

Oracle Pro Monitoring

release date. 2026. 09. 11.



Table Of Contents

• Oracle Pro Monitoring	5
◦ Getting started	7
◦ Support Environment	9
◦ Installation	
■ Agent Installation	14
■ Installation checking items	27
■ Troubleshooting	29
◦ Configuration	31
■ DMX Agent Setting	33
■ XOS agent configuration	43
◦ Management	46
◦ Dashboard	47
■ Instance List	49
■ Instance Map	51
■ Monitoring a Database Instance	59
■ Monitoring Multiple Instances	87
■ RAC Monitoring	112
■ Flex Board	140
■ Using Flex board	143
■ Managing the widgets	153
◦ Analysis	
■ Performance Comparison	
■ Trend Comparison	158
■ Top SQL Comparison	168

■ Workload Analysis	
■ Wait analysis.....	172
■ Top SQL by Wait	176
■ Cube	181
■ Session Analysis	
■ Lock tree	184
■ Parallel query tree	195
■ Session History	202
■ SQL Analysis	
■ Top SQL	204
■ SQL Plan analysis	214
■ Metrics Analysis	
■ Metrics Chart.....	219
■ Metrics Trend Analysis	242
◦ Metrics.....	245
■ Oracle Pro Metrics.....	247
■ Metrics Chart	257
■ Metrics Search	260
■ Metric Anomaly Detection	262
◦ Stat/Report	
■ Tablespace Size.....	265
■ SGA size	274
■ Database Parameter	276
■ Report	278
■ SQL Statistics.....	281
◦ Log	286

■ Application	288
■ Live Tail	290
■ Log Explorer	303
■ Log Trend	310
■ Log Search.....	317
■ Log setting	332
■ Parsing the logs.....	367
◦ Alert	382
■ Event Configuration	0
■ Metrics	389
■ Composite Metrics.....	411
■ Anomaly Detection	418
■ Real-time logs	422
■ Composite log	435
■ Event Configuration	449
■ Event Notification Settings.....	453
■ Event Notification Format.....	475
■ Event History and Monitoring	483
■ Real-time Notifications.....	491
■ System Events	494
◦ Management	
■ Correlated project management.....	497
◦ Laboratory	504
■ OpenMetrics Explorer.....	505

Oracle Pro Monitoring

The DB performance management monitors the database performance in real time, diagnoses and analyzes problems, tunes problematic SQLs, and monitors whether problems have been resolved.

The WhaTap database monitoring service is designed to manage a series of operations in cycles. It is configured to help maintain consistent levels of availability and performance by monitoring, diagnosing, and analyzing the database behaviors. When a problem occurs in the database, this service sends an alarm to the staff to recognize the situation and helps you identify the root cause by analyzing the database situation more precisely through the collected data.

❗ For more information about the supported features for each database platform, see [the following](#).

Oracle Pro Monitoring features

WhaTap's Oracle Pro monitoring products presents an innovation that goes beyond the traditional query-based data collection. Traditional query-based approaches place significant load on the database server, which can cause system performance degradation. This approach uses the Direct Memory Access (DMA) technique to directly access the server's memory and extract the required data, thereby placing little load on the database server. This is a huge advantage, especially in high-performance database environments.

Additionally, query-based collection methods have long data update cycles and can collect data only at specific query execution times. In comparison, the Oracle Pro monitoring can collect data almost in real time through memory access, and can collect sessions and SQL performance data at an interval of at least 1/20 seconds. This allows database administrators to monitor the system performance more precisely and respond immediately when any problem occurs.

Another important feature of the Oracle Pro monitoring product is that data collection is not interrupted even in a system failure or hang. Data collection in these scenarios can be difficult using traditional query-based approaches. By using Oracle Pro, database administrators can ensure data continuity under any circumstances, thereby maximizing the stability and availability of the system.

Ultimately, WhaTap's Oracle Pro monitoring product is an essential technology for improving the efficiency and accuracy of database monitoring, while minimizing system load and ensuring business continuity. This provides

a foundation for companies to make faster, more accurate, and more reliable decisions.

- **Memory access method**

WhaTap Oracle Pro collects data by directly accessing the DB server memory using the **Direct Memory Access** (DMA) technology. It overcomes the limitations of the existing query methods and provides fast and efficient data collection without server overload.

- **Load-free monitoring data collection**

Oracle Pro accesses the DB server memory directly to extract data, and collects the performance data 20 times per second with little server load. This supports stable and efficient monitoring even in high-performance DB environments.

- **Precise SQL performance data collection**

Oracle Pro collects data 20 times per second, enabling precise analysis and providing much more detailed information than the 5-second interval SQL collection of the existing query-based method. This allows administrators to identify issues in real time and respond immediately.

- **Monitoring analysis possible even when the DB is in Hang status**

Oracle Pro does not stop collecting data even in case of DB failure or hang situations. This ensures data continuity and enables analysis and response even upon a failure, maximizing the system stability and availability.

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.

Support Environment

Before starting the Oracle Pro monitoring, check the following support environment.

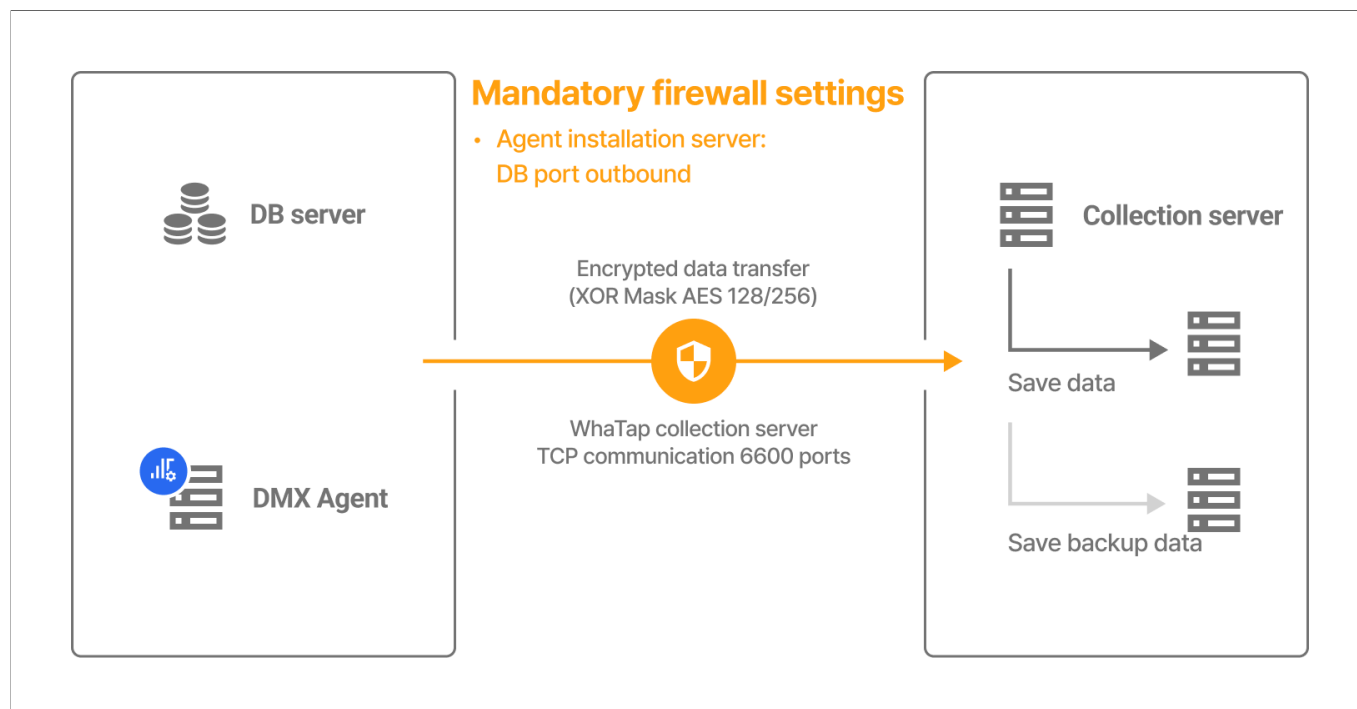
Supported versions for the database

Linux Server and Oracle 11gR2 or later are supported.

ⓘ It is required to open the outbound on the port 6600 so that the DBX agent can send data to the WhaTap server.

Agent structure

Oracle Pro Monitoring monitors by installing an agent directly on the database server. Unlike WhaTap's other database products, you can collect XOS metrics without installing the XOS agent.



Service deployment models

You can use WhaTap Monitoring as a SaaS or on-premises (self-hosted) service. Whichever you choose, you use the same agents and the same monitoring features.

Deployment	Support	Description
SaaS	✓	Connects to collection servers operated by WhaTap. See the firewall section for regional addresses
On-premises (self-hosted)	✓	Collection servers you build and operate yourself (air-gapped environments supported). Contact your sales representative for the scope

ⓘ You can also deploy WhaTap on-premises in air-gapped environments without internet connectivity.

Feature differences by deployment model

Whichever you choose, the monitoring features are the same. The mobile app is the only exception and is available on the service.whatap.io SaaS service only.

ⓘ To use certain features on-premises, a separate agreement is required.

Firewall

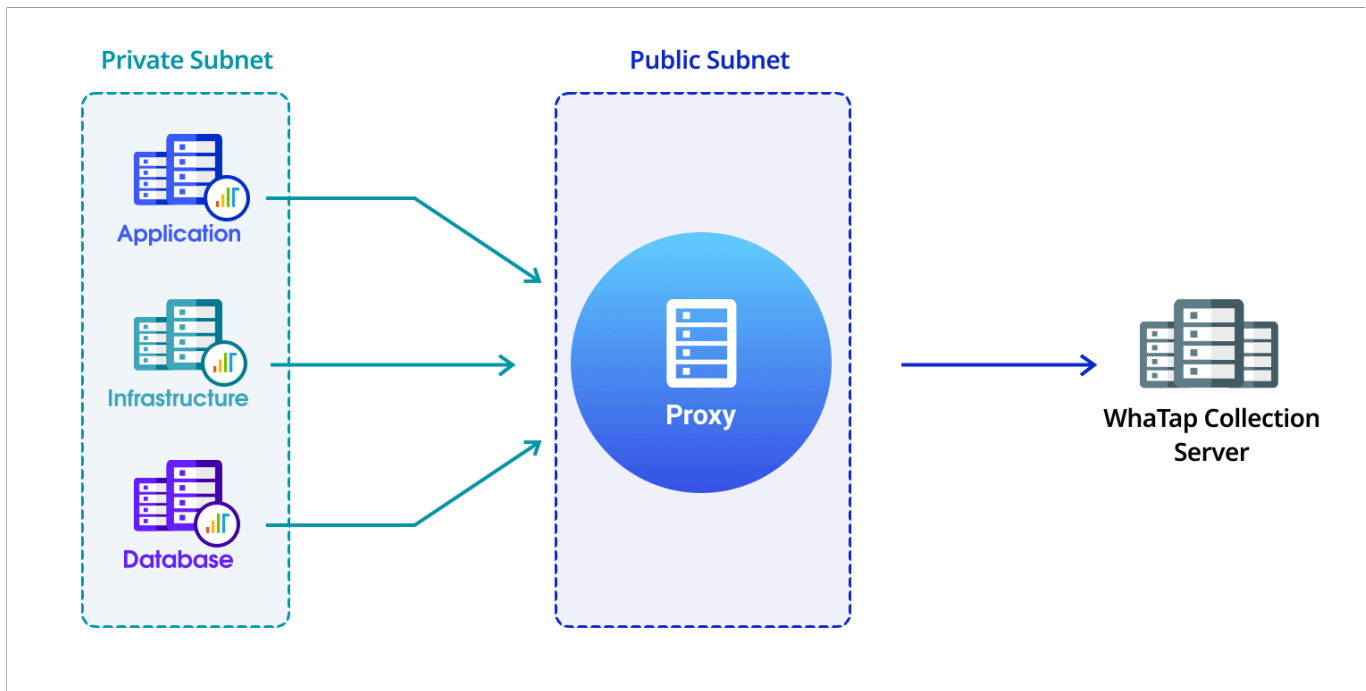
The WhaTap agent must be able to access the collection server through **TCP port 6600**. Allow access to the collection server address that is closest to the monitored target.

Source: WhaTap agent

Cloud	Destination	Destination IP	Port
AWS	WhaTap collection server (Seoul)	13.124.11.223 / 13.209.172.35	TCP 6600
	WhaTap collection server (Tokyo)	52.68.36.166 / 52.193.60.176	TCP 6600
	WhaTap collection server (Singapore)	18.138.0.93 / 18.139.67.236	TCP 6600
	WhaTap collection server (Jakarta)	108.136.91.69 / 108.137.158.44	TCP 6600
	WhaTap collection server (Thailand)	43.209.109.219 / 43.209.48.86	TCP 6600
	WhaTap collection server (Mumbai)	13.127.125.69 / 13.235.15.118	TCP 6600
	WhaTap collection server (California)	52.8.223.130 / 52.8.239.99	TCP 6600
	WhaTap collection server (Virginia)	107.23.220.101 / 54.236.221.105	TCP 6600
	WhaTap collection server (Frankfurt)	3.125.142.162 / 3.127.76.140	TCP 6600

Cloud	Destination	Destination IP	Port
Azure	WhaTap collection server (Seoul)	52.231.66.38 / 20.194.5.115	TCP 6600
	WhaTap collection server (Tokyo)	52.246.169.54 / 20.210.27.232	TCP 6600
Kakao	WhaTap collection server (Seoul)	61.109.237.237 / 61.109.238.166	TCP 6600

If the agent cannot directly access the collection server, use the provided proxy module to access it.



Supporting the browser

WhaTap Monitoring is available in web browsers and mobile Apps.

Browser	Recommended	Supported version
Google Chrome	O	111 or later

Browser	Recommended	Supported version
Mozilla FireFox	X	Latest version
Edge	X	Latest version
Safari	X	Latest version

Support guidance

- For browser compatibility and performance reasons, the latest version of Chrome is recommended.
- The user interface (UI) is implemented using HTML5 standard technologies, so Internet Explorer is not supported.

Limitations

WhaTap's web interface does not support mobile browsers. To access WhaTap on a mobile device, install the WhaTap app for Android or iOS. The WhaTap mobile app is optimized for mobile environments. For more information about the WhaTap mobile app, see [this document](#).

Agent Installation

The following guides you to the basic installation method for using the WhaTap database monitoring service.

1 Creating a project

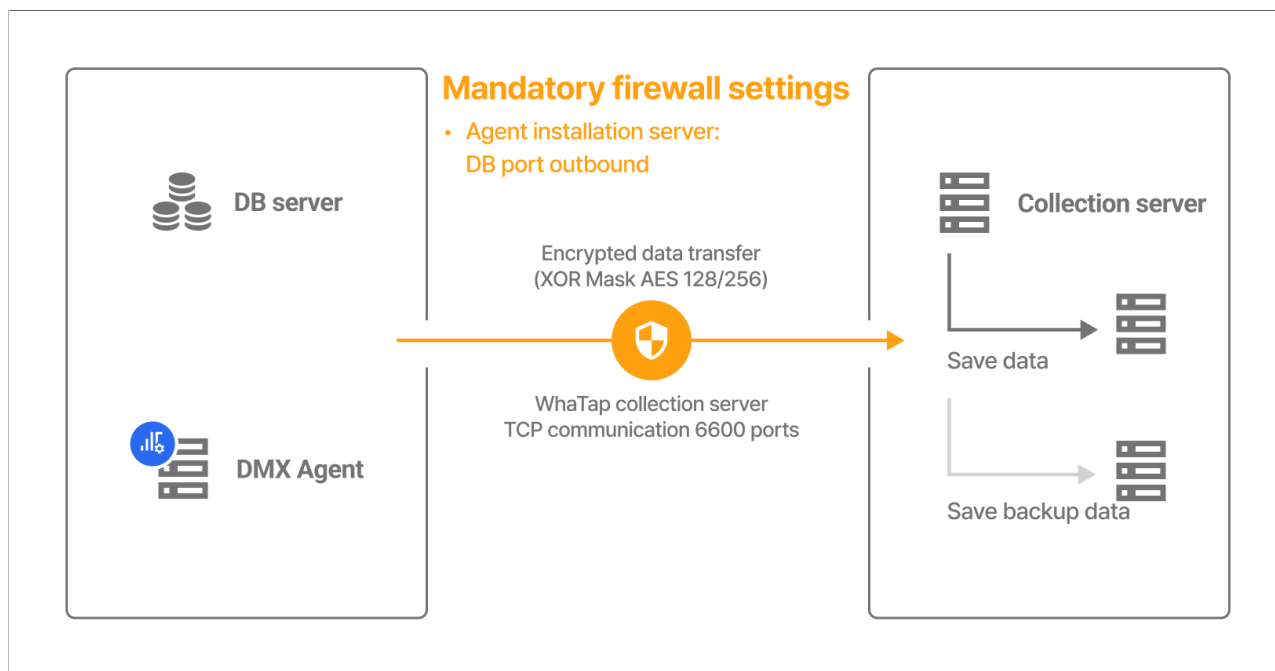
Create a project before installing the agent.

1. Log in to the WhaTap Monitoring Service.
2. In the left side menu, click **All Projects** > **+ Project**.
3. On the **Select product** screen, select the product you want to install.
4. Enter or select the following items:
 - **Project name**: Enter the project name.
 - **Data server region**: Select the data server region. A region is a group of data centers that provide cloud services. Your data will be stored in the selected region.
 - **Time zone**: Set the time zone to be used for alerts and report generation.
 - **Notification language setting**: Set the language for alert notifications. (Supports Korean and English)
 - **Project groups**: Group multiple projects for easier management. Select a group if applicable.
 - **Project description**: Add additional details or a description for your project.
5. After completing all settings, click the **Creating a project** button.

ⓘ If you add a project while an organization is selected, you must set the **Groups of organization** field.
For more information about groups, see the related document.

2 Checking the configuration diagram

It is the configuration to install the agent on the database server for Oracle Pro monitoring.



Firewall settings are required and all data is transmitted encrypted. The port 6600 is used for communication with the WhaTap collection server.

3

Account creation

Create an account with roles required for database monitoring. Log in with the root account and then create accounts.

- In general cases

SQL

```
create user {DB_User} identified by {DB_Password};
grant create session to {DB_User};
grant select any dictionary to {DB_User};
```

- In case the basic monitoring is used in the multi-tenant DB (Oracle 12c or later): The `c##` account is created.

SQL

```
create user c##{DB_User} identified by "{DB_Password}";
grant create session to c##{DB_User};
grant select any dictionary to c##{DB_User};
alter user c##{DB_User} set container_data = ALL container = CURRENT;
```

- in case of using the kill session

SQL

```
grant alter system to {DB_User}
```

- In case of viewing the hidden parameters

SQL

```
create view sys.whatap_vw_ksppi as select * from sys.x$ksppi;
grant select on sys.whatap_vw_ksppi to {DB_User};
create synonym {DB_User}.whatap_vw_ksppi for sys.whatap_vw_ksppi;

create view sys.whatap_vw_ksppcv as select * from sys.x$ksppcv;
grant select on sys.whatap_vw_ksppcv to {DB_User};
create synonym {DB_User}.whatap_vw_ksppcv for sys.whatap_vw_ksppcv;
```

Support features for each role

The roles other than the **Default role** do not need to be assigned depending on the environment. However, some functions may not work.

Role	command (Single DB)	command (Multitenant DB)	Menu and feature	Details
Default role	grant create session to	grant create session to	Access	-

Role	command (Single DB)	command (Multitenant DB)	Menu and feature	Details
	whatap;	c##whatap;		
	grant select any dictionary to whatap;	grant select any dictionary to c##whatap;	Monitoring a Database Instance / Monitoring Multiple Instances / SQL Statistics	When granting individual roles, see the following table
	-	alter user c##whatap set container_data = ALL container = CURRENT;		When using a multi-tenant DB (12C or higher), this is a required role to query the pdb.
Session kill	grant alter system to whatap;	grant alter system to c##whatap container=all;	Active sessions > Session kill	-
Hidden parameter query	create view sys.whatap_vw_ksppi as select * from sys.x\$ksppi; grant select on sys.whatap_vw_ksppi to DB_User;		Hidden parameters are included in the parameter query menu.	To query hidden parameters, separate view/synonym



Role	command (Single DB)	command (Multitenant DB)	Menu and feature	Details
	<pre>create synonym DB_User.whatap_vw_ksppi for sys.whatap_vw_ksppi;</pre> <pre>create view sys.whatap_vw_ksppcv as select * from sys.x\$ksppcv;</pre> <pre>grant select on sys.whatap_vw_ksppcv to DB_User;</pre> <pre>create synonym DB_User.whatap_vw_ksppcv for sys.whatap_vw_ksppcv;</pre>			creation and role assignment are required.

Instead of **select any dictionary** , you can grant the following roles individually.

Role	command (Single DB)	command (Multitenant DB)
Active sessions / Lock tree / Metrics information	<pre>grant select on v_\$sqlarea to whatap;</pre>	<pre>grant select on v_\$sqlarea to c##whatap;</pre>
	<pre>grant select on v_\$transaction to whatap;</pre>	<pre>grant select on v_\$transaction to c##whatap;</pre>
	<pre>grant select on v_\$instance to whatap;</pre>	<pre>grant select on v_\$instance to c##whatap;</pre>
	<pre>grant select on v_\$sysstat to whatap;</pre>	<pre>grant select on v_\$sysstat to c##whatap;</pre>



Role	command (Single DB)	command (Multitenant DB)
	grant select on v_\$lock to whatap;	grant select on v_\$lock to c##whatap;
	grant select on v_\$sesstat to whatap;	grant select on v_\$sesstat to c##whatap;
	grant select on v_\$event_name to whatap;	grant select on v_\$event_name to c##whatap;
	grant select on v_\$session to whatap;	grant select on v_\$session to c##whatap;
	grant select on v_\$sql to whatap;	grant select on v_\$sql to c##whatap;
	grant select on v_\$process to whatap;	grant select on v_\$process to c##whatap;
	grant select on v_\$statname to whatap;	grant select on v_\$statname to c##whatap;
	grant select on v_\$system_event to whatap;	grant select on v_\$system_event to c##whatap;
	grant select on v_\$osstat to whatap;	grant select on v_\$osstat to c##whatap;
	grant select on v_\$px_session to whatap;	grant select on v_\$px_session to c##whatap;

!	Role	command (Single DB)	command (Multitenant DB)
		grant select on v_\$sga to whatap;	grant select on v_\$sga to c##whatap;
		-	grant select on v_\$con_sysstat to c##whatap;
	Parameter information	grant select on v_\$parameter to whatap;	grant select on v_\$parameter to c##whatap;
	Wait information / Event information	grant select on v_\$event_name to whatap;	grant select on v_\$event_name to c##whatap;
		grant select on v_\$system_event to whatap	grant select on v_\$con_system_event to c##whatap;
		grant select on v_\$system_wait_class to whatap;	grant select on v_\$system_wait_class to c##whatap;
		-	grant select on v_\$con_system_wait_class to c##whatap;
		grant select on dba_data_files to whatap;	grant select on cdb_data_files to c##whatap;
	Tablespace information	grant select on	grant select on
		grant select on dba_data_files to whatap;	grant select on cdb_data_files to c##whatap;

!	Role	command (Single DB)	command (Multitenant DB)
		dba_free_space to whatap;	cdb_free_space to c##whatap;
		grant select on dba_temp_files to whatap;	grant select on v_\$sort_segment to c##whatap;
		grant select on v_\$temp_extent_pool to whatap;	grant select on v_\$tempfile to c##whatap;
		grant select on v_\$containers to whatap; (12c or later)	grant select on v_\$containers to c##whatap;

- To use the previous accounts, go to [Create DB User File](#). If you do not have any permission, you may not be able to proceed with normal monitoring.
- In the example code, `whatap` is the DB user account name. Change it to your account name.
- Enter your password in `DB_Password` in the example code.

4

Checking the access key

The **access key** is the unique ID to enable the WhaTap service.

In the installation guide section, select **Getting the access key**. After the **access key** has been issued automatically, proceed to the next step.

- After a project has been created, the **Agent installation** page appears automatically. If the

❗ **Agent installation** does not appear, select **Management > Agent installation** on the left of the screen.

5 Download agent

On the agent Installation screen, select the Oracle version that matches your environment and then select **Download**.

❗ For users who cannot download tar files due to security settings, ZIP files are also provided. On the installation screen, select the **.zip Download** button.

6 Agent CONFIG.

The agent is recommended to be installed with an Oracle account and the Oracle connection environment variables must have been set. Proceed with the agent configuration in the following order.

❗ In case of installation from the user account other than the Oracle account

- You need the role (`read`) to read the shared memory created by Oracle.
- If the `perms` value is **640** in the output of the `ipcs -m` command, the agent can be installed by another account in the same group.
- If the `perms` value is **** 600 **** and ****** in the result of the `ipcs -m` command for **Oracle 12c r2** or later, you can set the parameter to change the `perms` value to **** 640 **** as follows. To apply the changes, restart the DB.

```
alter system set allow_group_access_to_sga=true scope=spfile;
```

whatap.conf configuration

Go to the unzipped folder and then check the `whatap.conf` file. In `whatap.conf`, enter the project access key, WhaTap server data, and `ipckey`.

whatap.conf

```
license={AccessKey}
whatap.server.host=13.124.11.223/13.209.172.35 # WhaTap server information
ipckey={ipckey}
```

- ✓ • The `ipckey` option value can be acquired via the `ipcs -m` command. This is the key value (hexadecimal) of the Oracle to be monitored.
- If you have installed multiple databases on a server, see [the following](#) to enter the `ipckey`.

Creating the initial configuration file

To create an initial configuration file, connect to the database with the **SYS** account (or an equivalent account) to receive and process information. The account and password are not saved.

