HighResolutionTime	Largest Contentful Paint (LCP), which reports the rendering time of the largest image or text block within the viewport based on the first page loaded time.	12334
fid	HighResolutionTime	First Input Delay (FID), from the first time you interact with the page (e.g. clicking a link, tapping a button, or using a custom JavaScript-based control) to the time to start the event handler in response to that interaction.	7
cls	Number	Cumulative Layout Shift (CLS) is a score that quantifies the unexpected layout shifts caused by the dynamically loaded content.	2

Per-view CLS/INP measurement in SPAs

In a Single Page Application (SPA), once a page is loaded, the browser does not reload a new page even when a

view transition occurs. Such a view transition is called a **soft navigation**.

Previously, CLS and INP were measured based on only the single page first entered. Now, view transitions in an SPA are distinguished as individual views, and CLS and INP are measured separately for each transitioned view. This lets you identify more precisely which screen has layout shifts or slow responses to user input.

The INP metric collected alongside is as follows.

Name	Type	Description	Example
inp	HighResolutionTime	Interaction to Next Paint (INP). Time until the screen next responds to a user interaction. In ms	180