BASH

```
./dmxinit.sh {sys_user} {sys_password}
```

After executing the command, the following five files are created:

`dmx.init`, `class.init`, `dbname.init`, `stat.init`, `event.init`

- ⚠ • If you can connect without `sys_user` and `sys_password` (as connecting with `sqlplus '/as sysdba'`), you do not need to add `sys_user` and `sys_password` as arguments.

BASH



```
./dmxinit.sh
```

- If the `sys_user` and `sys_password` strings contain special characters, add the escape character (`\`) before each special character.

```
BASH
```

```
./dmxinit.sh whatap whatap\!@pwd
```

Creating DB users and changing passwords

Generate an encrypted UID for database connection. Enter the username and password and then run the shell script (or batch file).



When changing the password, enter the new password and follow the same procedure.

```
BASH
```

```
./uid.sh {DB_USER} {DB_PASSWORD}
```



- After setting it once, it collects data from the database server to be monitored through the encrypted UID.
- To create a DB user file, enter the project access key in the [whatap.conf](#) file. [Checking the access key](#)
- In the Azure database environment, enter `DB_USER` in the form of `DB_USER@DB_name`.
- If special characters are included in `DB_USER` or `DB_PASSWORD`, enter the escape character (`\`) together before any special characters.



Example

```
./uid.sh whatap whatap\!pwd
```

If there are multiple special characters, add the escape character(\) for each.

```
./uid.sh whatap whatap\!\@pwd
```

7

Starting the monitoring

Execute a shell script (or batch file) from the path where you have installed the agent.

```
./start.sh
```

To use it like a daemon, execute the following command. However, it works only in the environment where **nohup** has been installed.

```
./startd.sh
```

Next steps

Checking the installation

If you have created a project, installed an agent, and applied all agent options, see the checklist in [the following](#).

Installation troubleshooting

It provides various problems that may occur when installing the agent and specific instructions for resolving them. For more information, see [the following](#).

Agent setting

It provides various features for monitoring by applying some options to the agent configuration file ([whatap.conf](#)). For more information, see [the following](#).

Starting the monitoring

After configuring all settings, the agent starts collecting metrics data from the database server. First, check whether the monitoring data has been collected in **Instance List**. For more information about **Instance List**, see [the following](#).

Installation Checking Items

The database monitoring has been installed completely. After installation, check if the installation was successful by doing the following:

Checking the agent installation

Run the [start.sh](#) shell script file or the [start.bat](#) batch file from the installation path of the database agent. When you run the agent, it starts collecting monitoring data from the database. If the WhaTap logo is displayed in the terminal after executing the shell script, it is normally executed.

start.sh

```
$ ./start.sh
```

```
  _  _  _  _  _  
| | | | | _ _ _ _  
| | | | _ _ _ _ _  
| _ | _ _ _ _ _ _  
  _ _  
Just Tap, Always Monitoring  
WhaTap DBX version X.Y.Z YYYYMMDD
```

❗ To use it like a daemon, execute the `startd.sh` command. However, it works only in the environment where **nohup** has been installed.

❗ When the DMX agent is run, the **dmx** and **prx** processes are executed.

Checking the logs

- [whatap.log](#)

All logs related to the agent startup and collection can be found in `$WHATAP_HOME/logs/whatap.log` path. If the size of the `whatap.log` file exceeds 25 MB, it is backed up with the file name of `whatap.log.[n]`. To clean up files, delete the files starting from the greatest (old) number.

- `prx.log`

This log file records major activities and events that occur during DMA (Direct Memory Access)-based data collection. Logs are recorded when the DMX agent is restarted. You only need to run the `start.sh` or `startd.sh` (execution of daemon) file once in the agent installation path.

Checking the agent processes

You can see active WhaTap DMX agent and prx processes with the following commands.

```
$ ps -ef | grep dmx
```

```
$ ps -ef | grep prx
```

Checking the monitoring

To check that the log has been normally uploaded from the server and the console has been registered normally, go to the initial screen of [WhaTap Monitoring Service](#). Select the project to check and then select **Management > Agent List** on the left. From **Agent List**, you can check the name of the database server. To set the names of database servers separately, see [the following](#).

Status	Name (Initial)	ID	Start time	Active / Inactive	Agent version	Agent IP	DB IP	DB Port	DB Version	CPU Core	
Active	DBX-2-60-6379	2122651642	2024-03-13 18:36:35	2024-03-13 18:36:35	1.6.32 20240306	172.17.0.1	192.168.2.60	6379	7.0.11	1	
Active	DBX-2-60-6380	-2134789231	2024-03-13 18:36:10	2024-03-13 18:36:10	1.6.32 20240306	172.17.0.1	192.168.2.60	6380	7.0.11	0	
Active	DBX-2-60-6381	-137985273	2024-03-13 18:36:15	2024-03-13 18:36:15	1.6.32 20240306	172.17.0.1	192.168.2.60	6381	7.0.11	0	
Active	DBX-2-60-6382	1859113661	2024-03-13 18:36:20	2024-03-13 18:36:20	1.6.32 20240306	172.17.0.1	192.168.2.60	6382	7.0.11	0	

Troubleshooting

Check for possible problems upon database agent installation.

Firewall configuration checking

Unblock your firewall. Monitoring data cannot be sent to the server if the TCP outbound firewall is set for the WhaTap server.

How to check the firewall

It is normal only when the following connection-related information appears after executing the `telnet` command. (`telnet [Server IP] [Server Port]`)

Example

```
$ telnet 52.193.60.176 6600
Trying 52.193.60.176...
Connected to 52.193.60.176.
Escape character is '^['.
```

! The WhaTap collection server data can be found in **Management > Agent installation**.

Constraints

Multiple regions impossible to be selected as a single project

- In the [WhaTap monitoring service](#) website, when creating a project, you cannot select multiple regions. When using multiple regions, create separate projects.
- If servers exist in multiple regions as in the cloud environment, the collection servers must be located in each region to avoid constraints on the performance data collection such as network latency.

- WhaTap builds collection servers for each region to meet customer requirements.

Duplicate agent names (ONAME) impossible to be used

WhaTap DBX monitoring uses the agent's IP address and port to identify the agent. If the database server to be monitored in the user environment uses the same IP and port, the WhaTap server cannot identify the server instance.

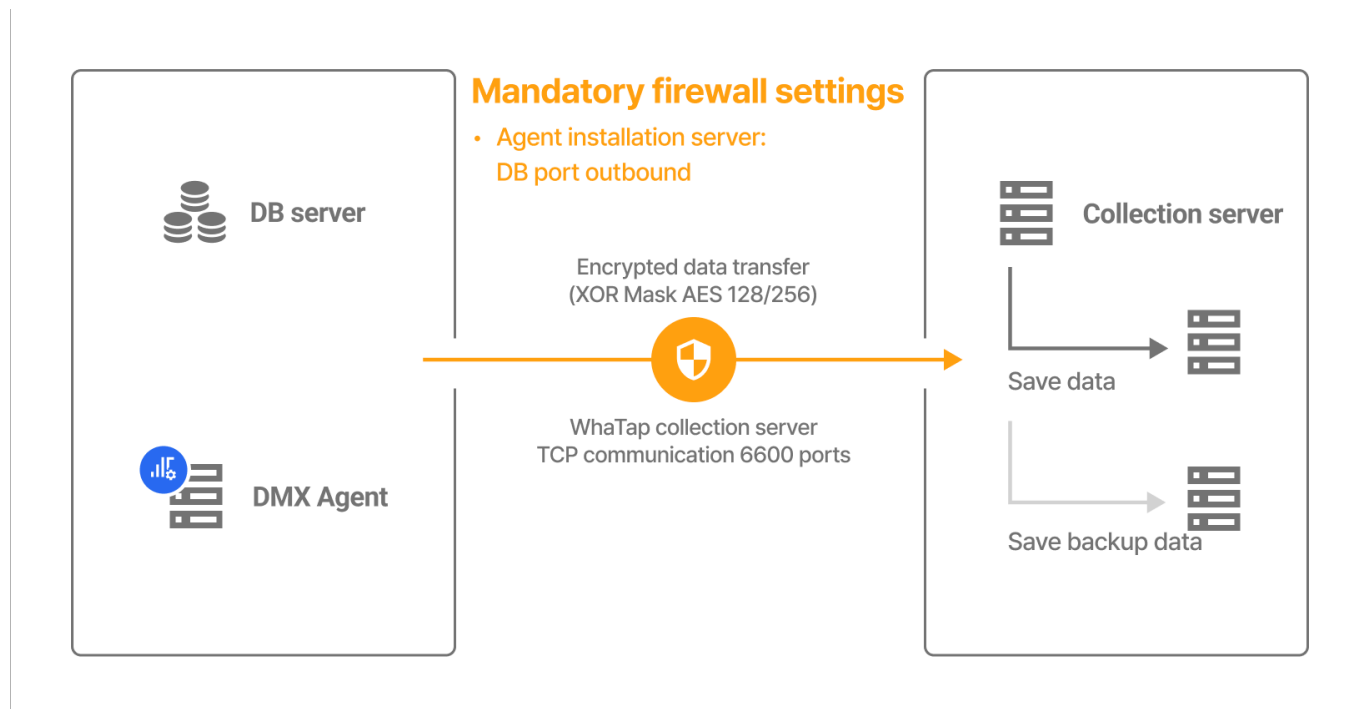
- If the server instance is not dynamically expanded, you can bypass it by manually setting a name for database identification by seeing [the following](#).
- If the databases to be monitored use internal addresses or duplicate IPs due to network virtualization, use separate database naming patterns by seeing [the following](#).

Configuration

You can take a look at the structure of the WhaTap agent, set additional functions through various options, and monitor the resources of the DB server to be monitored.

Agent structure

The following displays the WhaTap agent's structure map.



- **Collection server**

It collects and stores the database performance data collected by agents, extracts statistical data, and efficiently provides them to users. The collection server can be set by region. Different collection server addresses are allocated to each region. The collection server address may differ depending on the selected region. When creating a project, the region is also set.

- **DMX agent**

Collects performance data using the DMA (Direct Memory Access) method and sends it to the collection server. The agent is installed directly on the target database server.

- **Network:** The WhaTap monitoring agent uses the external communication (TCP) port 6600 to transfer the collected monitoring data to the WhaTap collection server.

prx process options

The **prx** process monitors the **dmx** process and restarts it when it is abnormally crashed or excessive CPU or memory is used. You can set the reboot criteria through the following options:

prx.conf

```
cpu_limit=10  
mem_limit=500000
```

- **cpu_limit**

Default `10` (unit: %)

Restarts the **dmx** process if the CPU utilization exceeds the specified value.

- **mem_limit**

Default `500000` (500 MB, unit: KB)

Restarts the **dmx** process if the memory usage exceeds the specified value.

DMX Agent Setting

Enter the settings required for the DMX agent in the [whatap.conf](#) file. For the options that can be set, see the following.

Default options

- **license** String

To install the agent, set the project access key granted from the server. The project access key includes the agent's project and the encryption key for secured communication.

- **whatap.server.host** String

Default `127.0.0.1,127.0.0.1`

Specify the IP address of the collection server to transfer the data collected by the agent. If there are multiple IP addresses for collection server redundancy, use comma (,) as the delimiter. The collection server proxy daemon must be serviced in the listening state for the set IP address.

- **ipckey** String

It can be obtained through the `ipcs -m` command. This is the key value (hexadecimal) of the Oracle to be monitored.

❗ The `ipcs` command provides information about inter-process communication (IPC). The `-m` option queries shared memory resources.

- **rac_name** String RAC

This option sets the group name of the instances grouped in Oracle's **Real Application Clusters (RAC)**. This option allows you to change complex RAC cluster group names to concise and easy ones to identify. The RAC cluster group name set using this option can be checked on the WhaTap Monitoring Service screen.

For example, if the RAC cluster group name is too long, such as '`my_long_rac_cluster_group_name`', you can set a concise name as follows:

```
whatap.conf
```

```
rac_name=my_rac_cluster_group
```

It is displayed as 'my_rac_cluster_group' in the WhaTap Monitoring Service, allowing users to quickly identify and monitor the RAC cluster group.

Database data collection options

- **bind_elapse** Int

Default 3

It sets to collect bind values if SQL runs longer than the specified time (seconds). If the setting is 0, this option does not work.

Even if the bind_interval is not set, the bind values are collected when the SQL execution time is greater than the set value.

- **bind_interval** Int

Default 0

It sets the bind capture interval in seconds. Setting the default value (0) disables the bind capture.

Collection is made at the interval set in v\$sql_bind_capture.

Because there can be a lot of data, collection is performed by bind_max_count since the last collection time.

- **bind_max_count** Int

Default 1000

It sets how many bind capture values to collect. Only the bind capture values after the last collection time can be collected.

- **codefrom / codeto** String

This option changes the character encoding when there is a problem with Korean characters broken in the SQL text. The value can be selected from the list in iconv -l.

e.g. codefrom=EUC-KR, codeto=UTF-8

- **conn_fail_count** Int

Default 15

It sends a notification when 15 consecutive connection errors occur. You can adjust the notification

occurrence criteria by entering the desired number.

- **event** String

Default `db file scattered read` , `db file sequential read` , `direct path read` , `direct path write` ,
`enq: TX - row lock contention` , `latch free` , `library cache lock` , `library cache pin`

In the RAC environment, two additional metrics are collected by default: gc cr block receive time and gc current block receive time.

It enables collecting event-related metrics. Write the name of the metric to collect by referencing the `EVENT` column such as `v$system_event` , and `v$event_name` . You can collect additional metrics along with the basic metrics by writing them in a way like `event+=` . You can enter multiple items using the comma (,) as the delimiter without spaces.

For each event, the following three values are collected.

- time waited: Total waiting time in the event (unit: 1/100 sec). e.g. db file sequential read time waited
- timeouts: Number of waits that exceeded the timeout. e.g. db file sequential read total timeouts
- total waits: Total number of events. e.g. db file sequential read total waits

- **interval** Int

Default `3`

It sets the data collection cycle in seconds.

❗ Oracle Pro collects active sessions and SQL data at an interval of 1/20 second through memory access. Because the stats are generated based on the data collected in this manner, data loss is minimized. When transmitting the current DB status according to the `interval` option, the stat, event, SQL stat, xview, lock, and PQ are read directly from memory for quick processing. On the other hand, the binds, DB parameters, plans, tables, table spaces, and wait classes are collected through queries.

- **multi_db_stat** Boolean

Default `true`

When using the Multi DB, set whether to collect metrics by database or by summing them for all instances.

- **ora_hidden_param** Boolean

Default false

It sets whether to collect hidden parameter values. Set the option value to true to collect hidden parameters.

- **plan_change** Int

Default 1

It sets whether to collect the change history for the plan.

- 1 : Excludes sys
- 2 : Includes sys

- **plan_elapse** Int

Default 3

It collects the plans that take more than 5 seconds to run.

- **segments** Boolean

Default true

It collects the segment size information. You can check the size by selecting the database or tablespace in the Statistics/Report > Tablespace size menu.

- **segments_exclude_sys** Int

Default 1

The sizes are collected except for accounts corresponding to sys.

- **segments_hour** Int

Default 5

It sets the time to collect segment sizes. The default value is 5 and collection starts at 5 AM.

- **segments_max_count** Int

Default 50

Only 50 large segments are collected and stored for each tablespace.

- **segments_min_size** Int

Default 10

The minimum unit for collecting segment sizes is 10 MB by default. Segments less than the set value are not collected.

- **segments_sys** String

Default:

Details

!

<summary>Viewing 18 accounts</summary>

`APPQOSSYS`, `AUDSYS`, `CTXSYS`, `DBSFUSER`, `DBSNMP`, `DVSYS`,

`GSMADMIN\_INTERNAL`, `LBACSYS`, `MDSYS`, `OJVM SYS`, `OLAP SYS`,

`ORDDATA`, `ORDSYS`, `OUTLN`, `SYS`, `SYSTEM`, `WMSYS`, `XDB`

It is the list of accounts corresponding to sys. To add to the above accounts, you can add with `segments_sys+='xxx'`.

- **sqlstat_send_interval** Int

Default: 1 minutes

Sets the interval, in minutes, for generating and sending SQL statistics data. You can specify any integer value of 1 or higher. For more precise statistical analysis, set it to 1. This setting enables detailed 1-minute analysis in the Top SQL screen.

- **stat** String

Default:

Details

!

<summary>Viewing 20 metrics</summary>

`CPU used by this session`, `data blocks consistent reads - undo records applied`, `db block changes`,

`db time`, `enqueue waits`, `execute count`, `logons current`, `opened cursors current`,



```
`parse count (hard)`, `parse count (total)`, `physical reads`, `physical writes`, `redo entries`,  
`rollback changes - undo records applied`, `session logical reads`, `sorts (disk)`, `sorts (memory)`,  
`transaction tables consistent reads - undo records applied`, `user commits`, `user rollbacks`
```

It enables collecting statistics-related metrics. Write the name of the metric to collect by referencing the `NAME` column such as `v$sysstat`, and `v$statname`. You can collect additional metrics along with the basic metrics by writing them in a way like `stat+=`. You can enter multiple items using the comma (,) as the delimiter without spaces.

- **tablespace** Int

Default `1`

Selects the view used to query tablespace free space. If a query timeout occurs when querying `dba_free_space`, you can switch to another view depending on your environment.

- `1` : Query `dba_free_space`
- `2` : Query `dba_lmt_free_space`
- `3` : Query `usage_metrics`

Supported version: dmX 1.0.1x or later

- **tablespace_interval** Int

Default `60`

Sets the collection interval for tablespace information in minutes. The minimum value is `5`; if you set a value smaller than `5`, it is automatically corrected to 5 minutes and a log noting the correction is recorded. You can also set a value greater than `60`.

Supported version: dmX 1.0.8c or later

- **tablespace_send_opt** Int

Default `0`

Sets the transmission method for tablespace information. This can mitigate errors caused by duplicate transmission of the same data in environments with many tablespaces.

- `0` : Split transmission

- **1** : Batch transmission

Supported version: dmX 1.0.8c or later

- **topevent** Int

Default **50**

It collects changes in key event-related metrics according to the values set in descending order. The data having the **stat_top** and **event_top** categories are used.

❗ If no value is set for this option, it operates according to the setting for **topstat** .

- **topstat** Int

Default **50**

It collects changes in key statistics-related metrics according to the values set in descending order. The data having the **stat_top** and **event_top** categories are used.

- **trace_tkprof_path** String

Specifies the path to the tkprof executable used when running SQL trace.

Supported version: dmX 1.0.4y or later

- **waitclass_opt** Int

Default **1**

Sets the collection method for Wait Class metrics.

- **1** : Memory-based collection. Collected at the same interval as other metrics.
- Values other than **1** : Maintains the existing method (query-based, fixed 1-minute interval).

Supported version: dmX 1.0.8c or later

- **xos** Boolean

Default **true**

This option enables the XOS agent ([xos.conf](#)) to additionally monitor the resources of the database server.

- **xview_min** Int

Default `100` (1 second)

The unit is centisecond (1/100 of a second). This option processes only the jobs whose elapsed time is greater than the set value.

- `-1` : This option is disabled.
- `0` : It collects all jobs.

User Metric Collection (SQLMON)

SQLMON periodically collects the results of user-defined SQL queries and stores them as metrics.

It is suitable for lightweight queries that return numeric results.

Multi-row results are stored up to a **maximum of 100 rows**, and a result value is recognized as numeric when it is an `int` or `float`. In addition to `SELECT`, you can also use `INSERT`, `UPDATE`, and `DELETE` statements.

INSERT/UPDATE/DELETE support: dmX 1.0.8s or later · Multi-row storage (up to 100 rows): dmX 1.0.5g or later

• `sqlmon` `Int`

Default value: `1`

- `1` : Run both dbx and sqlmon simultaneously
- `2` : Run only sqlmon

Supported version: dmX 1.0.4y or later

Configuration

1. Create a `sqlmon` directory under the dbx agent and create `<filename>.sql` files for the queries you want to collect.

```
dbx/
└─ sqlmon/
   └─ active_sessions.sql
   └─ lock_count.sql
```

2. Save the query to be collected as a `<filename>.sql` file.

Write the execution options on the **first line** and the SQL statement from the **second line** onward.

Option format

```
interval(min);run_time;tag_name
SQL statement
```

Item	Description	Example
Interval	Execution interval in minutes, minimum 1	1 → runs every 1 minute
Run time	HHMM-HHMM format, multiple ranges separated by comma (,). Set 0 to run at all times	0900-1100 → runs only 09:00–11:00
Tag name	Tag name stored in metrics. If omitted, the file name is used as the tag name	whatap

Example: 1 min interval, 09:00–11:00, tag name whatap

```
1;0900-1100;whatap
SELECT count(*) FROM active_sessions
```

Example: 1 min interval, always running, file name as tag

```
1;0
SELECT count(*) FROM active_sessions
```

- ⓘ • Run time can span across dates. (e.g., **2300-0100** → 23:00 to 01:00 the next day)
- When omitting the tag name, do not add a semicolon (;) after the run time.
- Do not add a semicolon (;) at the end of the SQL statement.

3. Enable SQLMON in **whatap.conf**.
4. Restart the agent.
5. In the  **Sitemap > Metrics Search** menu, check the **sqlmon** category.

```
# whatap.conf
sqlmon=1
```

Applying the AES 256 encryption

WhaTap DMX agent encrypts the collected data and transmits the encrypted data to the server. You can change this depending on the importance of the data or setting. Basically, the XOR operation and encryption through the AES algorithm are used, and encryption and decryption are performed by dividing the plain text into 128-bit units. It can be expanded up to 256 bits depending on the setting.

Add the following options to the [whatap.conf](#) file in the path where the WhaTap DMX agent was installed.

whatap.conf

```
1 cypher_level=256
```

Add the settings and then restart the agent.

Agent name setting

Automatic configuration

If you do not set the agent name in the agent configuration ([whatap.conf](#)), it is automatically created by combining the hostname in the following way: **DMX-{hostname}**.

Manual configuration

To set the agent name manually, enter a desired name for the [whatap.name](#) or [object_name](#) option in the [whatap.conf](#) file. The entered name is assigned to the agent name.

whatap.conf

```
whatap.name=agent-1234
# or
object_name=agent-1234
```

XOS Agent Configuration

You can additionally monitor resources on the database server by setting the following options in the `xos.conf` file. To use this option, set the `xos` option to `true` in the `whatap.conf` file.

- **interconnect_nic** String **RAC**

It collects network card information corresponding to the set option value. If this option is not set, interconnect-related metrics are not collected.

`xos.conf`

```
# example
interconnect_nic=enp0s3
```

Enter the network interface used for interconnect of **Real Application Cluster (RAC)** among the network interfaces confirmed by the `netstat -i` command. The set values are used to monitor the performance and activities of the network card.

- **cpu_limit** Int

Default 0

It collects the process data with a value greater than the set limit. If it is 1.23% (1/100% step), set it to 123 .

- **mem_limit** Int

Default 10240

It collects the process data with a value greater than the set limit. 10240 (Kbytes) indicates 10 MB.

- **disk_usage_interval** Second

Default 60

Displays the disk utilizations, free sizes, and total sizes for each instance. If the value is 0 , disk usage is not collected.

- **process** String

The execution status of processes can be monitored. Enter the process name as an option value. Multiple items can be registered using comma (,) as the delimiter.

`xos.conf`

```
process={process_name}
```

For example, to set the top and lock.sh processes to be monitored, set the options as follows:

```
xos.conf
```

```
process=top,lock.sh
```

ⓘ You can use the `process` option to send an alert when a process running on the database is ended. For more information, see [the following](#).

- **iplist_interval** Second

You can collect a list of IPs for your database servers. Set the time interval in seconds (at least 3 seconds) for collection.

- **file** String

To use the **Log** menus (**Live Tail**, **Log Trend**, **Log Search**, **Log Configuration**), enter the absolute path in the `file` option for the log files to be monitored.

```
xos.conf
```

```
file=/opt/oracle/log
```

✓ To add a logfile to monitor, add options in the same way as `file` , `file1` , `file2` .

```
xos.conf
```

```
file={log_file_path1}
```

```
file1={log_file_path2}
```

```
file2={log_file_path3}
```

```
file3={log_file_path4}
```


User Metric Collection (SHMON)

SHMON periodically collects the results of user-defined shell scripts and stores them as metrics.

- **shmon** Int

Default value: `1`

Supported version: dbx 2.55.01 or later & xos 1.2.0f or later

Configuration

1. Create a `shmon` directory in the agent installation path.
2. Save the script to be collected as a `.sh` file.
3. Enable SHMON in `xos.conf`.
4. Restart the agent.
5. In the  **Sitemap** > **Metrics Search** menu, check the `shmon` category.

```
# xos.conf  
shmon=1
```

Management

Agent deletion

Deleting from the server where the agent have been installed

Among Java processes running on the server, it terminates the processes that contain the string, `whatap.agent.dmx` . Run `stop.sh` from the path where you installed the agent. Afterwards, delete the agent file and user permission registered in the agent.

Deleting on the service screen

To delete the agent from [Whatap Monitoring Service](#), go to **Agent List**. Before deleting the agent, terminate the agent process on the server where the agent has been installed.

1. Select a project to delete the agent from the **Project List**.
2. On the left of the screen, select **Management > Agent List**.
3. Select  on the utmost right of the agent to delete from **Agent List**.

- ⓘ • Only inactive agents can be deleted.

Status	Name (Initial)	ID
 Active	 	
 Inactive	 	
 Active	 	

- Depending on the agent version, the delete function may not be supported. Update the agent to the latest version.

Dashboard

After installing the WhaTap agent, log in [WhaTap Monitoring Service](#). The initial screen is **Project List**. In **Project List**, you can see the installed project. If a project is selected from the **Project List**, you can enter [Instance List](#).

In the WhaTap database monitoring service, the dashboard allows you to see the overall status of the project at a glance. You can check the total resource size for monitoring and filter the resources to be monitored on the real-time dashboard.

- It is configured to show key performance metrics and status. The performance metrics are updated every 5 seconds.
- In a single instance, multiple databases are filtered.
- Based on the collected data, it provides the function to analyze the overall trend for a specific day. You can analyze slow queries after identifying active sessions at a specific time point.

Instance List

You can see the list of database servers in which the agent has been installed. It is configured to display key performance metrics and status. The performance metrics are updated every 5 seconds, and the columns in the list can be arranged through settings. For more information, see [the following](#).

Monitoring a Database Instance

- You can monitor key metrics and the details on active sessions of the database server in real time and check the warnings that may be problematic. You can select desired key metrics and monitor the overall system status.
- The real-time monitoring checks the performance metrics at an interval of 5 seconds. You can check the **Active sessions**, **Lock tree**, and **Process Info**.

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

Monitoring Multiple Instances

All instances registered in the project are monitored in real time at an interval of 5 seconds. You can also filter and monitor by specific instances. For more information, see [the following](#).

RAC Monitoring

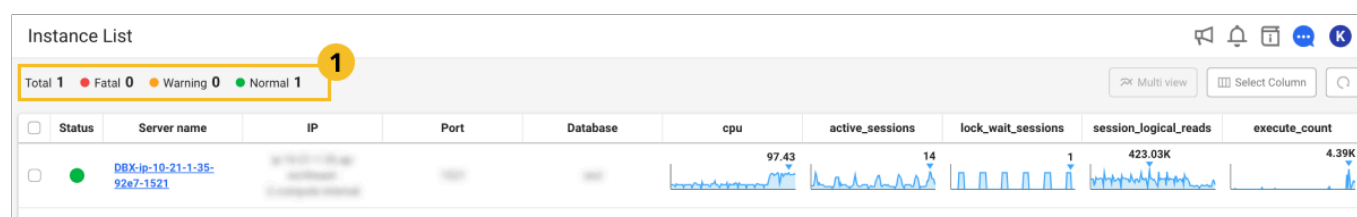
RAC Monitoring can monitor the databases formed as a cluster in Oracle Database. For more details, see [the following](#).

Instance List

Home > Select Project > **Dashboard** > **Instance List**

You can see the list of database servers in which the agent has been installed. You can check key performance metrics and status for each instance. The performance metrics are updated every 5 seconds, and the order of the columns and the metrics in the list can be changed through settings.

Basic screen guide



- The **1** area displays the status of instances set in the project. The displayed status data matches the total number of instances and matches the **Status** column data in the table.
 - Normal**: Number of normal agents.
 - Warning**: Number of agents where the threshold of the event set in **Alert > Notifications** exceeds the **Warning** level.
 - Fatal**: Number of agents where the threshold of the event set in **Alert > Notifications** exceeds the **Fatal** level.

ⓘ For more information about **Alert**, see [the following](#).

- If you click an abnormal icon in the **Status** column, the **Event** window appears. You can see the history of events in the instance.
- If you select each instance, **Monitoring a Database Instance** appears. You can monitor the detailed information about the selected instances in real time.

❗ For more information about **Monitoring a Database Instance**, see [the following](#).

- It provides the key performance metrics in graph charts. The highest number appears on the chart. Hover your mouse over the chart to see the number at that location. For more information about the performance metrics, see [the following](#).
- You can see real-time alerts by selecting  at the upper right of the screen. For more information about the alert settings, see [the following](#).

❗ For more information about the fixed area on the top of the screen, see [the following](#).

Comparing the instance metrics

You can select multiple instances to compare the performance metrics. If you select an instance to compare the performance metrics from the list, the **Multi view** button becomes active on the upper right of the screen. If you select **Multi view**, the **Monitoring Multiple Instances** menu appears. You can see the graph charts for comparing data for key performance metrics and the detailed information about active sessions and lock trees.

❗ For more information about **Monitoring Multiple Instances**, see [the following](#).

Displaying or hiding columns

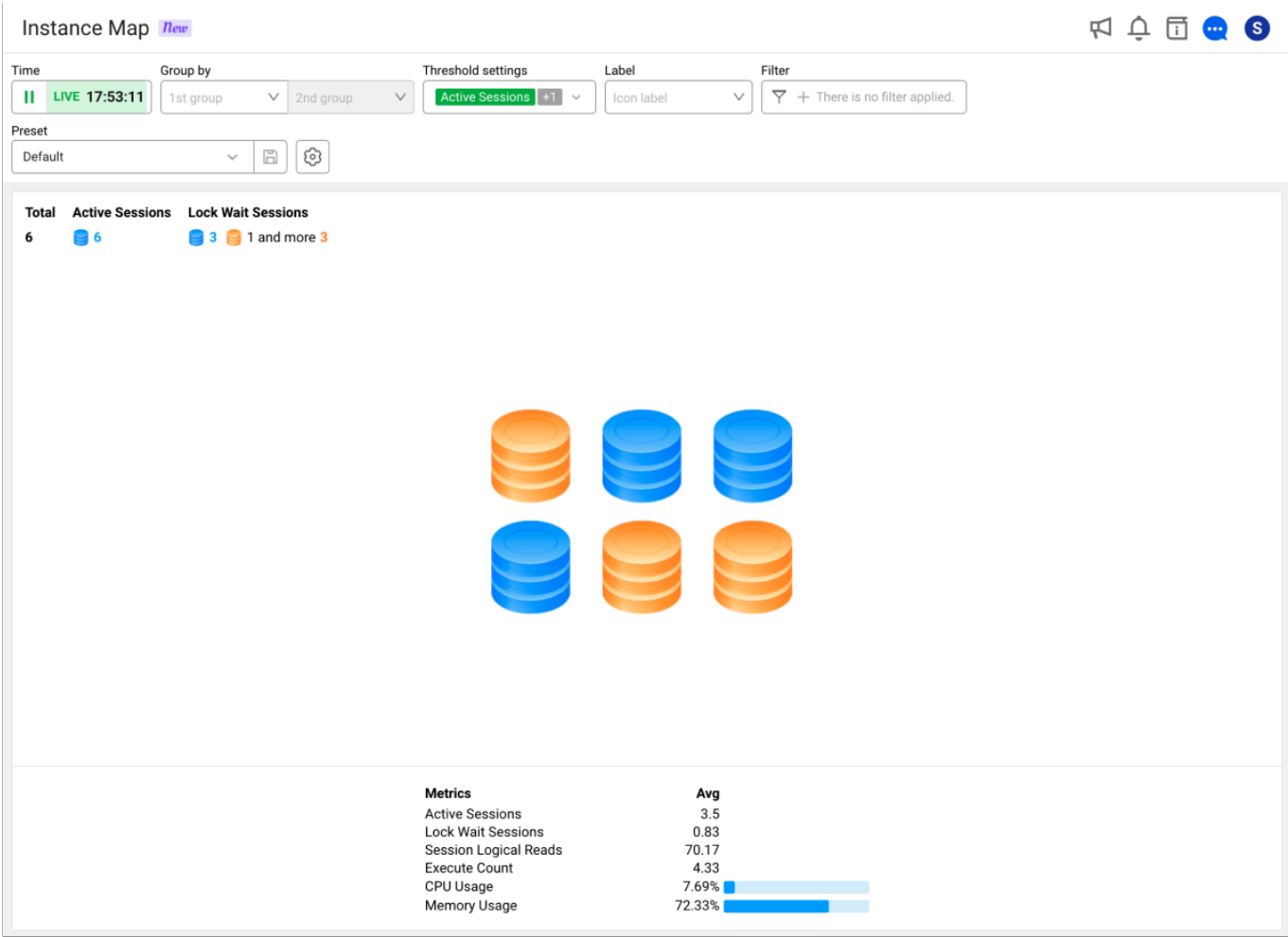
You can display or hide the performance metrics columns in the instance list. Select **Select Column** on the upper right of the screen. Check or uncheck the desired items. You can display or hide the columns.

Instance Map

Home screen > Select Project > **Dashboard** > **Instance Map**

The **Instance Map** visualizes the status of monitored database instances, making database monitoring easier to understand and access.

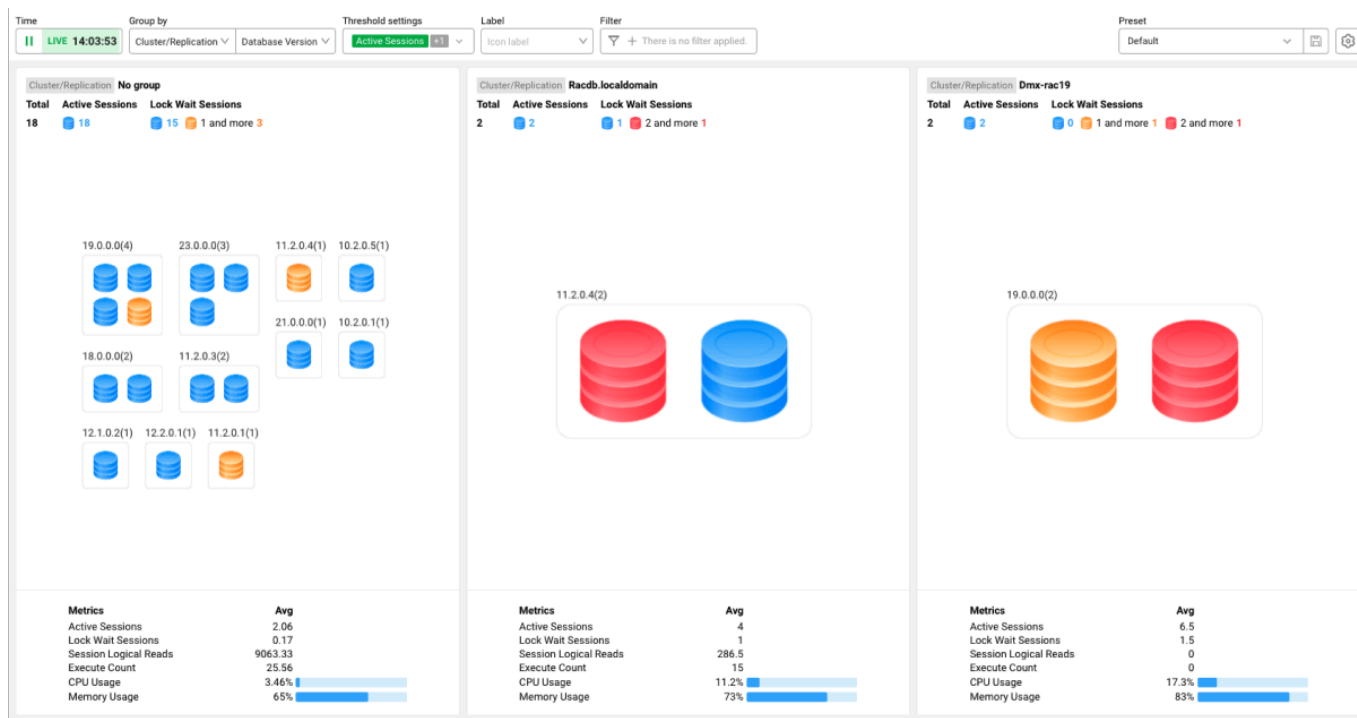
By setting thresholds on key metrics of database servers, you can grasp the status of each instance at a glance and monitor active sessions in real time. You can also navigate to screens such as Instance Monitoring or Multi-Instance Monitoring to quickly analyze databases when an issue occurs.



! To collect process usage data such as CPU and memory usage of a database server, run the XOS agent.

Default screen

The **Instance Map** consists of a **User Settings** section and the **Map Chart** section.



User Settings

At the top of the **Instance Map**, you can configure **Grouping**, **Thresholds**, **Labels**, **Filters**, **Presets**, and **Additional Settings**.

Grouping

You can group instances by various criteria or custom rules. Grouping supports up to two levels. When clicking on a group at the second level, a detail view appears, showing aggregated data for all instances within that group.

Threshold

By setting thresholds for key database monitoring metrics, you can visualize the instance status with color indicators. You can configure one or more metrics and specify **Warning** and **Critical** ranges for each. The **Normal** range is displayed in **blue**. The color applied will reflect the range the value belongs to.

1. At the top of the screen, click **Threshold**.
2. Select the desired metric and set the ranges for Warning and Critical.
3. To add more metrics, click + **Add Metric** at the bottom.
4. Once you finish configuring the thresholds, click **Save**.

Priority of Instance Status Display

The **Critical** range has the highest display priority. If any of the selected metrics fall under **Critical**, the instance will be shown in red. In the example below, both instances are displayed as **Critical**.

Label

Displays text labels inside the instance icons. Labels can include database information and key metrics.

Filter

You can specify filters to monitor only specific instances.

- **Filter keys:** Use database and instance information as filter keys.
 - Collected columns: instance name, cluster/replication info, agent version, CPU core count, DB type, installation location, version, IP, Port
- **Conditions:** Filters support numeric conditions such as $\geq$, $\leq$, $>$, $<$, and string conditions such as match, mismatch, contains, and does not contain.

Preset

You can save configured options as presets and load them as needed. Presets are saved on a per-project basis. This allows project members to monitor instances with the same configuration. A default preset is provided and cannot be deleted.

✓ Default Preset

The default preset in the instance map includes the following thresholds for `active_sessions` and `lock_wait_sessions` :

- `active_sessions`
 - **Warning**: Value ≥ 20
 - **Critical**: Value ≥ 30
- `lock_wait_sessions`
 - **Warning**: Value ≥ 1
 - **Critical**: Value ≥ 3

Additional Settings

Click the  icon at the top-right of the Instance Map to configure additional settings.

- **Show 'No Group'**: Option to display instances that don't belong to any group when grouping is applied
- **Show Metric Summary**: Option to display summary of key metric values at the bottom of groups

Instance Map Chart

The Instance Map chart consists of three main areas.

Top Section

Displays the total number of instances or, if grouped, the total of grouped items and the status counts of selected metrics.

Instance Visualization Section

A visualized dashboard that helps you intuitively understand the overall status and number of all instances.

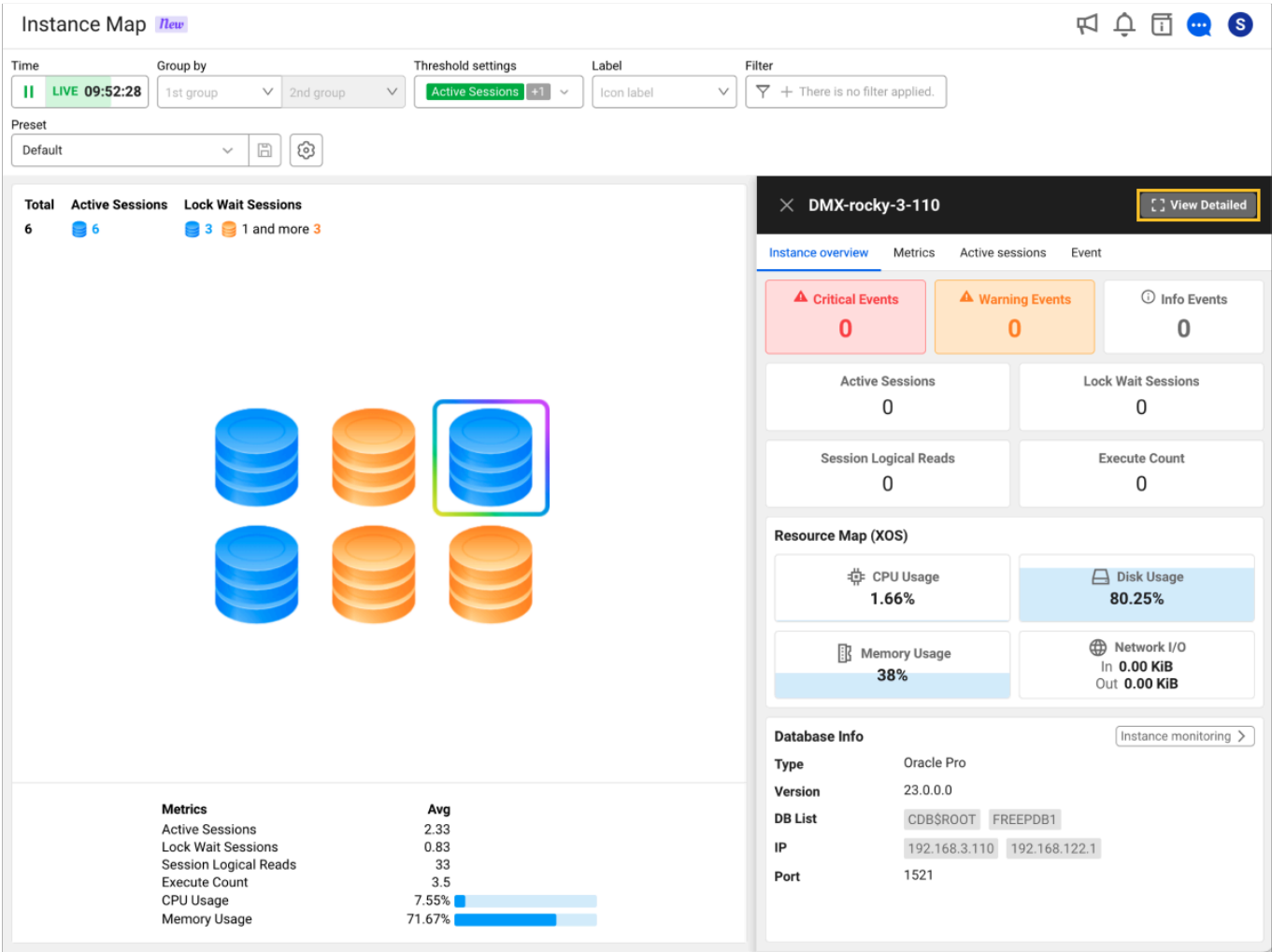
Bottom Section

Displays the average values of key metrics across instances. You can toggle this on or off using the **Show Metric**

Summary option in .

Summary View and Detailed Analysis

Select an instance you want to examine in the **Instance Map**. A summary view appears on the right where you can check **Instance Overview**, **Metrics**, **Active Sessions**, **Events**, and **Logs** tabs. Click the **Detailed Analysis** button to move to the detailed view screen.



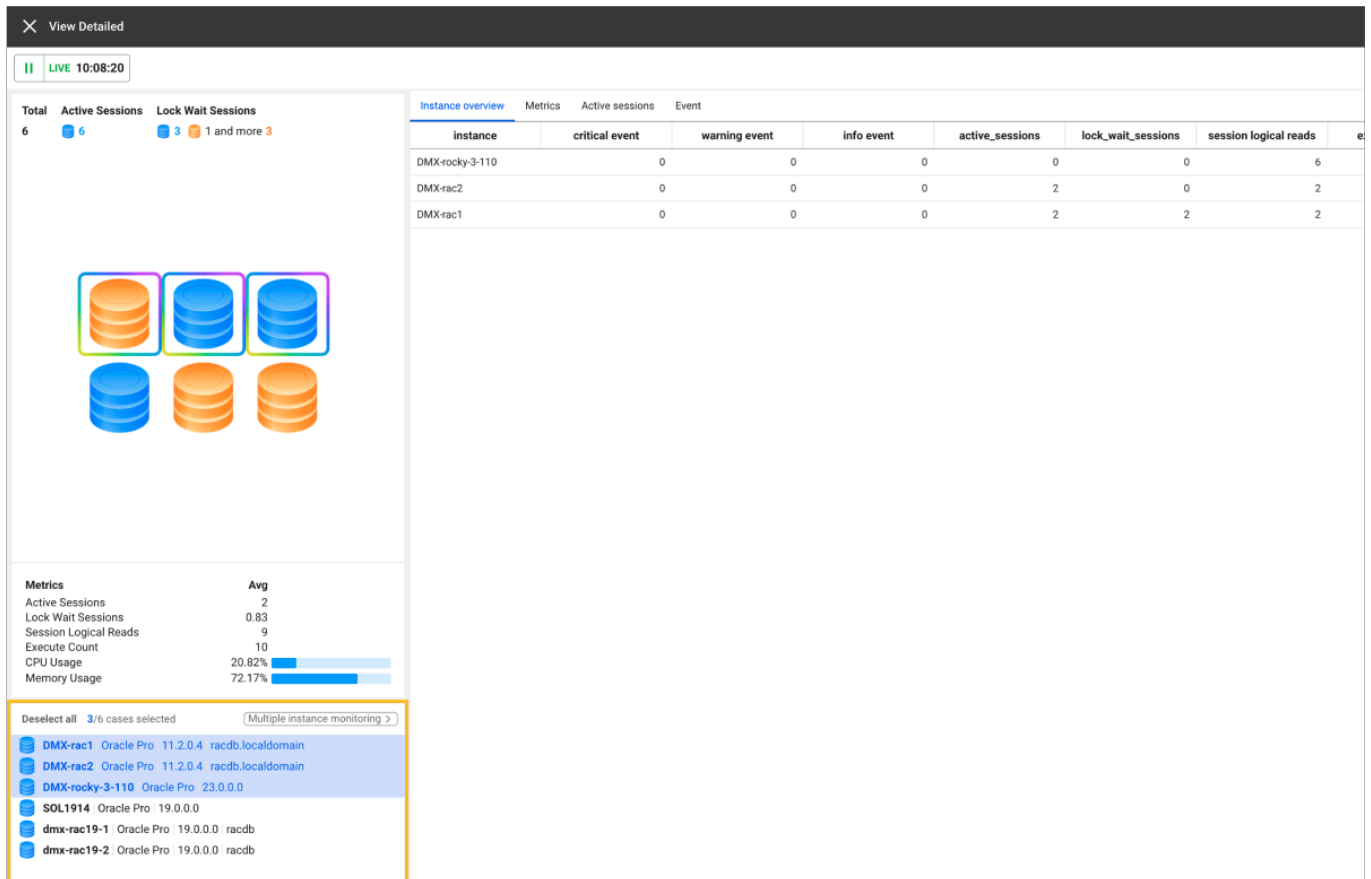
Instance Overview

In the **Instance Overview** tab of the detailed view page, you can check the monitored database information, metrics, and triggered events. You can also navigate to the Instance Monitoring menu. If the instance is inactive, the view links to monitoring at the time of deactivation.

- **Events Section:** Provides information about events triggered based on the user-defined alert rules. Click on each event card to view event details.
- **Metrics Section:** Shows real-time status of major database metrics and resource usage. The resource usage section is available if the XOS agent is configured.
 - **Database Metrics**
 - Active Sessions: Number of sessions currently executing queries
 - Lock Wait Sessions: Number of sessions waiting for locks
 - Session Logical Reads: Number of data blocks read from memory (sum of db block gets and consistent gets)
 - Execute Count: Total number of SQL statement executions
 - **Resource Usage (Resource Map via XOS)**
 - CPU Usage
 - Disk Usage
 - Memory Usage
 - Network I/O
- **Database Info Section:** Displays detailed database information. You can click **Instance Monitoring** to navigate to the corresponding monitoring screen for the selected instance.

Detailed View of Instance Overview

In the **Instance Overview** tab of the detailed view screen, you can monitor major metrics and database information in a table format by selecting multiple instances. Click the **Detailed View** button at the top right of the **Instance Overview** tab, or click the second-level group name in the visualization area after setting the second-level group to access the detailed view screen.



- In the **Detailed View** screen, click the instance icon in the visualization area or the instance list in the bottom-left corner to select or deselect instances. You can also click **Select All** or **Deselect All** to toggle all instances.
- If you select multiple instances, click the **Multi Instance Monitoring >** button to switch to the **Multi Instance Monitoring** screen for the selected instances.

Metrics

In the **Metrics** tab, you can check the metric trends of selected instances. Use the time selector to view real-time or historical trends.

Active Sessions

The **Active Sessions** tab displays the currently running sessions and their queries.

ⓘ In the table, text color indicates query latency: black → orange → red in increasing order of slowness.

Events

In the **Events** tab, you can view events that occurred within the past hour. Clicking an event item will navigate to the instance monitoring screen at the time of the event.

Logs

The **Logs** tab lets you view log entries and live logs. Click the **Detailed View** button on the right to see more detailed log information.

Monitoring A Database Instance

Home > Select Project > **Dashboard** > **Monitoring a Database Instance**

In **Monitoring a Database Instance**, you can monitor the key metrics of the database server, information details on active sessions, lock tree, PQ tree, and process information in real time. You can also see the alerts that could be problematic warnings. By default, the collected data for the last 10 minutes is retrieved in real time. Collected data is updated every 5 seconds.

Basic screen guide



1 Selecting the lookup time and target

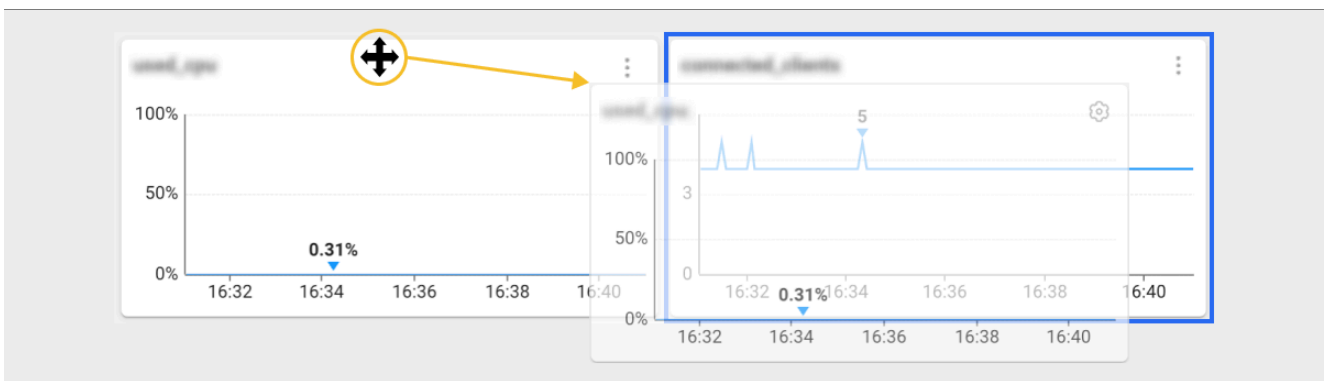
By default, the collected data for the last 10 minutes is retrieved in real time. If you select values in **Time** and **Instance**, the collected data is automatically applied to the **2** graph chart and **4 Active sessions** table.

- To view the past data, click the green button in **Time** and then select the desired lookup time. You can search data up to the last 3 weeks.
- To query a random date and time, select the **||** button in **Time**. If you click the text range for date and time, the option appears to select the date and time. To search data in real time again, select .
- To search data from other agents, click **Instance** and then select a desired agent name.

2 Graph chart section

3 Default displays 16 metrics saved as presets as a graph chart widget. The horizontal axis of each widget is time, and the vertical axis is the metric number. To know the meaning of each metric, select  next to the name or see [the following](#).

- To change the position with another widget, select and drag the upper part of the widget. However, the size of the widget cannot be changed.



- If you stop the **real-time query** mode of the time and click a specific time on the chart in the widget, you can check the session performed at the selected time in the **4 Active sessions** table. After that, a time bar that allows you to view the data every 5 seconds appears on the **Active sessions** table.

- ⓘ
 - The selected time can be seen on the upper right of the **Active sessions** table.
 - In the **Active sessions** table list, the text colors are changed black → orange → Red, which means that the performance of the session is getting slower.

- After stopping the real-time query mode, you can download the chart data as a CSV file by clicking the  button on the upper right of the widget.

- **Top Stat:** If the button displayed in the widget is selected, the top statistics about the user commits are provided. It displays the session information and key performance metrics for transaction commits occurred on the monitoring instance.
 - You can check whether transactions are concentrated in a specific **sid** or **module**.
 - The **program** column allows you to determine through which process a specific task was performed.
- **Top Event:** If a button displayed in the widget is selected, the information about the major event waits for the metric is provided. This data can help you determine the cause of performance degradation and diagnose problematic sessions.
 - You can identify bottlenecks by checking sessions with high **time waited** values.
 - You can check whether wait events are concentrated in a specific **module** or **program**.

! The **Top Stat** or **Top Event** button may not be available depending on the provided metrics.

4 Active Sessions / Lock Tree / Process Info

You can search active sessions, lock trees, and processes information in real time. If you select a specific time of the chart in the widget, the **real-time inquiry** mode is stopped and the session performed at the selected time appears in the table. A time selection button appears above the table to search data in 5-second increments.

- **Active sessions:** You can view active sessions and queries.
 - If you select a session running, the **Session detail** window appears. You can see the details for the selected session. For more information, see [the following](#).
 - If you select an item in the rightmost **query_text** column in the table, the **SQL details** window appears. You can see the full text and plan information for the selected query. For more information, see [the following](#).

- **Lock tree**

You can see the lock session information, and holder and waiter information in real time. For more information about the table columns, see [the following](#).

- **Process Info**

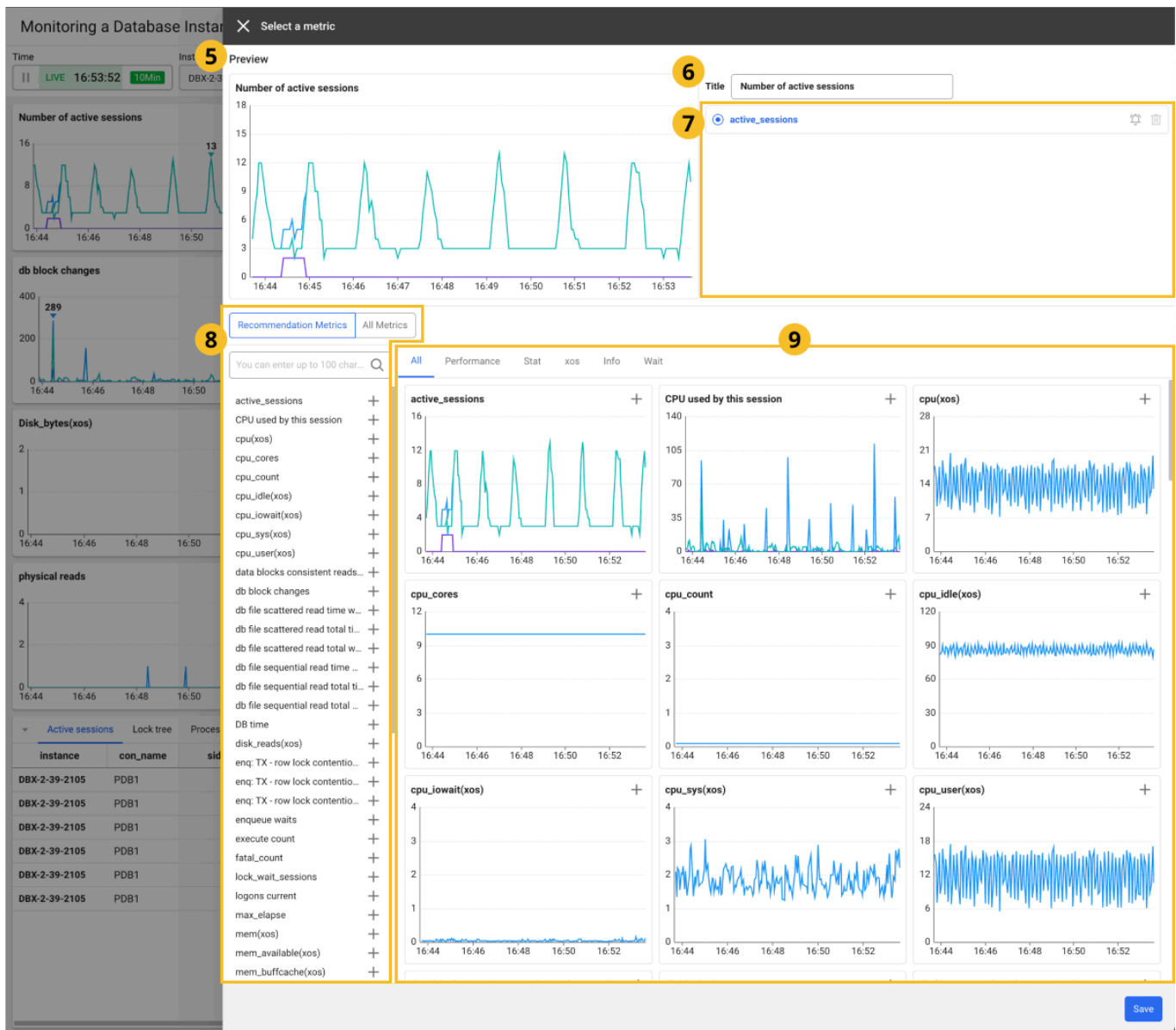
You can query the resource usage of the database server in real time.

✓ By adding a linked project, you can view information details about active transactions associated with active sessions in the **Active sessions** table in real-time. For more detailed information about adding and using a linked project, see [the following] (linked-projects).

ⓘ In the table list, the text colors are changed black → orange → Red, which means that the performance of the session is getting slower.

Selecting metrics

To add or replace the metric graph to the widget, select ⋮ on the upper right of the widget. The **Select a metric** window appears. After completing the desired settings, select **Save**.



- **5 Preview:** You can preview the graph for the selected metric.
- **6 Title:** Enter the widget name.
- **7** It displays the metrics displayed on the chart of the selected widget. You can select up to 4 metrics.

ⓘ When you select  on the right of a metric, the **Event Configuration** *new* menu appears, where you can configure the alerts based on the metric.

- **8 Recommendation Metrics / All Metrics:** If you select **Recommendation Metrics**, the list of frequently viewed metrics in Database Monitoring appears. The metrics displayed in the list can be changed or added in the widget.
- In the **9** area, you can see the trend of metrics in a graph and add or change any widget. By selecting a desired tab, you can narrow down the list of metrics according to the category.

Adding metrics

Multiple metrics can be added in a widget. You can add up to four metrics.

1. Select  on the upper right of the widget to add the metrics in.
2. If the **Metric Select** window appears, in the **8** metrics list or **9** graph list, select  on the right of the metric to add.
3. To add more metrics, repeat the step 2.
4. To easily identify the widget, change the **6 Title**.
5. Select **Save** on the lower right of the screen.

Charts are generated as many times added to the widget to which metrics are added. If you select  next to the widget name, you can see the name and description of the added metric. You can also check the figure through a tooltip by hovering your mouse over the chart.

 The tooltip of the  button is not supported in multiple languages.

Changing a metric

You can change a metric in the widget to another.

1. Select  on the upper right of the widget to change the metrics in.
2. If the **Metric Select** window appears, select a metric to change in the **7** area.
3. In the **8** metrics list or **9** graph list, select a desired metric.
4. In the **7** area, check that the name of the selected metric has been changed.
5. Select **Save** on the lower right of the screen.

Deleting a metric

You can delete unnecessary metrics while multiple metrics have been set in the widget.

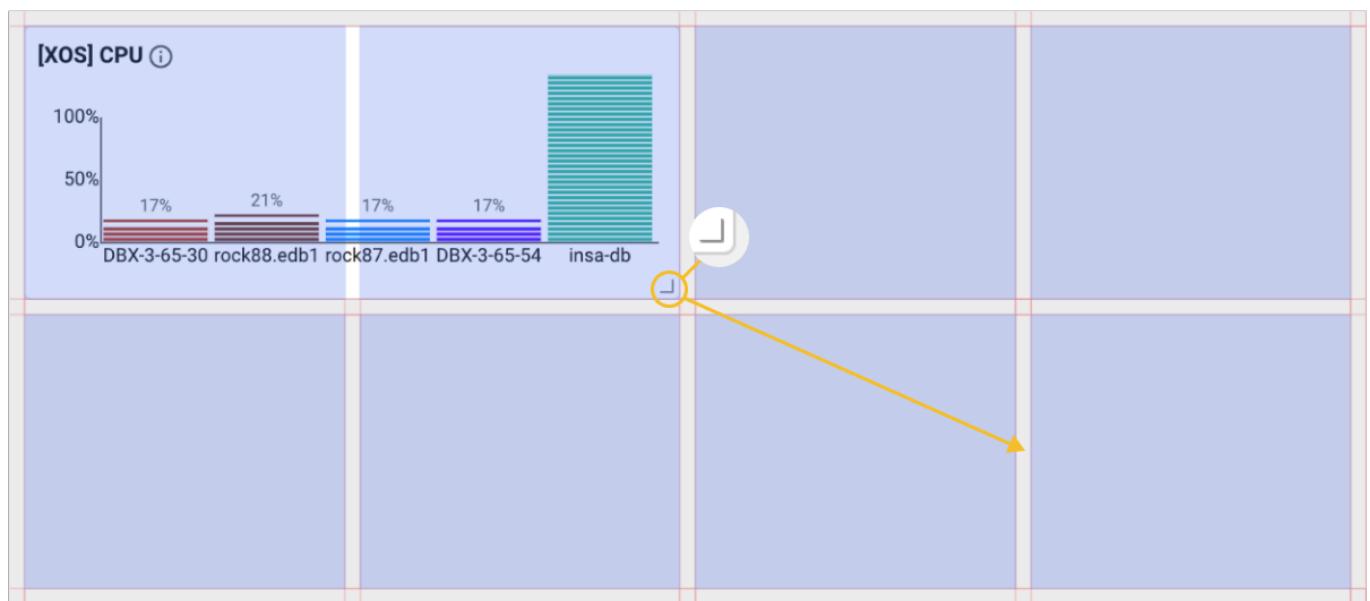
1. Select  on the upper right of the widget to delete the metrics from.
2. If the **Metric Select** window appears, select  on the right of the metric to delete from the **7** area.
3. Select **Save** on the lower right of the screen.

 At least one metric must be selected for a widget. You cannot delete all metrics in a widget.

Editing a dashboard widget

The widgets on the dashboard can be adjusted to the desired size and placed in the desired positions. You can also delete unnecessary widgets or add them again.

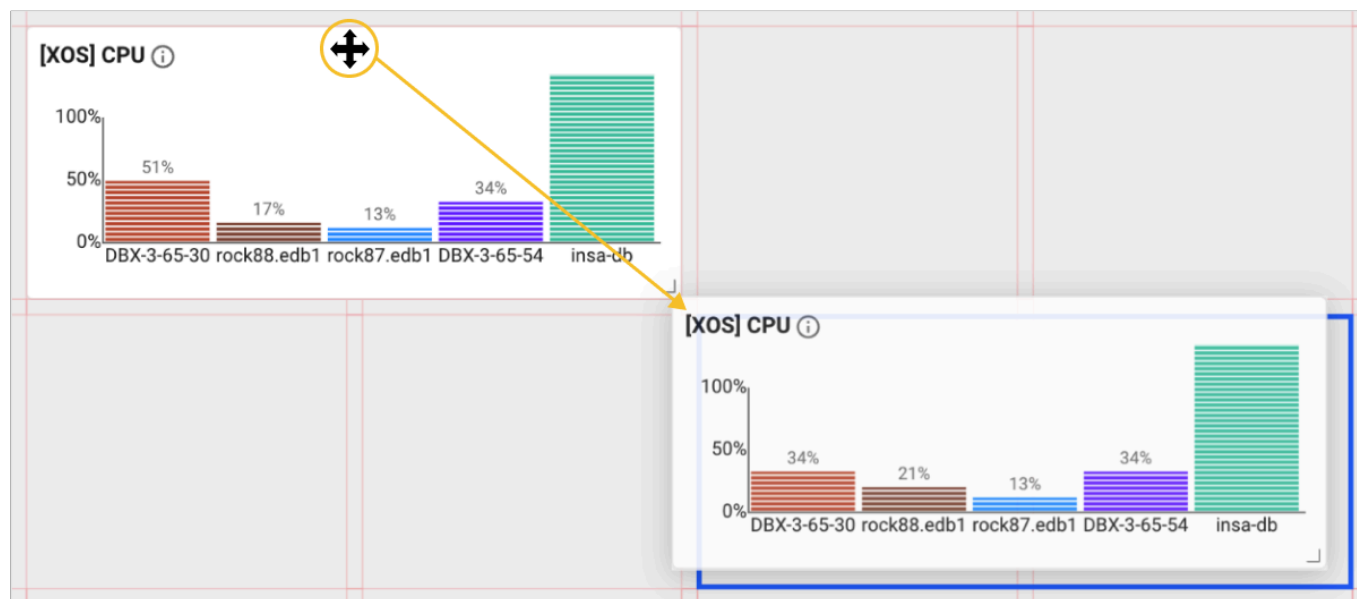
Resizing a widget



Click and hold the mouse on the  element at the lower right of the widget, and then drag it to the desired size.

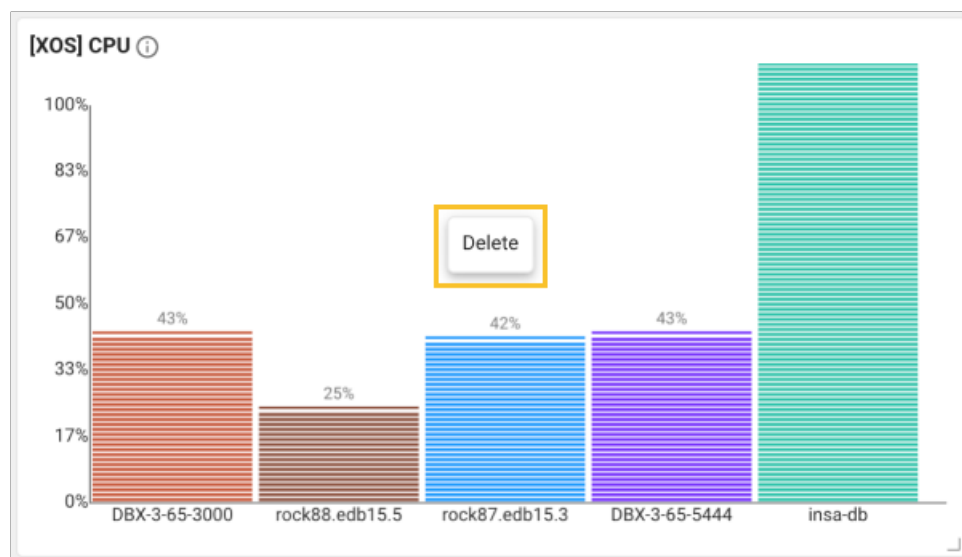
Grids with a uniform horizontal-to-vertical ratio appear and the size of the widget can be adjusted in each grid.

Moving a widget



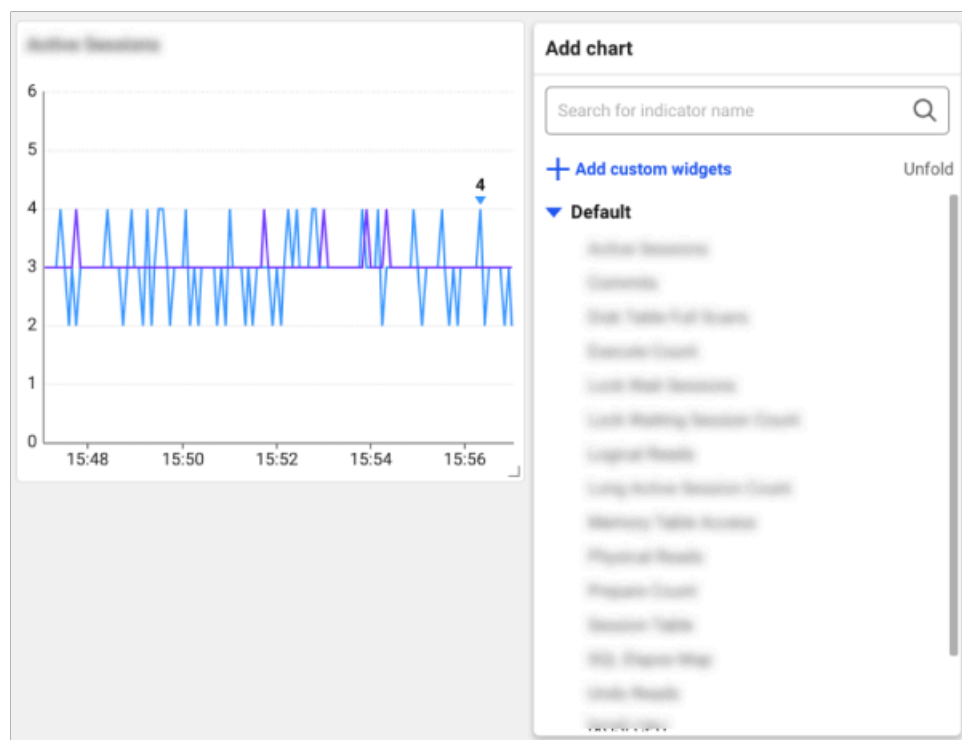
When you move the mouse cursor to the top of the widget, the cursor shape changes to **+**. At this time, you can move the widget by dragging with your left mouse button and move to the desired position.

Deleting a widget



Right-click on a widget to delete. If you select **Delete**, the widget is deleted from the dashboard.

Adding a widget



Move the mouse cursor to an empty space on the dashboard and then right-click on it. Select a widget to add from the pop-up menu. Place the widget to the desired position and then resize it.

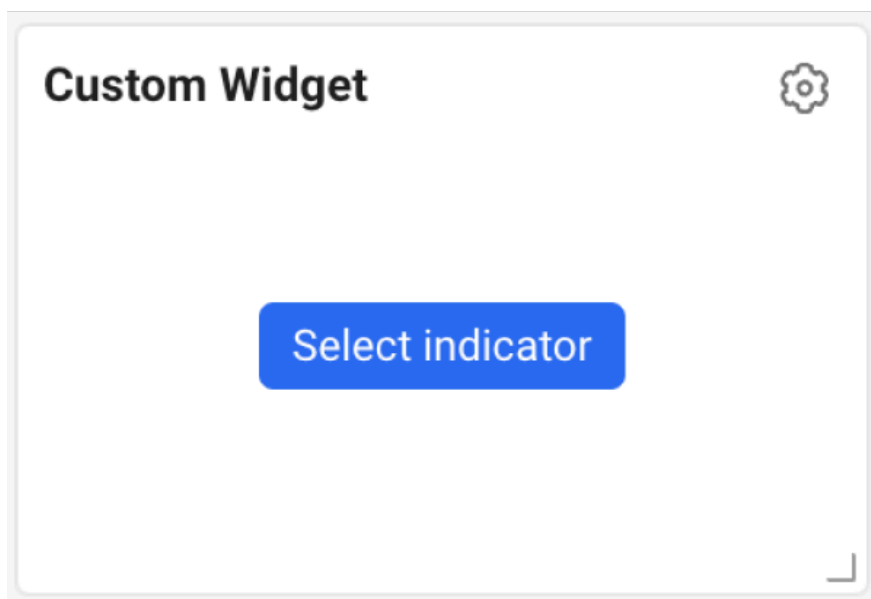
Custom widget

The custom widget feature allows users to add their own metrics in addition to the metrics provided by default on the dashboard. If you have metrics to continuously monitor on the dashboard in addition to the metrics provided by default, use the custom widget feature.

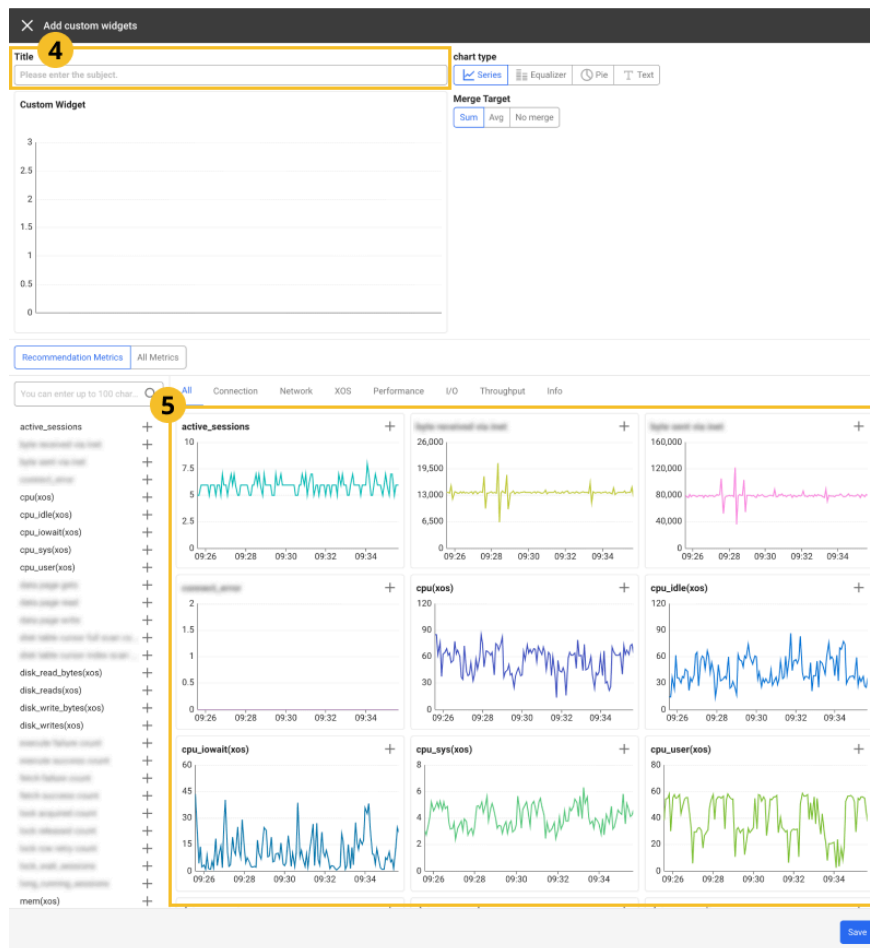
Add

1. Right-click an empty space on the dashboard.
2. On the pop-up menu, select **Add Custom Widgets**.

3. If **Custom Widget** is created on the dashboard, select **Select indicator**.



4. When the **Add Custom Widgets** window appears on the right of the screen, enter the widget name in **Title**.



5. In the metrics list at the bottom of the screen, select the + button on the upper right of the metric to add.
6. After the metric has been changed, select **Save**.

- ! • You can add up to four metrics. To add metrics, select +.
- To change the metric, select a different metric in the metrics list. If you select +, the metric is added.
- If you do not enter the widget title, it is saved as a custom widget.

Selecting the chart type

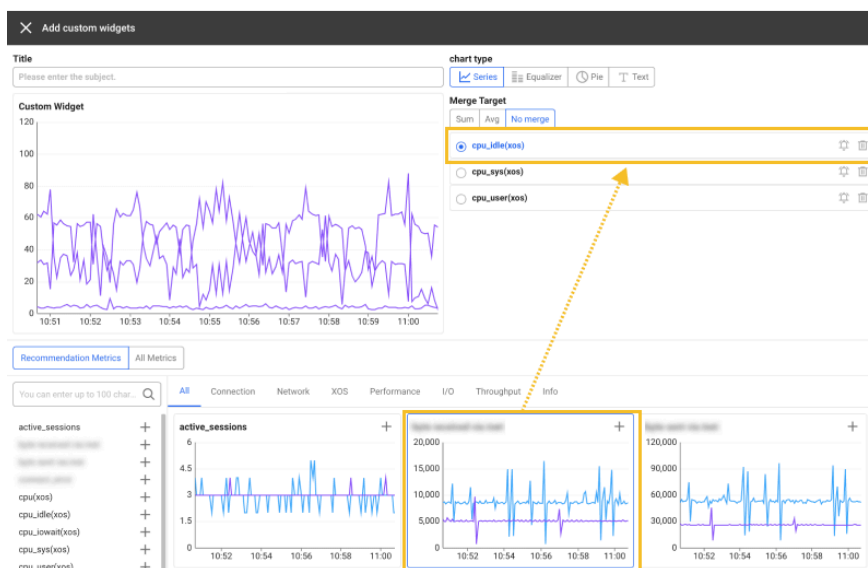
In **chart type** of the **Add Custom Widgets** window, you can select the following types.

-  **Series**: Time series chart that displays the metric changes over time.
-  **Equalizer**: Bar graph chart that can compare the performance for multiple metrics. By visually displaying various metrics on one screen, you can easily see the overall performance status.
-  **Pie**: Pie chart useful for checking the composition ratio for multiple metrics relative to the whole.
-  **Text**: It displays the figures for the metrics in text.

Changing a metric

You can change or add metrics for the metrics added to the custom widget.

1. Select  on the upper right of the custom widget added on the dashboard.
2. When the **Add Custom Widgets** window appears on the right of the screen, select a metric to change.



3. Select a metric to change from the metric list at the bottom of the screen. To add a metric, select **+** on the upper right of the metric to add.
4. After the metric has been changed, select **Save**.

- ✓ The features of the icon buttons in the list for added or changed metrics are as follows:
- : You can exclude the metric from the custom widget.
 - : You can set the alert for the metric. When the button is selected, the **Alert > Event Configuration** appears. For more information about the **Metrics event** setting, see [the following](#).

Widget Options

The icons displayed on the widget function as follows:

- : You can see the features and information for major widgets. (multilingual support planned)
-  or : You can enlarge or reduce the range of the vertical axis.
- : You can view the widget's data on a wider screen.
- : A detail window appears where you can search the data of the widgets separately by agent.

ⓘ The provided options may differ depending on the widget.

Preset

You can save and recall user-defined widget settings and active session table arrangements as presets. The **Default** preset cannot be modified.

Preset

Default



Save Preset

In the **Preset** list, select a saved item to fetch the settings.

1. Add or modify the widget metrics.
2. On the upper right of the screen, select .
3. In **Save preset**, change the name and then select **Save**.

- The **Default** preset cannot be modified.
- Depending on the database environment, some presets may not be supported.
- Including the modified metric graph, the column order, column width, and filter application content of the active sessions are also applied for custom presets.
- You can save or modify the presets with the **Edit project** role. For more information about the member roles, see [the following](#).

Creating a preset

1. Place the widgets in the desired format on the dashboard. You can also resize and place only the widgets to be frequently checked.
2. On the upper right of the screen, click the  icon.
3. Enter a new preset name.
4. Click **Save**. You can see the newly saved preset from the preset list.

- If any changes are made to the newly created preset, save the preset again. Click the  icon and then save the preset with the same name. Any changes are overwritten in the existing preset.
- If you go to another menu without saving changes on the dashboard, the changes are not saved.
- Presets are saved on a project basis and can be shared with other users.

Deleting presets

If you select **Preset** on the upper right of the screen, the preset list appears. Select  on the right of the preset to delete.

Using the active session section

Through **Active sessions** at the bottom of the screen, you can see the real-time active sessions. The following provides the guide for the button on the upper right of the table.

- : Using the column header entries in the table, the list can be filtered.
- : The list of tables is refreshed.
- : The column header entries in the table can be displayed or hidden.

 For more information about the column configuration.

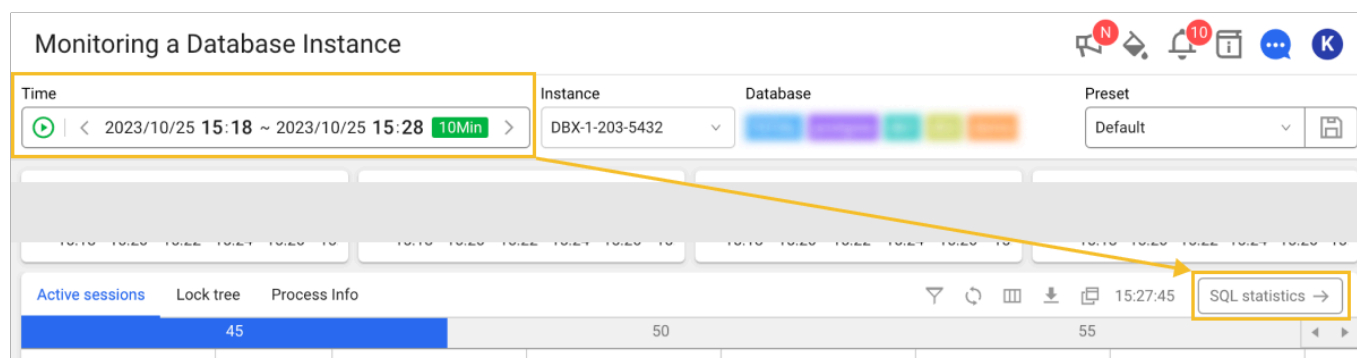
- : The active session can be stopped.

 This feature is only available to the members with **Edit** role. For more information about member roles, see [the following](#).

- : You can download the content of the table as a CSV file.
- : A new window appears to see the list on a wider screen.

Checking the SQL statistics

You can see the list of top SQLs for the time period along with the trend of key performance metrics that occurred during the specific time in the past.



1. In **Time**, select **||** to switch from real-time mode to non-real-time mode.
2. In **Time**, set the time zone to view.
3. On the upper right of the **Active sessions** area, select **SQL statistics**.

SQL statistics appears in a new window. In **Instance monitoring**, you can see the top SQL list executed for the set time. For more information about **SQL statistics**, see [the following](#).

Checking the session information details

If you select a session in the **Active sessions** tab, the **Session detail** window appears. You can see the information details about the selected session.

Active sessions

Lock tree

Process Info

instance	con_name	sid	serial#	cpu(xos)	pga memory	username	last_call_et	event	query_text	sql_id	session logical...
DBX-2-39-2105	PDB1	422	46285	0.76%	0	KWLEE	3	enq: TX - row lock co...	update kwlee set b='111' where a=x	9xn1nukt0jcw7	0
DBX-2-39-2105	PDB1	212	29166	0.00%	0	KWLEE	23	PL/SQL lock timer	declare aa number; begin aa := 1; execute imm...	54qb6k9mf4n99	0
DBX-1-97-1521		17	7023	0.00%	0	WHATAP	0	SQL*Net message to ...	select /* WhaTap3D#4 */ 0 as con_id,pl.plan_h...	12zmp1v34h271	0

Session detail (sid: 17, serial#: 7023)

||

LIVE

10:43:04

↺

↻

⛔ Kill Session

🔗

Session history >

Session Stat

ALL

9M

7.5M

6M

4.5M

3M

1.5M

0

10:39

10:40

10:41

10:42

10:43

session logical reads

physical reads

execute count

session pga memory

db block changes

parse count (hard)

Delta

Sigma

Event	Total Waits	Total Timeouts	Time Waited
Disk file operations I/O	0	0	
control file sequential read	0	0	
latch: cache buffers chains	0	0	
buffer busy waits	0	0	
read by other session	0	0	
dh file sequential read	0	0	

Stat	Value
logons cumulative	4
logons current	0
opened cursors cumulative	1
Requests to/from client	0
user commits	7
recursive calls	5
recursive cpu usage	0
pinned cursors current	0
user logons cumulative	0

Session Info

username	WHATAP	status	1
schemaname	WHATAP	command	3
spid	5964	last_call_et	0
osuser	whatap	sql_id	12zmp1v34h271
machine	dbx-agent	sql_hash_value	3326609633
program	JDBC Thin Client	plan_hash_value	57415992
blocking_session	0	module	JDBC Thin Client
row_wait_block#	130	action	
row_wait_obj#	-1		

Wait Info

seq#	23770	p1	1952673792
event#	431	p2	1
event	SQL*Net message to client	p3	0
wait_class		wait_time	-1
state(xos)	none		

SQL Text

Plan

Default Bind

Reset Bind

Formatting

```
select /* WhaTap3D#4 */ 0 as con_id,pl.plan_hash_value, pl2.plan_hash_value
prev_plan_hash_value,saddr,s.sid,s.serial#,s.username,command,taddr,lockwait,s.status,
osuser,p.spid,process,machine,s.terminal,s.program,type,sql_address,s.sql_hash_value,s.
sql_child_number,prev_sql_addr,prev_hash_value,prev_sql_id,
prev_child_number,s.module,s.action,client_info,row_wait_obj#,
row_wait_file#,row_wait_block#,row_wait_row#,logon_time,last_call_et,pdml_status,
pddl_status,pq_status,client_identifier,blocking_session_status,
blocking_instance,blocking_session,
seq#,event#,event,p1,p2,p3,wait_class#,wait_time,
seconds_in_wait,service_name,
x.xidusn,x.used_ublk,x.used_urec
,plsql_entry_object_id,plsql_entry_subprogram_id,plsql_object_id,plsql_subprogram_id
,port,sql_exec_start,sql_exec_id,prev_exec_start,prev_exec_id,final_blocking_session_stat
,p.cpu_used
from v$session s,v$process p,v$transaction x ,v$sql pl,v$sql pl2
```

ⓘ For more information about the **Session detail** window.

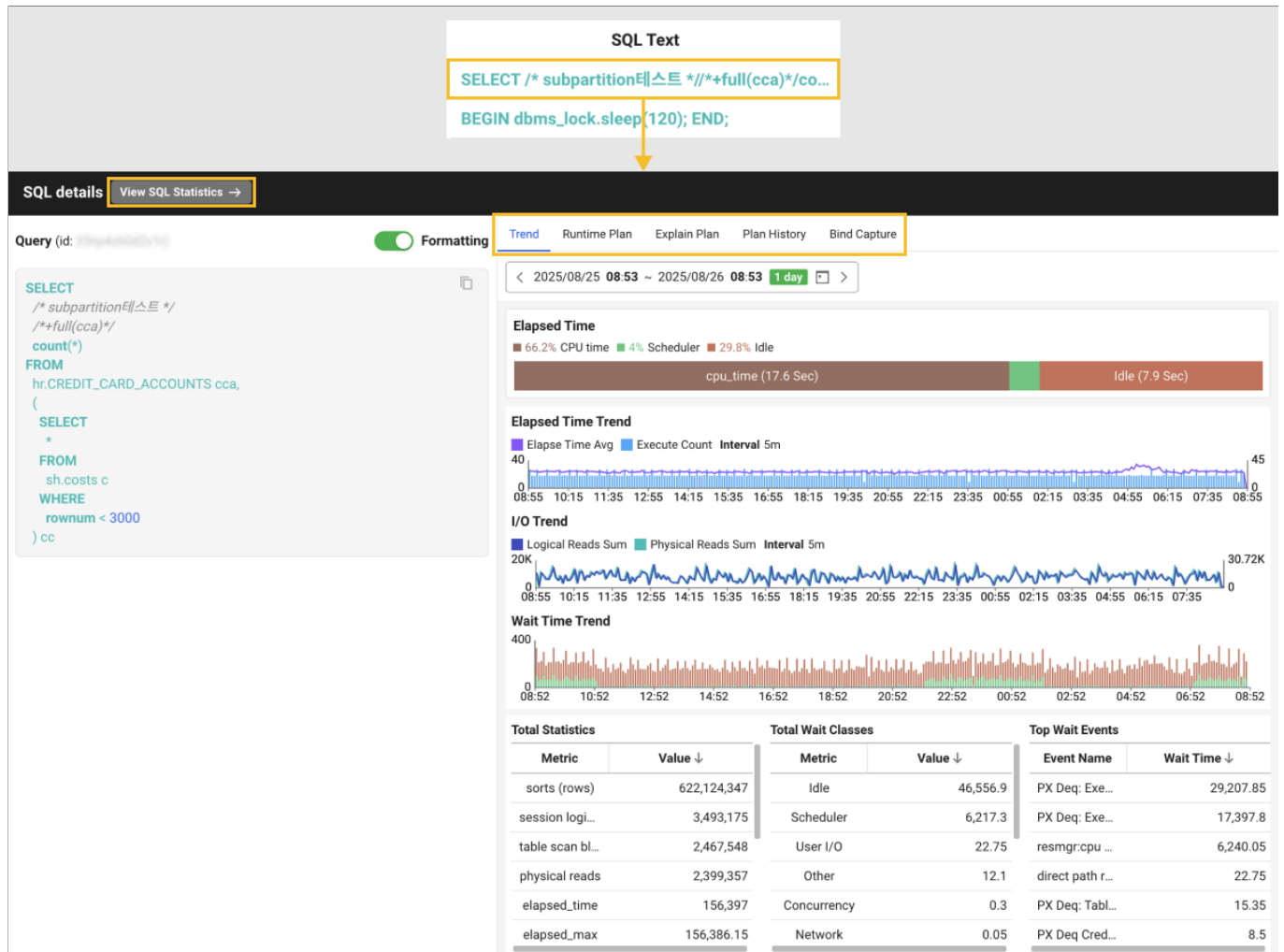
77

[See query details](#)

To check the SQL query information of the current or past session, refer to the **SQL_Text** column item.

The **SQL details** window will appear, showing the SQL query and plan information.

By clicking the **View SQL Statistics** → button, you can move to the **SQL statistics** menu to check the statistical information related to the SQL query.



The tabs in the **SQL Details** window are as follows:

- **Trend:** Provides a comprehensive analysis of the execution time, I/O, and wait events of a specific SQL, enabling quick identification of performance bottlenecks.
- **Runtime Plan:** Displays the execution plan and runtime information of the selected SQL query, including

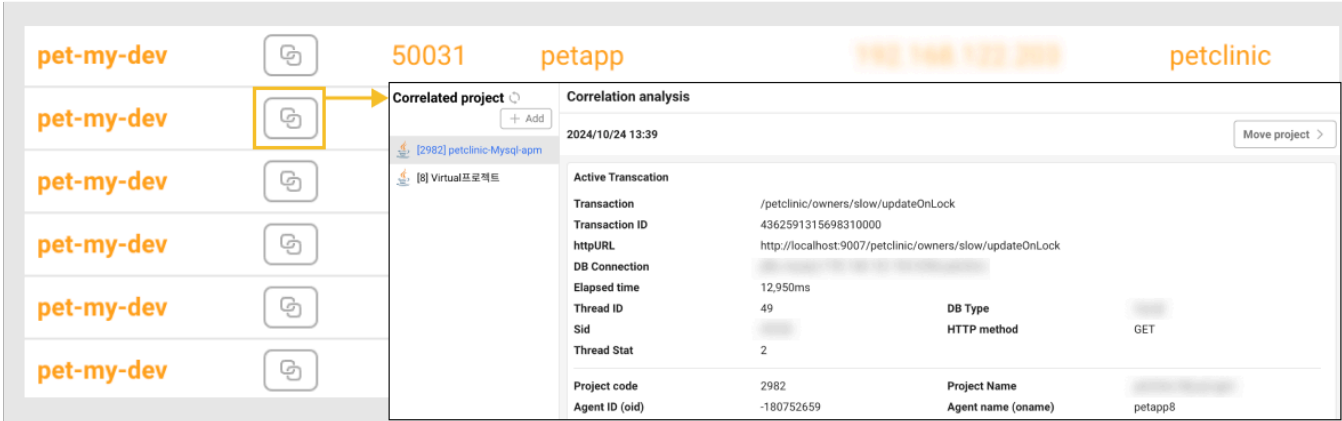
execution count, average execution time, and average physical reads.

- **Explain Plan:** Shows the execution plan predicted by the optimizer, including cost, operation, object name, and cardinality.
- **Plan History:** Displays the execution plan history of SQL queries executed in the database.
- **Bind Capture:** Shows the values of bind variables used in SQL queries executed in the database, allowing you to understand the actual content of query execution.

- For more information about the **SQL details** window.
- You can see the tuning guide after selecting the  button in the Plan tree. The tuning guide can be viewed when the nodes in the Plan Tree are `TABLE ACCESS (FULL)` and `INDEX (SKIP SCAN)` and the nodes contain the keyword, `CARTESIAN`.

Checking the linked projects

In **Correlated Project Management**, when another application project is added, the  button is added to the active session entry.



Active Transaction			
Transaction	/petclinic/owners/slow/updateOnLock		
Transaction ID	4362591315698310000		
httpURL	http://localhost:9007/petclinic/owners/slow/updateOnLock		
DB Connection	[REDACTED]		
Elapsed time	12,950ms		
Thread ID	49	DB Type	[REDACTED]
Sid	[REDACTED]	HTTP method	GET
Thread Stat	2		
Project code	2982	Project Name	[REDACTED]
Agent ID (old)	-180752659	Agent name (oname)	petapp8

Through the **Correlation analysis** window that appears when the  button is selected, you can check active transactions of the applications linked to active sessions. You can check specific metrics such as transaction ID, thread ID, client IP, and CPU usage time, along with basic information such as transaction URL and database connection information, DB type, HTTP method, and execution time. Additionally, you can monitor the SQL query execution time, number of SQL calls, DB connection time, and such, which allows you to comprehensively analyze the system performance.

In particular, it provides the call stack information and SQL queries through stack traces, helping you trace and resolve detailed causes when problems occur. This allows you to perform in-depth analysis of interactions between applications and databases, and to quickly diagnose performance bottlenecks.

❗ For more detailed information about adding and using a linked project, see [the following] (linked-projects).

Stopping an active session

1. Select  on the upper right of the table.
2. Select the session to stop from the table list. You can select multiple items.
3. Select **Stop**.
4. If the **Session kill** window appears, enter the password ([Parameter Key](#)). You can check the password in the [paramkey.txt] file located in the agent installation directory.
5. Select **Apply**. The selected session is stopped.

- ❗
- Through the [paramkey.txt](#) file in the DBX agent installation path, you can find the password.
 - This function can be used by only the members with the **Edit** role. For more information about the member roles, see [the following](#).

Zooming in the graph chart section

If the current screen is small and difficult to check the widgets, you can see the graph chart widget on a wider screen. Select ▼ at the bottom of the **Active sessions** section. The **Active sessions** section is collapsed and the graph chart widget section is expanded.

To zoom out the graph chart widget, select ▲ at the bottom of the **Active sessions** section.

Setting the table columns

You can hide the table header columns or add any of them. You can also change the column order. Select .

- After configuration, select **Confirm** 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.
- Images may differ depending on the product, project, or menu.

Adding columns

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

Deleting columns

From the **1** list, unselect the columns to delete. Alternatively, select **×** on the right of the item to delete from the **2** list.

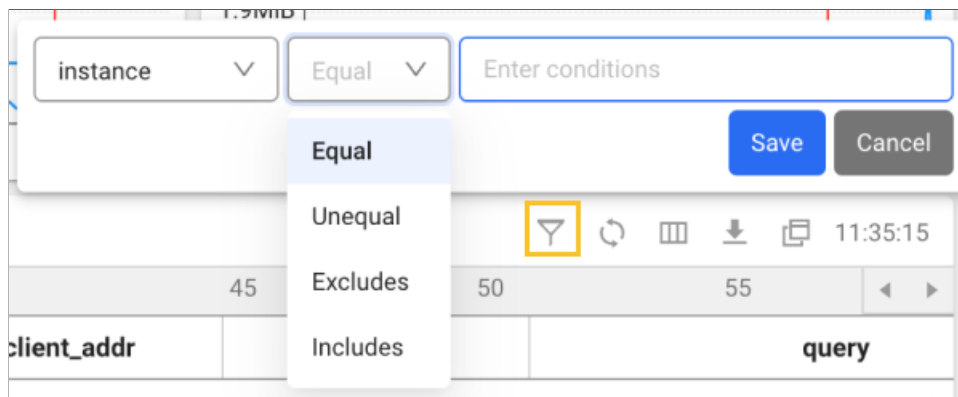
Changing the column order

Drag an item to reposition from the **2** list, and then move it to the desired position.

Initializing the configuration

To cancel all changes and reset them, select **↺ Initialized**.

Filtering the table data



1. Select  on the upper right of the table.
2. Select a column header and a condition in the table.
3. Enter a desired value in the **Enter conditions** field.
4. Select **Save**.

Column information guide

For more information about columns, see [the link](#).

Item	Description
<code>sid</code>	Session ID
<code>cpu(xos)</code>	CPU utilization collected by the installed XOS
<code>serial#</code>	Value to increment to determine if the same session ID has been used.
<code>username</code>	DB user name.
<code>command</code>	Same as <code>COMMAND_TYPE</code> ; SQL command number
<code>status</code>	Session status
<code>schemaname</code>	Schema. Same as the username.
<code>osuser</code>	Username of the OS that accessed the session.
<code>spid</code>	OS process ID.
<code>process</code>	ID of the client process that accessed the session.
<code>logon_time</code>	Session created time point.
<code>last_call_et</code>	If the session is active, it is the elapsed time (sec) since activation.
<code>sql_text</code>	<code>query</code> , running statement, SQL statement that separates literal strings

Item	Description
sql_param	Literal value separated from query
prev_sql_text	Previously executed query
prev_sql_param	Previously executed query param
program	Program information executed by client
type	session type - can be adjusted in user background recursive (whatap.conf).
seq#	Number that uniquely identifies the current or last wait (incremented with each wait)
event	Number of the resource or event if the session is waiting.
p1	First waiting event parameter (decimal point)
p2	Second waiting event parameter (decimal point)
p3	Third waiting event parameter (decimal point)
wait_class	Class name of the waiting event
wait_time	The value is 0 if the session is waiting. It is the last waiting time (milliseconds) if the value is greater than 0.
seconds_in_wait	Time spent waiting for the wait if the session is waiting. Otherwise, it is the time elapsed since the last waiting.
state	Waiting state: WAITING , WAITED UNKNOWN TIME , WAITED SHORT TIME , WAITED KNOWN TIME
machine	Name of the client server that accessed the session (hostname)
port	Client port number

Item	Description
<code>terminal</code>	Client terminal name
<code>module</code>	Name of the module set by the user of the program executed on the client.
<code>action</code>	Action name set by the user of the running module.
<code>client_info</code>	Client information set by user
<code>client_identifier</code>	Client ID set by user
session logical reads	Number of times data was read into the buffer cache
physical reads	Number of data reads from the physical disk
execute count	SQL execution count
parse count (hard)	Total number of syntax analysis calls (actual syntax analysis)
parse count (total)	Total number of syntax analysis calls (hard, soft, describe)
opened cursors current	Number of open cursors
db block changes	Change count of all blocks in the database
session pga memory	Session's current PGA size
<code>undo_segid</code>	Undo information ID
<code>undo_blk</code>	Undo block count
<code>undo_rec</code>	Undo record count
<code>sql_address</code>	SQL statement identification data
<code>sql_hash_value</code>	SQL statement identification data

Item	Description
<code>sql_id</code>	SQL ID of the running SQL statement
<code>sql_child_number</code>	Sub number of the running SQL statement
<code>sql_exec_start</code>	Start time point of SQL running in the session
<code>sql_exec_id</code>	SQL running ID
<code>prev_sql_addr</code>	Information that identifies the last run SQL statement
<code>prev_child_number</code>	Sub number of the last SQL statement executed
<code>prev_exec_start</code>	Start time point of the last SQL statement executed
<code>prev_exec_id</code>	Start ID of the last SQL statement executed
<code>plsql_entry_object_id</code>	Object ID of the top PL/SQL subprogram on the stack.
<code>plsql_entry_subprogram_id</code>	ID of the top PL/SQL subprogram on the stack.
<code>plsql_object_id</code>	Subprogram ID of the running PL/SQL object
<code>taddr</code>	Transaction address
<code>lockwait</code>	lock wait address (lock address where the session is waiting)
<code>row_wait_obj</code>	Object ID of the table containing the row specified in <code>ROW_WAIT_ROW#</code>
<code>row_wait_file</code>	ID of the data file containing the row specified in <code>ROW_WAIT_ROW#</code>
<code>row_wait_block</code>	ID of the block containing the row specified in <code>ROW_WAIT_ROW#</code>
<code>row_wait_row</code>	Currently locked row
<code>pdml_status</code>	If the value is <code>ENABLED</code> , the session is in <code>PARALLEL DML</code> activation mode. If the value is <code>DISABLED</code> , the session does not support the <code>PARALLEL DML</code> activation

Item	Description
	mode. If the value is <code>DISABLED</code> , the session does not support the <code>PARALLEL DML</code> activation mode.
<code>pq_status</code>	If the value is <code>ENABLED</code> , the session is in <code>PARALLEL QUERY</code> activation mode. If the value is <code>DISABLE</code> , the session does not support the <code>PARALLEL QUERY</code> activation mode. If the value is <code>FORCED</code> , it indicates the session was changed to force the use of <code>PARALLEL QUERY</code> .
<code>blocking_session_status</code>	It provides details about whether there is a blocking session.
<code>blocking_instance</code>	Instance ID for the blocking session
<code>blocking_session</code>	Session ID for the blocking session
<code>final_blocking_session_status</code>	It provides details about whether there is a last block session.
<code>final_blocking_instance</code>	Instance ID for the last blocking session
<code>final_blocking_session</code>	Session ID for the last blocking session
<code>service_name</code>	Service name of the session. It is similar to the DB name.
<code>saddr</code>	Session address
<code>con_id</code>	ID of the container associated with the data (only for multi-database support)
<code>rss(xos)</code>	OS metrics collected by the XOS agent
<code>pss(xos)</code>	OS metrics collected by the XOS agent
<code>ioread(xos)</code>	OS metrics collected by the XOS agent
<code>iowrite(xos)</code>	OS metrics collected by the XOS agent

! WhaTap basically stores the client-related information.

Monitoring Multiple Instances

Home > Select Project > **Dashboard** > **Monitoring Multiple Instances**

While you can monitor only one agent (instance) in **Monitoring a Database Instance**, **Monitoring Multiple Instances** allows you to monitor and manage multiple agents (instances).

You can compare and analyze key performance metrics occurring in each instance at a glance. You can effectively perform monitoring by aggregating data such as the number of agents running, the sum of all active sessions, and the number of transactions. Users can easily check the changes in metric values over time and high performance metrics for each instance through graph charts.

Additionally, users can see the longest-lasting active sessions, which can be used as a basis for diagnosing and resolving the system performance bottlenecks.



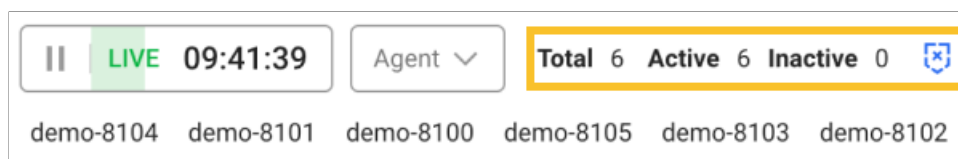
✔ If you select an agent to compare in **Dashboard > Instance List**, and then select **Multi view**, the **Monitoring Multiple Instances** menu appears.

Query Time Setting

The dashboard basically provides the real-time monitoring feature but the past data can be viewed using the time selector at the top of the screen. In the time selector, select **||** and then set the following time. It updates the data in widgets placed on the dashboard based on the set time.

Agent Check

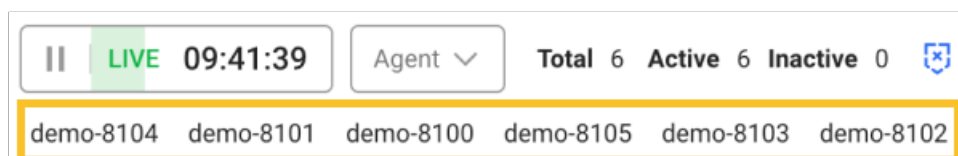
Checking the agent connection status



On the upper left of the screen, the right area of the time selector provides information that allows you to check the status of agents connected to the project. This allows you to immediately check whether the target application server is running.

- **Total**: Number of all agents connected to the project
- **Active**: Number of active agents
- **Inactive**: Number of inactive agents
- : Can display or hide inactive agents.

Agent-based monitoring



By default, the dashboard displays the metrics collected from all agents in charts and you can also view the data by agent. Select one or more agents under the time selector. The data of the widgets on the dashboard is

updated with the metrics of the selected agent(s).

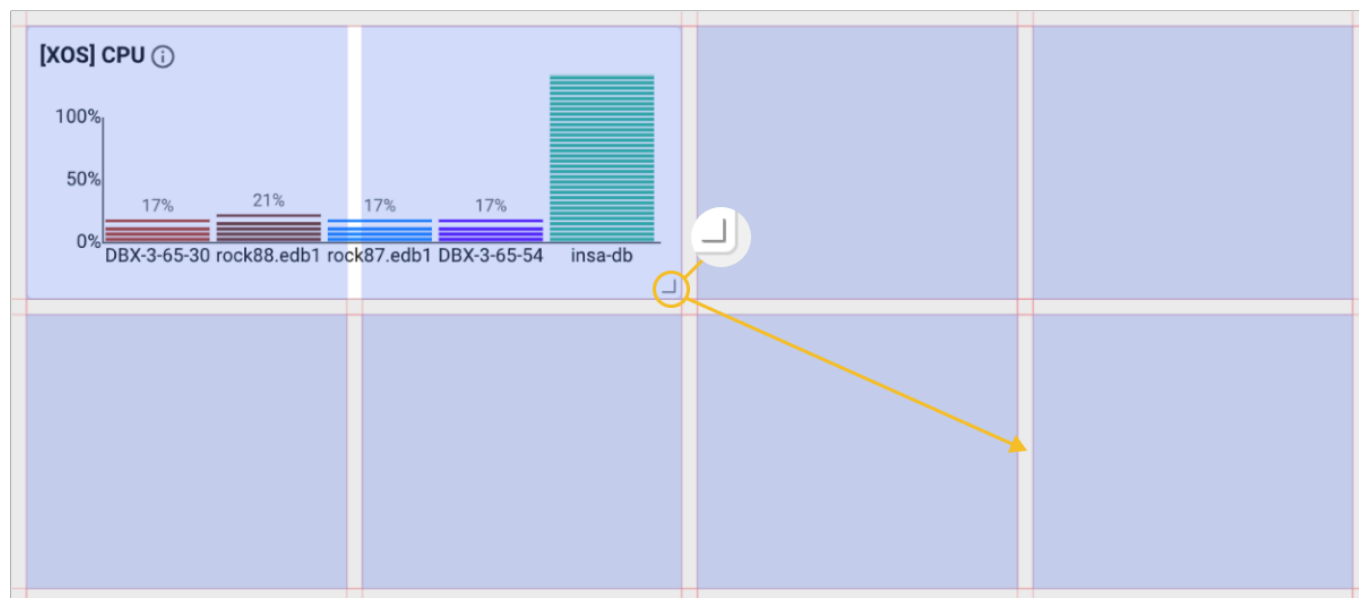
✓ To select all agents again while one or more agents is selected, deselect them or click [Total](#).

! If there are numerous agents connected to the project, it is efficient to set the agent names to be short. For more information about the agent name setting, see [the following](#).

Widget Edit

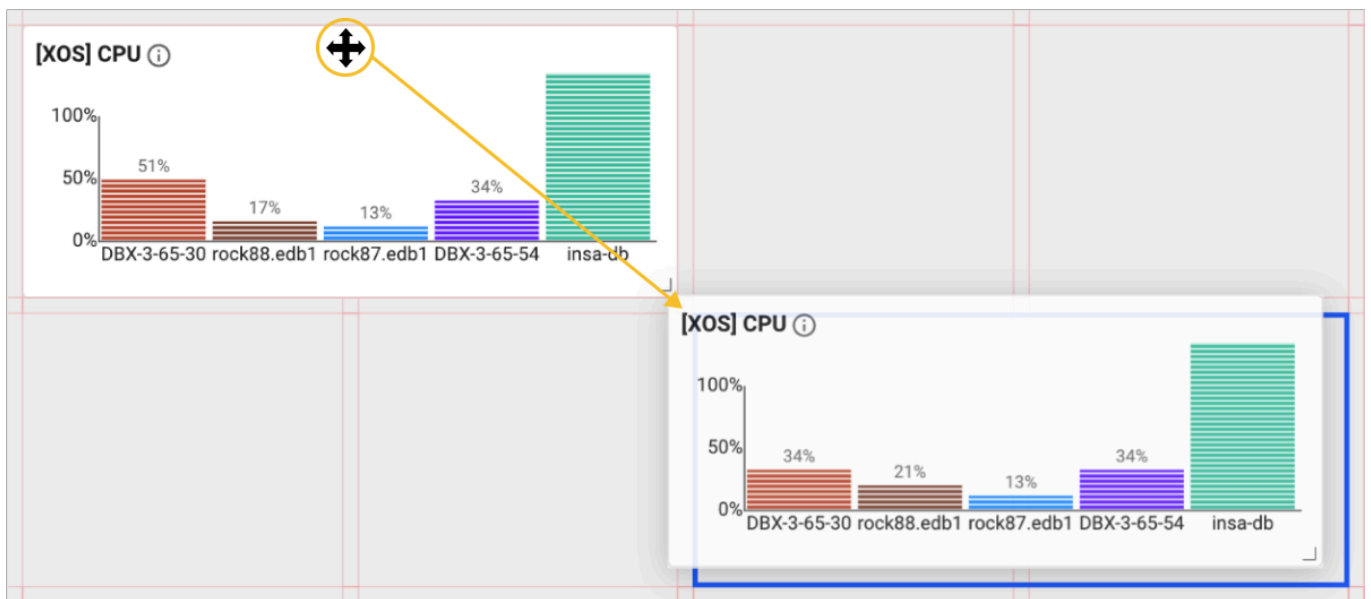
The widgets on the dashboard can be adjusted to the desired size and placed in the desired positions. You can also delete unnecessary widgets or add them again.

Resizing a widget



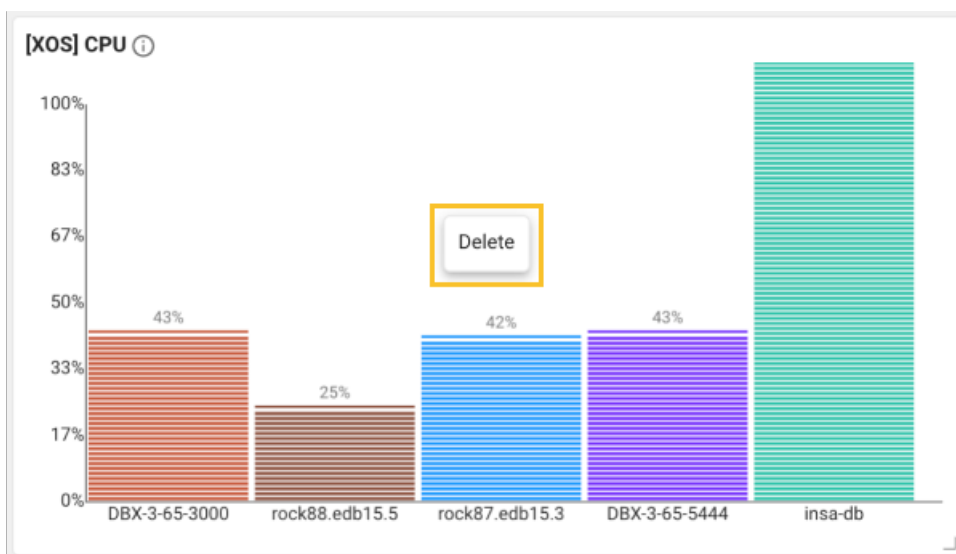
Click and hold the mouse on the  element at the lower right of the widget, and then drag it to the desired size. Grids with a uniform horizontal-to-vertical ratio appear and the size of the widget can be adjusted in each grid.

Moving a widget



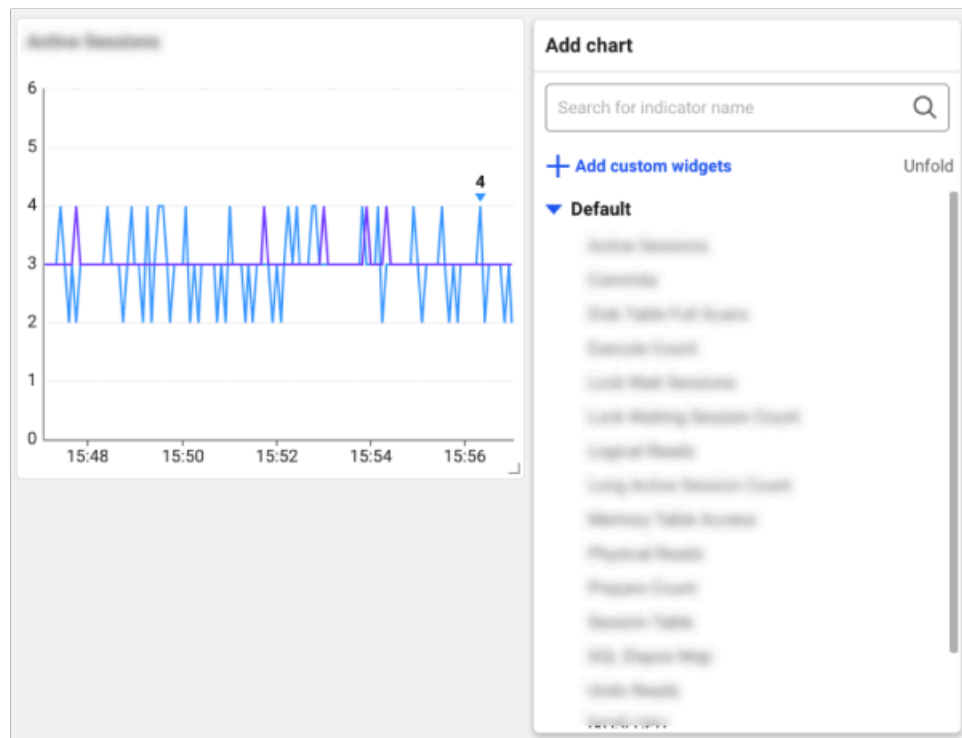
When you move the mouse cursor to the top of the widget, the cursor shape changes to **+**. At this time, you can move the widget by dragging with your left mouse button and move to the desired position.

Deleting a widget



Right-click on a widget to delete. If you select **Delete**, the widget is deleted from the dashboard.

Adding a widget



Move the mouse cursor to an empty space on the dashboard and then right-click on it. Select a widget to add from the pop-up menu. Place the widget to the desired position and then resize it.

- For more information about the widgets that can be placed on the dashboard, see [the following](#).
- In addition to the metrics provided by default on the dashboard, you can add the desired metrics. For more information about custom widgets, see [the following](#).

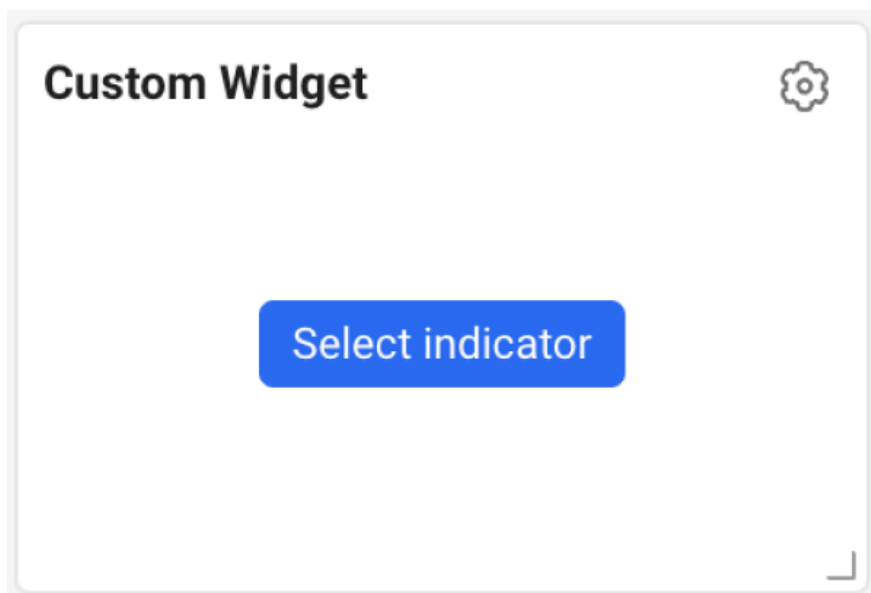
Custom widget

The custom widget feature allows users to add their own metrics in addition to the metrics provided by default on the dashboard. If you have metrics to continuously monitor on the dashboard in addition to the metrics

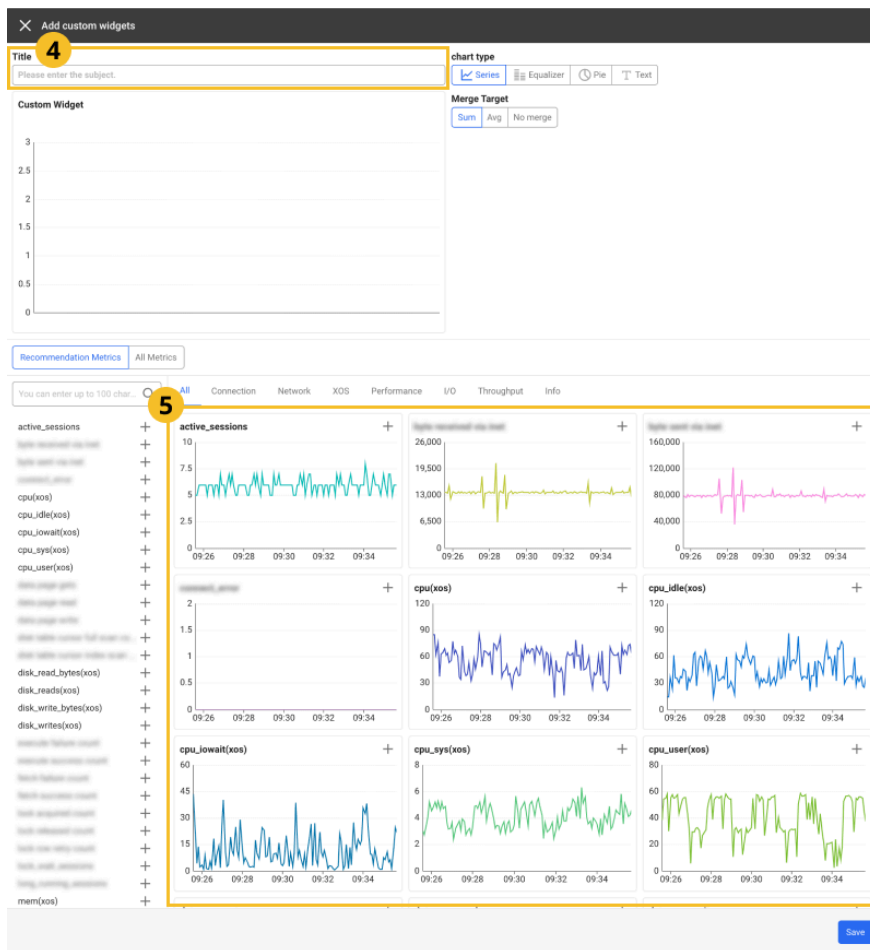
provided by default, use the custom widget feature.

Add

1. Right-click an empty space on the dashboard.
2. On the pop-up menu, select **Add Custom Widgets**.
3. If **Custom Widget** is created on the dashboard, select **Select indicator**.



4. When the **Add Custom Widgets** window appears on the right of the screen, enter the widget name in **Title**.



5. In the metrics list at the bottom of the screen, select the + button on the upper right of the metric to add.
6. After the metric has been changed, select **Save**.

- ! • You can add up to four metrics. To add metrics, select +.
- To change the metric, select a different metric in the metrics list. If you select +, the metric is added.
- If you do not enter the widget title, it is saved as a custom widget.

Selecting the chart type

In **chart type** of the **Add Custom Widgets** window, you can select the following types.

-  **Series**: Time series chart that displays the metric changes over time.
-  **Equalizer**: Bar graph chart that can compare the performance for multiple metrics. By visually displaying various metrics on one screen, you can easily see the overall performance status.
-  **Pie**: Pie chart useful for checking the composition ratio for multiple metrics relative to the whole.
-  **Text**: It displays the figures for the metrics in text.

Selecting the data merging method

In **Merge target** of the **Add Custom Widgets** window, you can select the way to display metrics collected from agents.

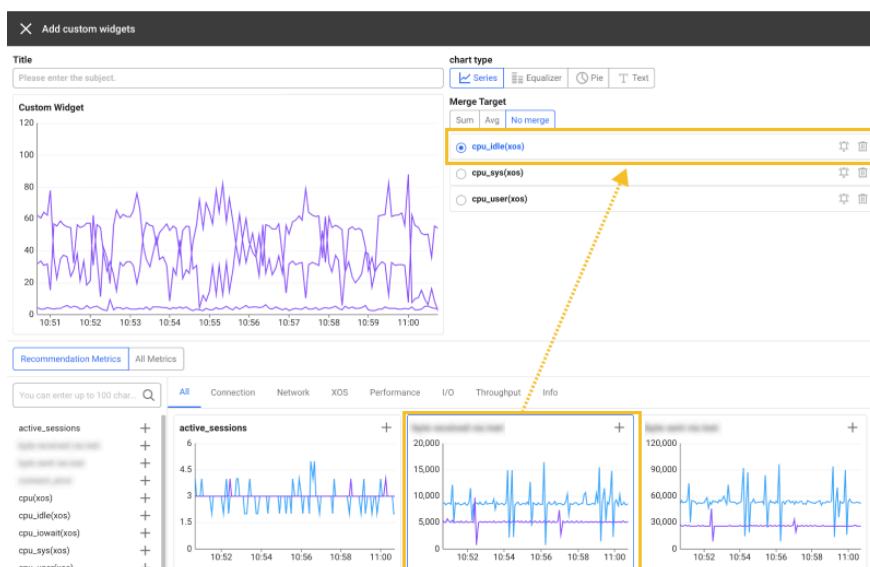
- **Sum**: It sums up the metric values collected from multiple agents and displays them as a single value. This option is useful for displaying the overall total by merging data from multiple agents.
- **Avg**: It averages the metric values collected from multiple agents and displays them as a single value. This option is useful for checking the average performance or status of all agents.
- **Max**: By selecting the largest value among the metric values collected from multiple agents, it displays a single value. This option is useful for identifying the highest performance consumption or worst performance cases among agents.
- **No Merge**: It displays the metrics for each agent.

 The available merge method may differ depending on the chart type.

Changing a metric

You can change or add metrics for the metrics added to the custom widget.

1. Select  on the upper right of the custom widget added on the dashboard.
2. When the **Add Custom Widgets** window appears on the right of the screen, select a metric to change.



3. Select a metric to change from the metric list at the bottom of the screen. To add a metric, select + on the upper right of the metric to add.
4. After the metric has been changed, select **Save**.

- ✓ The features of the icon buttons in the list for added or changed metrics are as follows:
- 🗑️: You can exclude the metric from the custom widget.
 - 🔔: You can set the alert for the metric. When the button is selected, the **Alert > Event Configuration** appears. For more information about the **Metrics event** setting, see [the following](#).

Widget Options

The icons displayed on the widget function as follows:

- ⓘ: You can see the features and information for major widgets. (multilingual support planned)
- ^ or v: You can enlarge or reduce the range of the vertical axis.
- 🖱️: You can view the widget's data on a wider screen.
- 🔍: A detail window appears where you can search the data of the widgets separately by agent.

ⓘ The provided options may differ depending on the widget.

Preset Setting

You can save and load the settings and layouts for widgets on the dashboard. You can create a new preset by adjusting its size and placing it on the desired position.



- **Default:** It is a preset consisting of key DB metrics and XOS metrics.

- ⓘ
- The dashboard settings that can be stored into a preset are as follows:
 - Widget size and location
 - Settings defined in the custom widget
 - Width, filter, and selected column values of the table defined in the **Session Table** widget.
 - The default preset (**Default**) cannot be changed.
 - Depending on the database environment, some presets may not be supported.
 - You can save or modify the presets with the **Edit project** role. For more information about the member roles, see [the following](#).

Creating a new preset

1. Place the widgets in the desired format on the dashboard. You can also resize and place only the widgets to be frequently checked.
2. On the upper right of the screen, select .
3. Enter a new preset name.

To save the selected agent data, select **Include agent selection history**.

4. Select **Save**.

You can see the newly saved preset from the preset list.

- ⓘ • If any changes are made to the newly created preset, save the preset again. Select  and then save the preset with the same name. Any changes are overwritten in the existing preset.
- If you go to another menu without saving changes on the dashboard, the changes are not saved.
- Presets are saved on a project basis and can be shared with other users.

Deleting presets

If you have unused presets, you can delete them from the preset list. Select  on the right of the item to delete from the preset list.

- ⓘ The **Default** preset cannot be deleted.

Learning about the main widgets

The types of widgets that can be placed on the dashboard are as follows.

XOS widgets

- **[XOS] CPU**

You can see the CPU usages for each agent in real time.

- **[XOS] Memory**

You can see the memory usages for each agent in real time.

- **[XOS] Disk Usage**

Displays the disk utilizations, free sizes, and total sizes for each agent.

- : The **Disk usage details** pop-up window appears. You can check the chart that displays the disk usage by time zone. You can check the data by instance. By selecting **Size**, **Free**, or **Used**, you can see the total size, remaining size, and size in use.
- : You can expand the table on the widget to a wider screen.

✓ If there is no data displayed, add the following settings to the [xos.conf](#) file.

xos.conf

```
# Off if the disk usage is 0; unit: second
disk_usage_interval=60
```

Database widgets

The provided widgets allow you to check the status of various system resources, such as CPU, memory, and disk I/O, as key metrics for database performance analysis.

- **Active Sessions**

It visually displays the numbers of active sessions over time in a time series chart. This widget traces and graphs the numbers of sessions running in the database over time. It is useful for visually checking the session usage patterns and load.

- **Lock Wait Sessions**

It visually displays the numbers of lock waiting sessions over time in a time series chart. This widget traces the numbers of sessions in lock waiting state in the database over time and displays it on the graph in real time.

- **Long Active Session Count**

It provides information about active sessions running on the database for long time. A chart appears for information aggregated by time zone with grouped by the session execution time. It can identify long-running sessions with performance issues or resource leaks in the database, or classify the sessions based on the execution time to visually display which sessions stay active for long time.

It appears in **blue** for less than 3 seconds, **green** for 3 seconds or more that is less than 10, **orange** for 10 seconds or more that is less than 15, and **red** for 15 seconds or more.

- **Long Waiting Session Count**

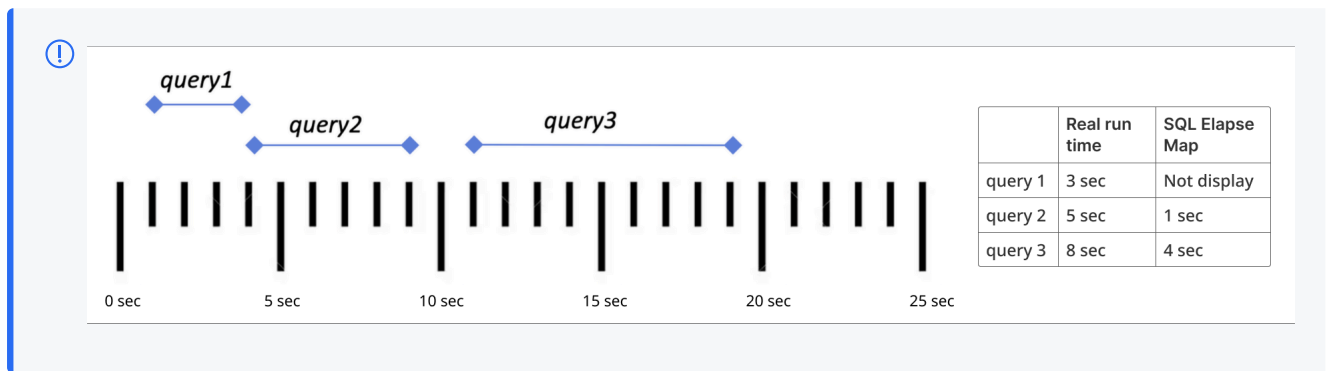
It provides information about the long-term latency of the sessions holding or waiting for locks in the database. By combining lock information and active session information, you can trace long-term latencies for queued sessions. It monitors the sessions that are holding locks or waiting for long time. You can visually see which sessions are holding locks or waiting for long time.

It appears in **blue** for less than 5 seconds, **green** for 5 seconds or more that is less than 10, **orange** for 10 seconds or more that is less than 60, and **red** for 60 seconds or more.

- **SQL Elapse Map**

Displays a real-time chart for the elapsed times of running queries. If you drag a specific part of the chart, the **Query List** appears. You can enlarge or reduce the range of the vertical axis by selecting **^** or **∨**.

ⓘ The execution times of queries are calculated after collecting active sessions. Therefore, depending on the active session collection cycle, there may be a margin of error. See the following figure. The execution times of SQL queries that have ended within the collection cycle are not calculated. A shorter collection cycle may decrease the margin of error, but considering the database load, it is typically set to 5 seconds. It is useful for monitoring long-running queries. The same is true for the data in **SQL statistics**.



- **Wait Event**

It displays a real-time graph for the names of the waiting events that each backend process is waiting on, along with the frequency with which those events occurred. It mainly monitors waiting events related to application, configuration, administrative, network, and the like. You can visually see the frequency of each waiting event. If you hover your mouse over the graph, a tooltip is displayed providing more information about each waiting event.

- **Physical Reads**

This widget monitors physical reads occurring in the database. It monitors the I/O performance of the database, by providing statistics on the operations where the database reads data from the disk.

- **Excute Count**

This widget monitors the numbers of queries or procedures executed in the database. It traces the numbers of queries or procedures executed by the database in real time and provides them to the user.

- **Session Logical Reads**

This widget monitors logical reads executed in the database session. It traces and provides real-time visibility to users on the number of logical read operations executed in the database session.

Session Table widget

You can view real-time active sessions and lock trees in the [Session Table](#) widget at the bottom of the screen.

- **Active sessions:** You can view active sessions and queries.

- If you select a session running, the **Session detail** window appears. You can see the details for the

selected session. For more information, see [the following](#).

- If you select an item in the rightmost [query_text](#) column in the table, the **SQL details** window appears. You can see the full text and plan information for the selected query. For more information, see [the following](#).

- **Lock tree**

You can see the lock session information, and holder and waiter information in real time. For more information about the table columns, see [the following](#).

ⓘ In the table list, the text colors are changed black → orange → Red, which means that the performance of the session is getting slower.

Checking the session information details

If you select a session in the **Active sessions** tab, the **Session detail** window appears. You can see the information details about the selected session.

Active sessions
Lock tree
Process Info

instance	con_name	sid	serial#	cpu(xos)	pga memory	username	last_call_et	event	query_text	sql_id	session logical...
DBX-2-39-2105	PDB1	422	46285	0.76%	0	KWLEE	3	enq: TX - row lock co...	update kwlee set b='111' where a=x	9xn1nukt0jcw7	0
DBX-2-39-2105	PDB1	212	29166	0.00%	0	KWLEE	23	PL/SQL lock timer	declare aa number; begin aa := 1; execute imm...	54qb6k9mf4n99	0
DBX-1-97-1521		17	7023	0.00%	0	WHATAP	0	SQL*Net message to ...	select /* WhaTap3D#4 */ 0 as con_id,pl.plan_h...	12zmp1v34h271	0

Session detail (sid: 17, serial#: 7023)

LIVE 10:43:04
Kill Session

Session history

Session Stat
ALL

Delta
Sigma

Event	Total Waits	Total Timeouts	Time Waited
Disk file operations I/O	0	0	
control file sequential read	0	0	
latch: cache buffers chains	0	0	
buffer busy waits	0	0	
read by other session	0	0	
db file sequential read	0	0	

Stat	Value
logons cumulative	4
logons current	0
opened cursors cumulative	1
Requests to/from client	0
user commits	7
recursive calls	5
recursive cpu usage	0
pinned cursors current	0
user logons cumulative	0

Session Info

username	WHATAP	status	1
schemaname	WHATAP	command	3
spid	5964	last_call_et	0
osuser	whatap	sql_id	12zmp1v34h271
machine	dbx-agent	sql_hash_value	3326609633
program	JDBC Thin Client	plan_hash_value	57415992
blocking_session	0	module	JDBC Thin Client
row_wait_block#	130	action	
row_wait_obj#	-1		

Wait Info

seq#	23770	p1	1952673792
event#	431	p2	1
event	SQL*Net message to client	p3	0
wait_class		wait_time	-1
state(xos)	none		

SQL Text
Plan

Default Bind
Reset Bind
Formatting

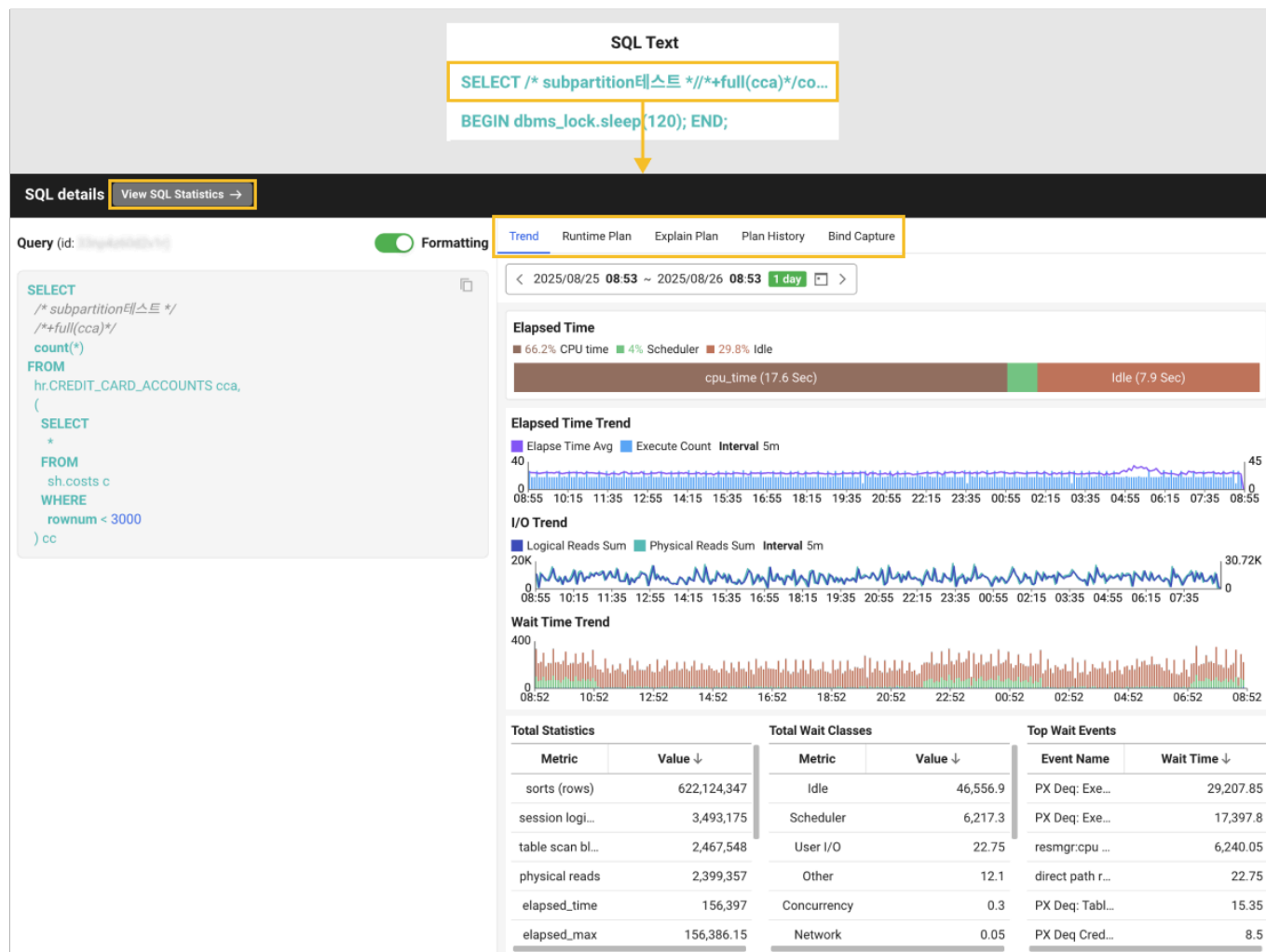
```
select /* WhaTap3D#4 */ 0 as con_id,pl.plan_hash_value, pl2.plan_hash_value
prev_plan_hash_value,saddr,s.sid,s.serial#,s.username,command,taddr,lockwait,s.status,
osuser,p.spid,process,machine,s.terminal,s.program,type,sql_address,s.sql_hash_value,s.
sql_child_number,prev_sql_addr,prev_hash_value,prev_sql_id,
prev_child_number,s.module,s.action,client_info,row_wait_obj#,
row_wait_file#,row_wait_block#,row_wait_row#,logon_time,last_call_et,pdml_status,
pddl_status,pq_status,client_identifier,blocking_session_status,
blocking_instance,blocking_session,
seq#,event#,event,p1,p2,p3,wait_class#,wait_time,
seconds_in_wait,service_name,
x.xidusn,x.used_ublk,x.used_urec
,plsql_entry_object_id,plsql_entry_subprogram_id,plsql_object_id,plsql_subprogram_id
,port,sql_exec_start,sql_exec_id,prev_exec_start,prev_exec_id,final_blocking_session_stat
,p.cpu_used
from v$session s,v$process p,v$transaction x ,v$sql pl,v$sql pl2
```

! For more information about the **Session detail** window.

See query details

When you select the query column item in the result list, the **SQL Details** window appears.

Click the **View SQL Statistics →** button to move to the **SQL statistics** menu, where you can check statistical information related to the SQL query.



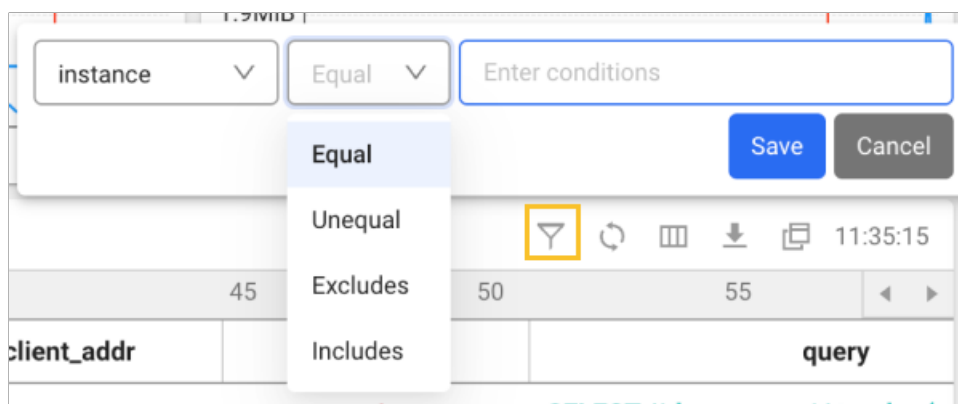
The tabs in the **SQL Details** window are as follows:

- **Trend:** Provides a comprehensive analysis of the execution time, I/O, and wait events of a specific SQL, allowing quick identification of performance bottlenecks.
- **Runtime Plan:** Displays the execution plan and runtime information of the selected SQL query, including execution count, average execution time, and average physical reads.

- **Explain Plan:** Shows the execution plan predicted by the optimizer, including cost, operations, object names, and cardinality.
- **Plan History:** Displays the history of execution plans for SQL queries executed in the database.
- **Bind Capture:** Shows the values of bind variables used in SQL queries executed in the database, helping you understand the actual execution content of the query.

! You can see the tuning guide after selecting the  button in the Plan tree. The tuning guide can be viewed when the nodes in the Plan Tree are `TABLE ACCESS (FULL)` and `INDEX (SKIP SCAN)` and the nodes contain the keyword, `CARTESIAN`.

Filtering the table data



1. Select  on the upper right of the table.
2. Select a column header and a condition in the table.
3. Enter a desired value in the **Enter conditions** field.
4. Select **Save**.

Setting the table columns

You can hide the table header columns or add any of them. You can also change the column order. Select .

- ! • After configuration, select **Confirm** 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.
- Images may differ depending on the product, project, or menu.

Adding columns

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

Deleting columns

From the **1** list, unselect the columns to delete. Alternatively, select **×** on the right of the item to delete from the **2** list.

Changing the column order

Drag an item to reposition from the **2** list, and then move it to the desired position.

Initializing the configuration

To cancel all changes and reset them, select **↺ Initialized**.

Column information guide

For more information about columns, see [the link](#).

Item	Description
sid	Session ID
cpu(xos)	CPU utilization collected by the installed XOS
serial#	Value to increment to determine if the same session ID has been used.
username	DB user name.

Item	Description
command	Same as <code>COMMAND_TYPE</code> ; SQL command number
status	Session status
schemaname	Schema. Same as the username.
osuser	Username of the OS that accessed the session.
spid	OS process ID.
process	ID of the client process that accessed the session.
logon_time	Session created time point.
last_call_et	If the session is active, it is the elapsed time (sec) since activation.
sql_text	<code>query</code> , running statement, SQL statement that separates literal strings
sql_param	Literal value separated from <code>query</code>
prev_sql_text	Previously executed query
prev_sql_param	Previously executed query param
program	Program information executed by client
type	session type - can be adjusted in user background recursive (whatap.conf).
seq#	Number that uniquely identifies the current or last wait (incremented with each wait)
event	Number of the resource or event if the session is waiting.
p1	First waiting event parameter (decimal point)
p2	Second waiting event parameter (decimal point)

Item	Description
p3	Third waiting event parameter (decimal point)
wait_class	Class name of the waiting event
wait_time	The value is 0 if the session is waiting. It is the last waiting time (milliseconds) if the value is greater than 0.
seconds_in_wait	Time spent waiting for the wait if the session is waiting. Otherwise, it is the time elapsed since the last waiting.
state	Waiting state: WAITING , WAITED UNKNOWN TIME , WAITED SHORT TIME , WAITED KNOWN TIME
machine	Name of the client server that accessed the session (hostname)
port	Client port number
terminal	Client terminal name
module	Name of the module set by the user of the program executed on the client.
action	Action name set by the user of the running module.
client_info	Client information set by user
client_identifier	Client ID set by user
session logical reads	Number of times data was read into the buffer cache
physical reads	Number of data reads from the physical disk
execute count	SQL execution count
parse count (hard)	Total number of syntax analysis calls (actual syntax analysis)
parse count (total)	Total number of syntax analysis calls (hard, soft, describe)

Item	Description
opened cursors current	Number of open cursors
db block changes	Change count of all blocks in the database
session pga memory	Session's current PGA size
undo_segid	Undo information ID
undo_blk	Undo block count
undo_rec	Undo record count
sql_address	SQL statement identification data
sql_hash_value	SQL statement identification data
sql_id	SQL ID of the running SQL statement
sql_child_number	Sub number of the running SQL statement
sql_exec_start	Start time point of SQL running in the session
sql_exec_id	SQL running ID
prev_sql_addr	Information that identifies the last run SQL statement
prev_child_number	Sub number of the last SQL statement executed
prev_exec_start	Start time point of the last SQL statement executed
prev_exec_id	Start ID of the last SQL statement executed
plssql_entry_object_id	Object ID of the top PL/SQL subprogram on the stack.
plssql_entry_subprogram_id	ID of the top PL/SQL subprogram on the stack.

Item	Description
<code>plsql_object_id</code>	Subprogram ID of the running PL/SQL object
<code>taddr</code>	Transaction address
<code>lockwait</code>	lock wait address (lock address where the session is waiting)
<code>row_wait_obj</code>	Object ID of the table containing the row specified in <code>ROW_WAIT_ROW#</code>
<code>row_wait_file</code>	ID of the data file containing the row specified in <code>ROW_WAIT_ROW#</code>
<code>row_wait_block</code>	ID of the block containing the row specified in <code>ROW_WAIT_ROW#</code>
<code>row_wait_row</code>	Currently locked row
<code>pdml_status</code>	If the value is <code>ENABLED</code> , the session is in <code>PARALLEL DML</code> activation mode. If the value is <code>DISABLED</code> , the session does not support the <code>PARALLEL DML</code> activation mode. If the value is <code>DISABLED</code> , the session does not support the <code>PARALLEL DML</code> activation mode.
<code>pq_status</code>	If the value is <code>ENABLED</code> , the session is in <code>PARALLEL QUERY</code> activation mode. If the value is <code>DISABLE</code> , the session does not support the <code>PARALLEL QUERY</code> activation mode. If the value is <code>FORCED</code> , it indicates the session was changed to force the use of <code>PARALLEL QUERY</code> .
<code>blocking_session_status</code>	It provides details about whether there is a blocking session.
<code>blocking_instance</code>	Instance ID for the blocking session
<code>blocking_session</code>	Session ID for the blocking session
<code>final_blocking_session_status</code>	It provides details about whether there is a last block session.
<code>final_blocking_instance</code>	Instance ID for the last blocking session
<code>final_blocking_session</code>	Session ID for the last blocking session

Item	Description
<code>service_name</code>	Service name of the session. It is similar to the DB name.
<code>saddr</code>	Session address
<code>con_id</code>	ID of the container associated with the data (only for multi-database support)
<code>rss(xos)</code>	OS metrics collected by the XOS agent
<code>pss(xos)</code>	OS metrics collected by the XOS agent
<code>ioread(xos)</code>	OS metrics collected by the XOS agent
<code>iowrite(xos)</code>	OS metrics collected by the XOS agent

ⓘ WhaTap basically stores the client-related information.

Downloading the view list

You can download the current [Session Table](#) list in CSV format. Select  on the upper right of the [Session Table](#) section.

The naming format of the downloaded file is as follows:

- **Active sessions:** `activeSession_HH_MM_SS.csv`
- **Lock tree:** `lockTree_HH_MM_SS.csv`

Viewing in a new window

You can see the [Session Table](#) list on a new wider screen. Select  on the upper right of the [Session Table](#) section.

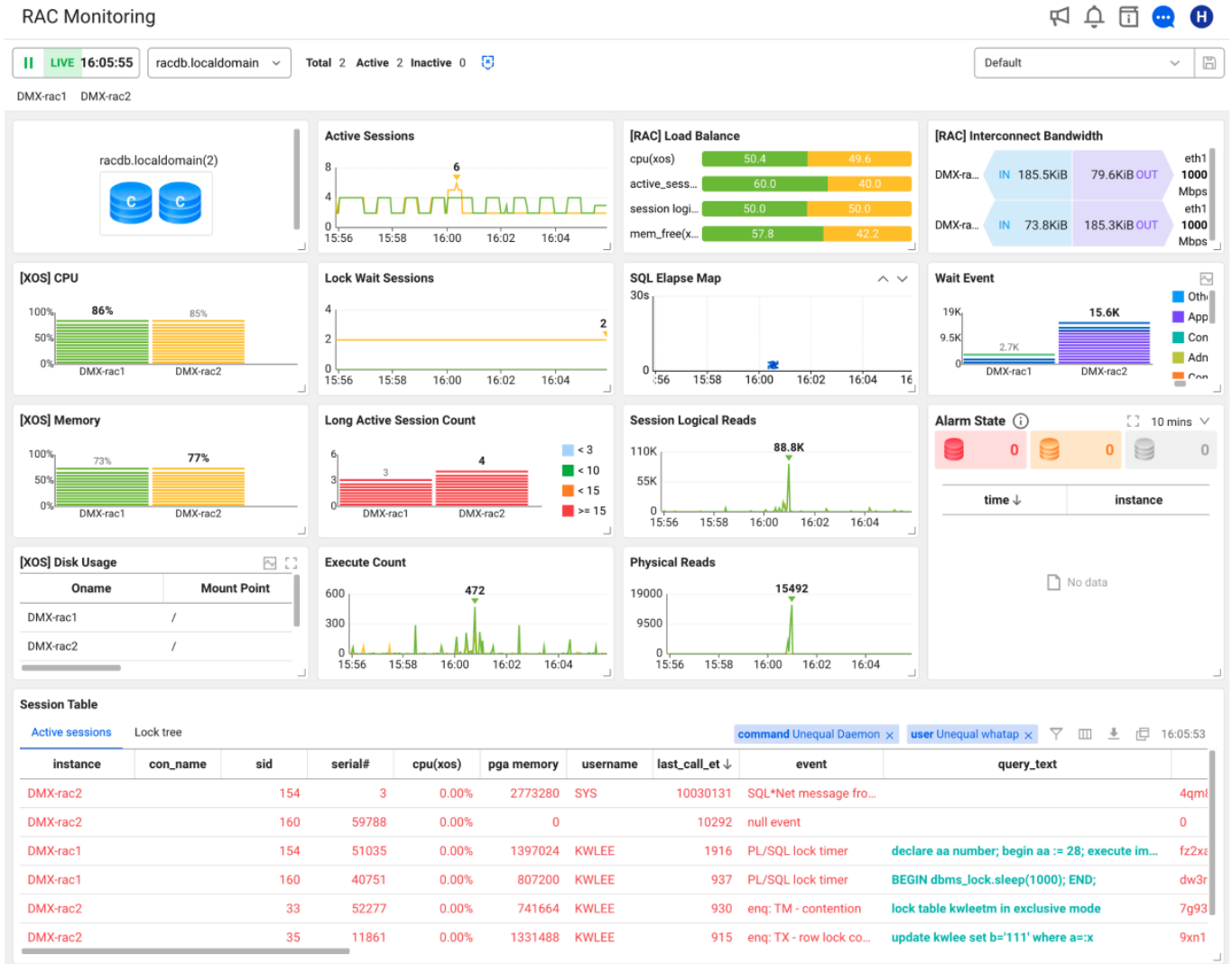
RAC Monitoring

Home > Select Project > **Dashboard** > **RAC Monitoring**

[RAC Monitoring](#) is configured to allow monitoring of databases formed as a cluster in Oracle Database. The default presets provided in the [RAC Monitoring](#) menu are optimized for the monitoring. The edit role is required to save, delete, and change the presets.

You can compare and analyze key performance metrics occurring in each instance at a glance. You can effectively perform monitoring by aggregating data such as the number of agents running, the sum of all active sessions, and the number of transactions. Users can easily check the changes in metric values over time and high performance metrics for each instance through graph charts.

Additionally, users can see the longest-lasting active sessions, which can be used as a basis for diagnosing and resolving the system performance bottlenecks.



✓ Difference from Multiple instance monitoring

The RAC monitoring screen was provided as an instance selector that can select RAC presets and clusters in the previous multi-instance monitoring, but has been separated into a dashboard.

- Available widgets may differ.
 - [RAC] Load Balance
 - [RAC] Interconnect Bandwidth
 - [RAC] Main Waits
- The default preset is optimized for RAC monitoring.



- Instance Select has been changed to Cluster Select.

Query Time Setting

The dashboard basically provides the real-time monitoring feature but the past data can be viewed using the time selector at the top of the screen. In the time selector, select **II** and then set the following time. It updates the data in widgets placed on the dashboard based on the set time.

Agent Check

Checking the agent connection status

II **LIVE** 09:41:39 Agent ▾ **Total 6 Active 6 Inactive 0** ⓘ

demo-8104 demo-8101 demo-8100 demo-8105 demo-8103 demo-8102

On the upper left of the screen, the right area of the time selector provides information that allows you to check the status of agents connected to the project. This allows you to immediately check whether the target application server is running.

- **Total**: Number of all agents connected to the project
- **Active**: Number of active agents
- **Inactive**: Number of inactive agents
- ⓘ: Can display or hide inactive agents.

Agent-based monitoring

II **LIVE** 09:41:39 Agent ▾ **Total 6 Active 6 Inactive 0** ⓘ

demo-8104 demo-8101 demo-8100 demo-8105 demo-8103 demo-8102

By default, the dashboard displays the metrics collected from all agents in charts and you can also view the data by agent. Select one or more agents under the time selector. The data of the widgets on the dashboard is updated with the metrics of the selected agent(s).

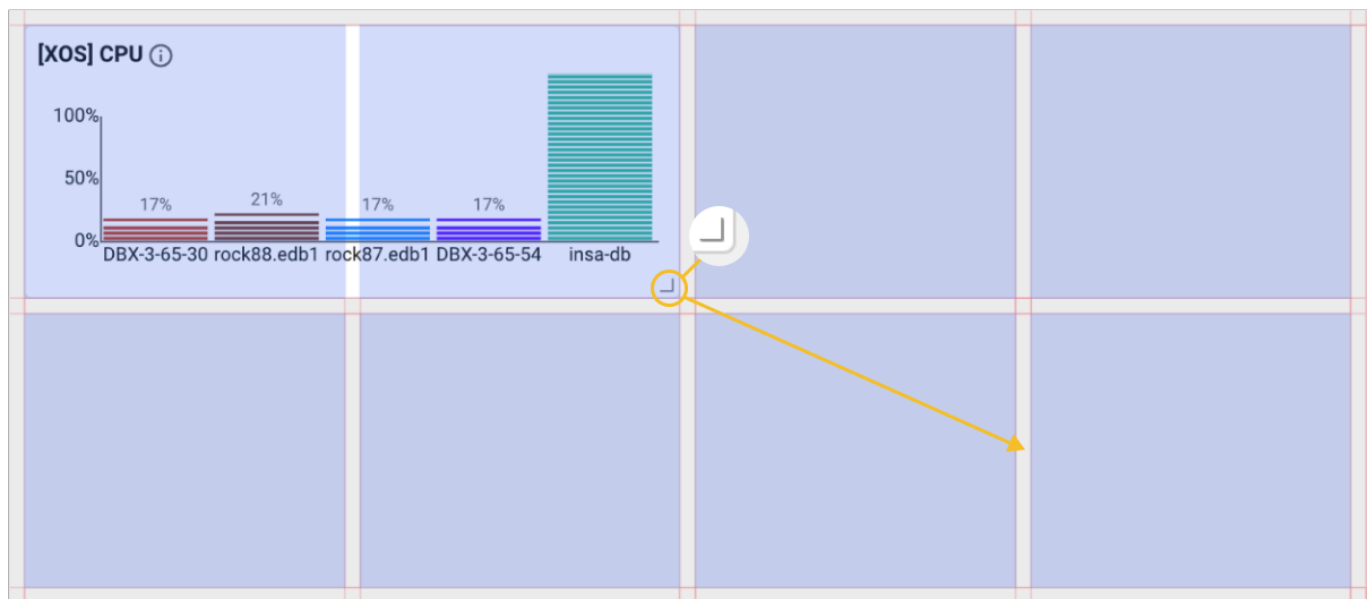
✓ To select all agents again while one or more agents is selected, deselect them or click [Total](#).

! If there are numerous agents connected to the project, it is efficient to set the agent names to be short. For more information about the agent name setting, see [the following](#).

Widget Edit

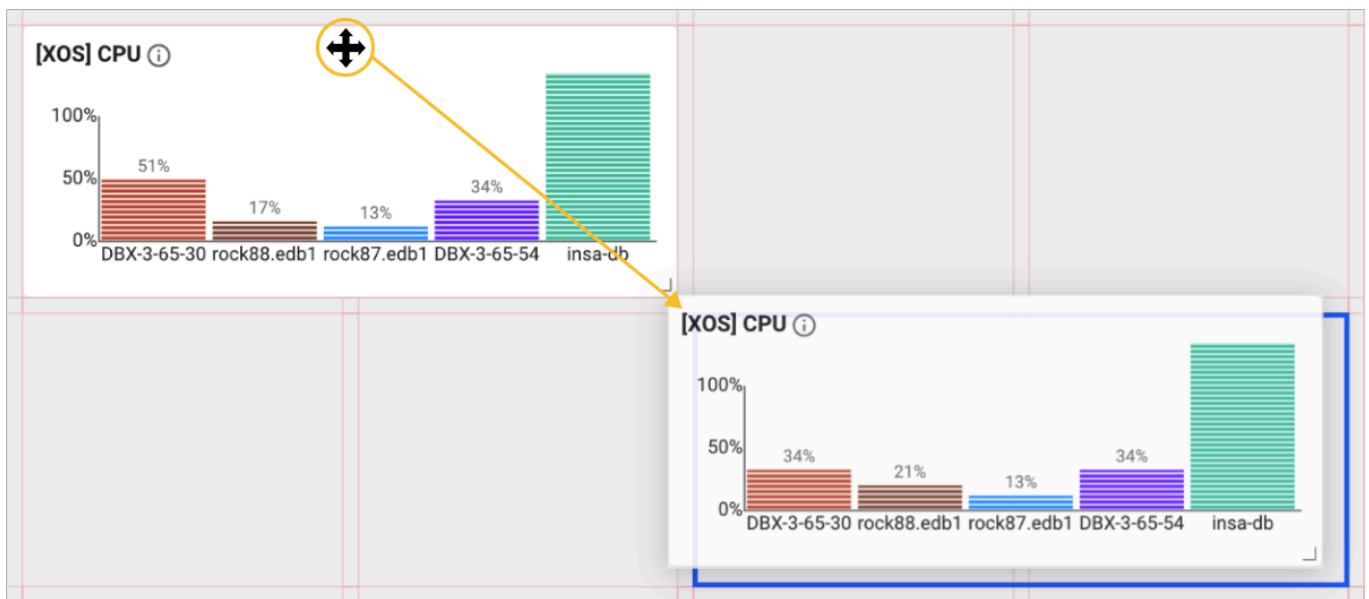
The widgets on the dashboard can be adjusted to the desired size and placed in the desired positions. You can also delete unnecessary widgets or add them again.

Resizing a widget



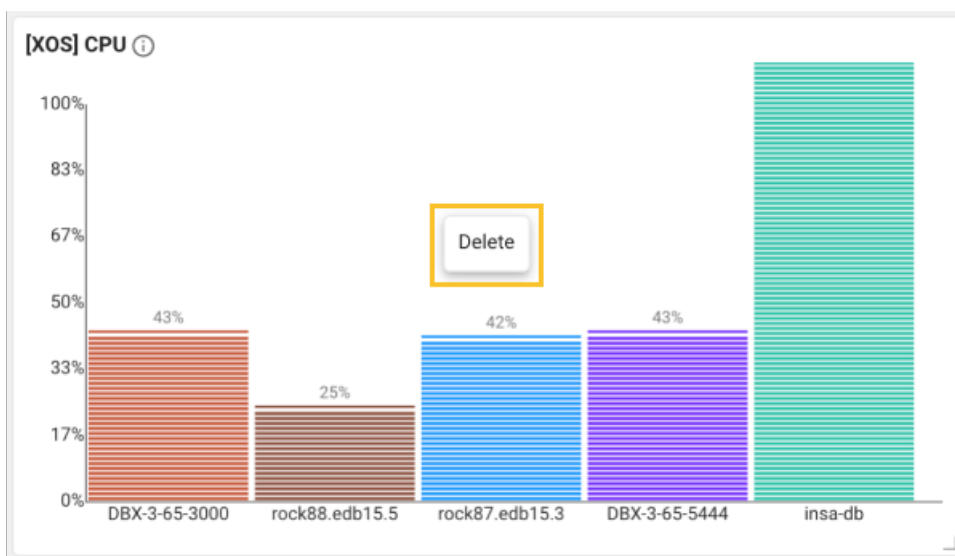
Click and hold the mouse on the  element at the lower right of the widget, and then drag it to the desired size. Grids with a uniform horizontal-to-vertical ratio appear and the size of the widget can be adjusted in each grid.

Moving a widget



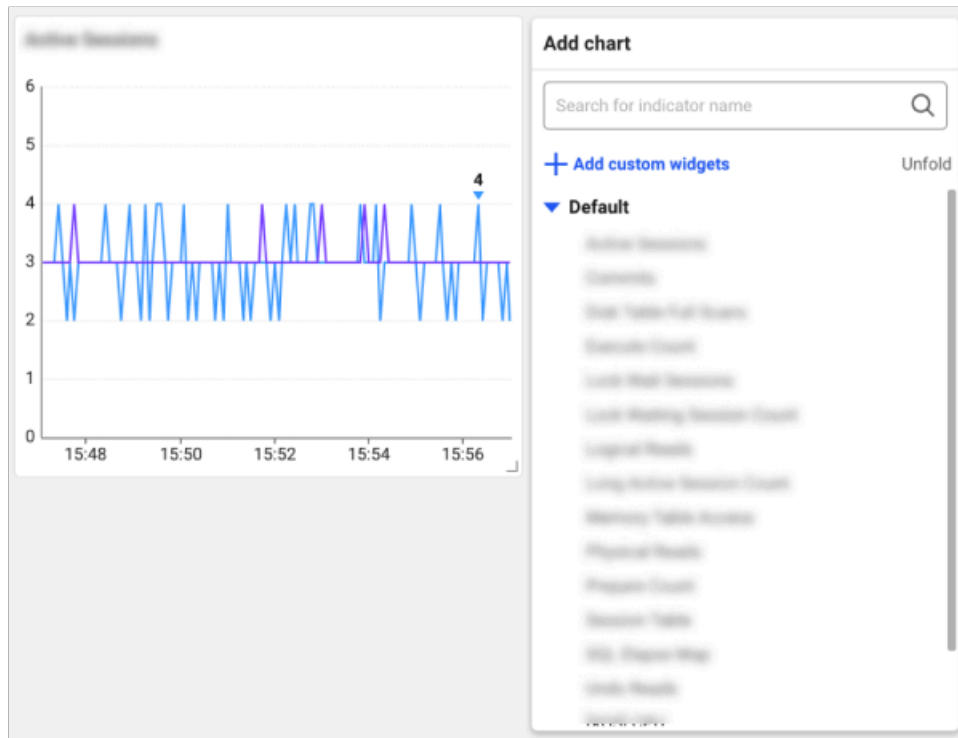
When you move the mouse cursor to the top of the widget, the cursor shape changes to **+**. At this time, you can move the widget by dragging with your left mouse button and move to the desired position.

Deleting a widget



Right-click on a widget to delete. If you select **Delete**, the widget is deleted from the dashboard.

Adding a widget



Move the mouse cursor to an empty space on the dashboard and then right-click on it. Select a widget to add from the pop-up menu. Place the widget to the desired position and then resize it.

- For more information about the widgets that can be placed on the dashboard, see [the following](#).
- In addition to the metrics provided by default on the dashboard, you can add the desired metrics. For more information about custom widgets, see [the following](#).

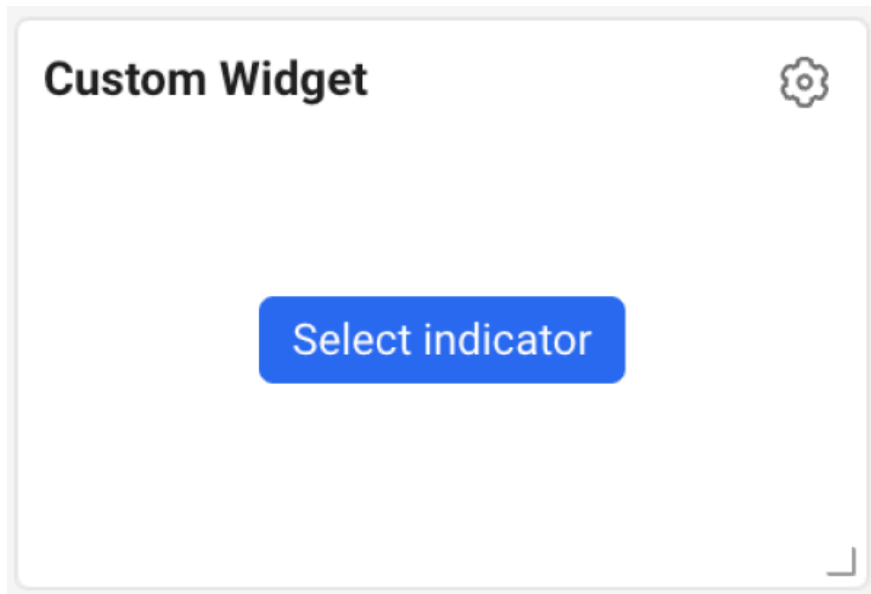
Custom widget

The custom widget feature allows users to add their own metrics in addition to the metrics provided by default on the dashboard. If you have metrics to continuously monitor on the dashboard in addition to the metrics

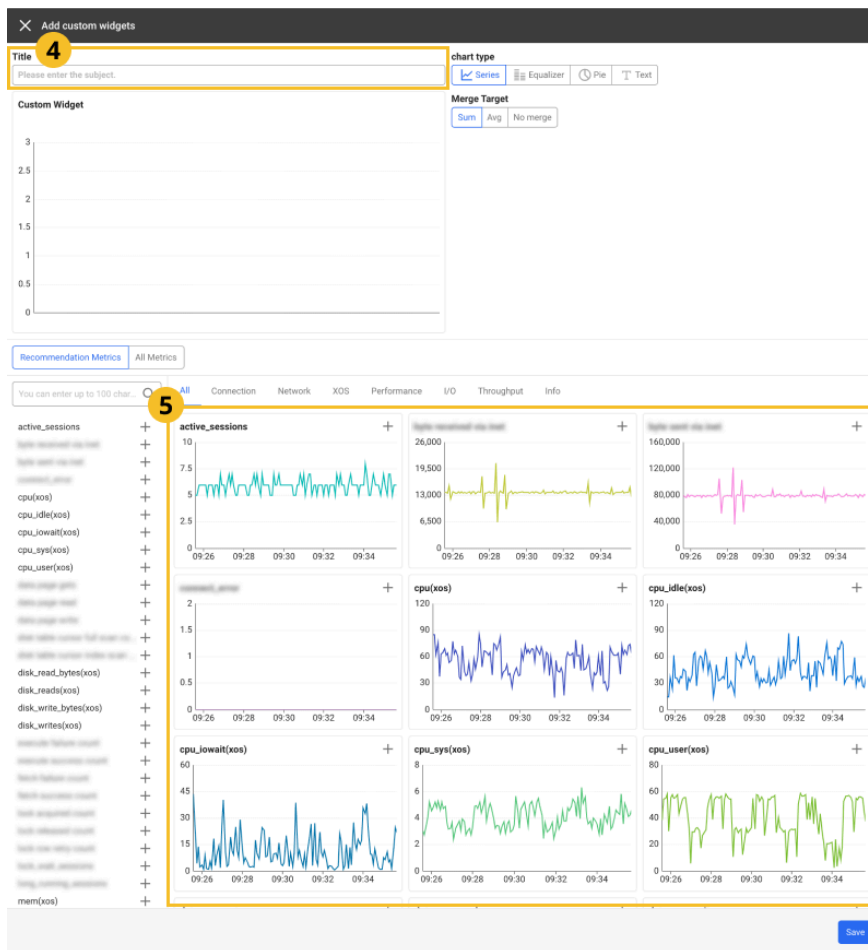
provided by default, use the custom widget feature.

Add

1. Right-click an empty space on the dashboard.
2. On the pop-up menu, select **Add Custom Widgets**.
3. If **Custom Widget** is created on the dashboard, select **Select indicator**.



4. When the **Add Custom Widgets** window appears on the right of the screen, enter the widget name in **Title**.



5. In the metrics list at the bottom of the screen, select the + button on the upper right of the metric to add.
6. After the metric has been changed, select **Save**.

- ! • You can add up to four metrics. To add metrics, select +.
- To change the metric, select a different metric in the metrics list. If you select +, the metric is added.
- If you do not enter the widget title, it is saved as a custom widget.

Selecting the chart type

In **chart type** of the **Add Custom Widgets** window, you can select the following types.

-  **Series**: Time series chart that displays the metric changes over time.
-  **Equalizer**: Bar graph chart that can compare the performance for multiple metrics. By visually displaying various metrics on one screen, you can easily see the overall performance status.
-  **Pie**: Pie chart useful for checking the composition ratio for multiple metrics relative to the whole.
-  **Text**: It displays the figures for the metrics in text.

Selecting the data merging method

In **Merge target** of the **Add Custom Widgets** window, you can select the way to display metrics collected from agents.

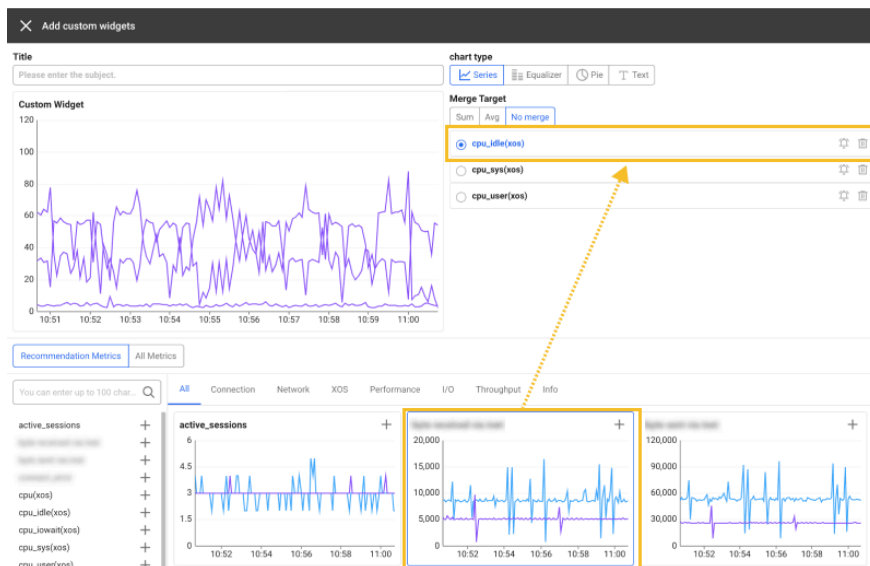
- **Sum**: It sums up the metric values collected from multiple agents and displays them as a single value. This option is useful for displaying the overall total by merging data from multiple agents.
- **Avg**: It averages the metric values collected from multiple agents and displays them as a single value. This option is useful for checking the average performance or status of all agents.
- **Max**: By selecting the largest value among the metric values collected from multiple agents, it displays a single value. This option is useful for identifying the highest performance consumption or worst performance cases among agents.
- **No Merge**: It displays the metrics for each agent.

 The available merge method may differ depending on the chart type.

Changing a metric

You can change or add metrics for the metrics added to the custom widget.

1. Select  on the upper right of the custom widget added on the dashboard.
2. When the **Add Custom Widgets** window appears on the right of the screen, select a metric to change.



3. Select a metric to change from the metric list at the bottom of the screen. To add a metric, select **+** on the upper right of the metric to add.
4. After the metric has been changed, select **Save**.

- ✓ The features of the icon buttons in the list for added or changed metrics are as follows:
- : You can exclude the metric from the custom widget.
 - : You can set the alert for the metric. When the button is selected, the **Alert > Event Configuration** appears. For more information about the **Metrics event** setting, see [the following](#).

Widget Options

The icons displayed on the widget function as follows:

- : You can see the features and information for major widgets. (multilingual support planned)
- or : You can enlarge or reduce the range of the vertical axis.
- : You can view the widget's data on a wider screen.
- : A detail window appears where you can search the data of the widgets separately by agent.

! The provided options may differ depending on the widget.

Setting presets

You can save and load the settings and layouts for widgets on the dashboard. You can create a new preset by adjusting its size and placing it on the desired position.

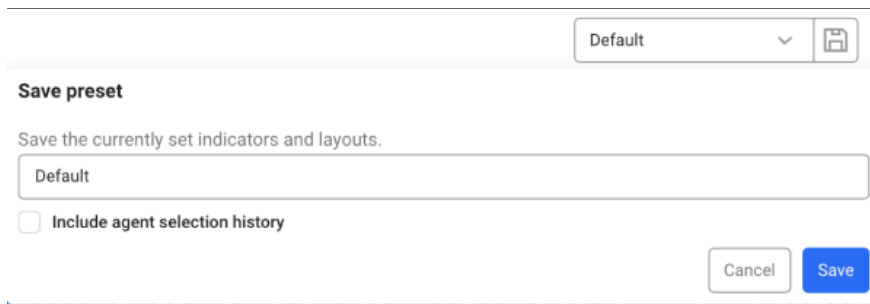


- **Default:** [RAC Monitoring](#) preset consisting of key DB and XOS metrics on the screen. You can check the metrics based on the instances running on the RAC database.

- ! • The dashboard settings that can be stored into a preset are as follows:
- Widget size and location
 - Settings defined in the custom widget
 - Width, filter, and selected column values of the table defined in the **Session Table** widget.
 - The default preset (**Default**) cannot be changed.
 - Depending on the database environment, some presets may not be supported.
 - You can save or modify the presets with the **Edit project** role. For more information about the member roles, see [the following](#).

Creating a new preset

1. Place the widgets in the desired format on the dashboard. You can also resize and place only the widgets to be frequently checked.
2. On the upper right of the screen, select .
3. Enter a new preset name.



To save the selected agent data, select **Include agent selection history**.

4. Select **Save**. You can see the newly saved preset from the preset list.

- ! • If any changes are made to the newly created preset, save the preset again. Select  and then save the preset with the same name. Any changes are overwritten in the existing preset.
- If you go to another menu without saving changes on the dashboard, the changes are not saved.
- Presets are saved on a project basis and can be shared with other users.

Deleting presets

If you have unused presets, you can delete them from the preset list. Select  on the right of the item to delete from the preset list.

- ! The **Default** preset cannot be deleted.

Learning about the main widgets

The types of widgets that can be placed on the dashboard are as follows.

XOS widgets

- [XOS] CPU

You can see the CPU usages for each agent in real time.

- **[XOS] Memory**

You can see the memory usages for each agent in real time.

- **[XOS] Disk Usage**

Displays the disk utilizations, free sizes, and total sizes for each agent.

- : The **Disk usage details** pop-up window appears. You can check the chart that displays the disk usage by time zone. You can check the data by instance. By selecting **Size**, **Free**, or **Used**, you can see the total size, remaining size, and size in use.
- : You can expand the table on the widget to a wider screen.

✓ If there is no data displayed, add the following settings to the [xos.conf](#) file.

xos.conf

```
# Off if the disk usage is 0; unit: second
disk_usage_interval=60
```

Oracle RAC widget

- **[RAC] Load Balance**

You can monitor the resources of instances running on the Oracle RAC (Real Application Clusters) database in real time. Users can visually monitor the load and performance metrics for their database. You can check the CPU performance, active sessions, physical read operations, and available memory.

- **[RAC] Interconnect Bandwidth**

This widget monitors the interconnect bandwidth used in Oracle RAC (Real Application Clusters). You can monitor the network traffic between databases and instances within a cluster by displaying the communication bandwidth between database nodes in real time.

- **[RAC] Main Waits**

You can see the global lock status and the numbers of waits for each event. This widget queries the total

number of waits for each event through metrics collected for a specific database category and extracts the most recent data for each agent (instance). This allows you to see the database wait events promptly at a specific time.

Database widgets

The provided widgets allow you to check the status of various system resources, such as CPU, memory, and disk I/O, as key metrics for database performance analysis.

- **Active Sessions**

It visually displays the numbers of active sessions over time in a time series chart. This widget traces and graphs the numbers of sessions running in the database over time. It is useful for visually checking the session usage patterns and load.

- **Lock Wait Sessions**

It visually displays the numbers of lock waiting sessions over time in a time series chart. This widget traces the numbers of sessions in lock waiting state in the database over time and displays it on the graph in real time.

- **Long Active Session Count**

It provides information about active sessions running on the database for long time. A chart appears for information aggregated by time zone with grouped by the session execution time. It can identify long-running sessions with performance issues or resource leaks in the database, or classify the sessions based on the execution time to visually display which sessions stay active for long time.

It appears in **blue** for less than 3 seconds, **green** for 3 seconds or more that is less than 10, **orange** for 10 seconds or more that is less than 15, and **red** for 15 seconds or more.

- **Long Waiting Session Count**

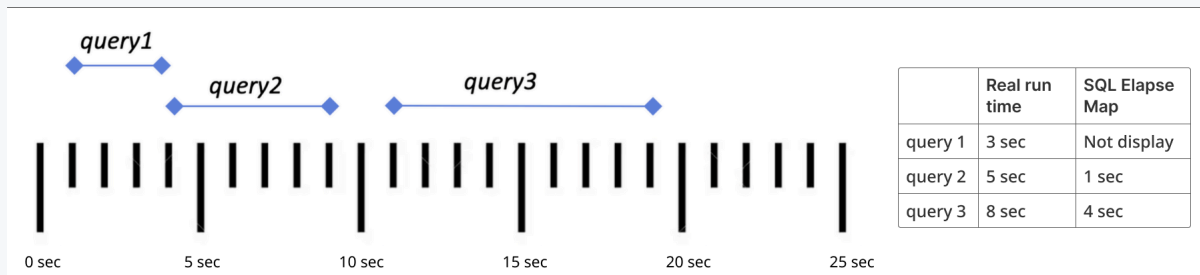
It provides information about the long-term latency of the sessions holding or waiting for locks in the database. By combining lock information and active session information, you can trace long-term latencies for queued sessions. It monitors the sessions that are holding locks or waiting for long time. You can visually see which sessions are holding locks or waiting for long time.

It appears in **blue** for less than 5 seconds, **green** for 5 seconds or more that is less than 10, **orange** for 10 seconds or more that is less than 60, and **red** for 60 seconds or more.

• SQL Elapse Map

Displays a real-time chart for the elapsed times of running queries. If you drag a specific part of the chart, the **Query List** appears. You can enlarge or reduce the range of the vertical axis by selecting $\wedge$ or $\vee$.

- ① The execution times of queries are calculated after collecting active sessions. Therefore, depending on the active session collection cycle, there may be a margin of error. See the following figure. The execution times of SQL queries that have ended within the collection cycle are not calculated. A shorter collection cycle may decrease the margin of error, but considering the database load, it is typically set to 5 seconds. It is useful for monitoring long-running queries. The same is true for the data in **SQL statistics**.



• Wait Event

It displays a real-time graph for the names of the waiting events that each backend process is waiting on, along with the frequency with which those events occurred. It mainly monitors waiting events related to application, configuration, administrative, network, and the like. You can visually see the frequency of each waiting event. If you hover your mouse over the graph, a tooltip is displayed providing more information about each waiting event.

• Physical Reads

This widget monitors physical reads occurring in the database. It monitors the I/O performance of the database, by providing statistics on the operations where the database reads data from the disk.

• Excute Count

This widget monitors the numbers of queries or procedures executed in the database. It traces the numbers of queries or procedures executed by the database in real time and provides them to the user.

• Session Logical Reads

This widget monitors logical reads executed in the database session. It traces and provides real-time visibility

to users on the number of logical read operations executed in the database session.

Session Table widget

You can view real-time active sessions and lock trees in the [Session Table](#) widget at the bottom of the screen.

- **Active sessions:** You can view active sessions and queries.
 - If you select a session running, the **Session detail** window appears. You can see the details for the selected session. For more information, see [the following](#).
 - If you select an item in the rightmost [query_text](#) column in the table, the **SQL details** window appears. You can see the full text and plan information for the selected query. For more information, see [the following](#).

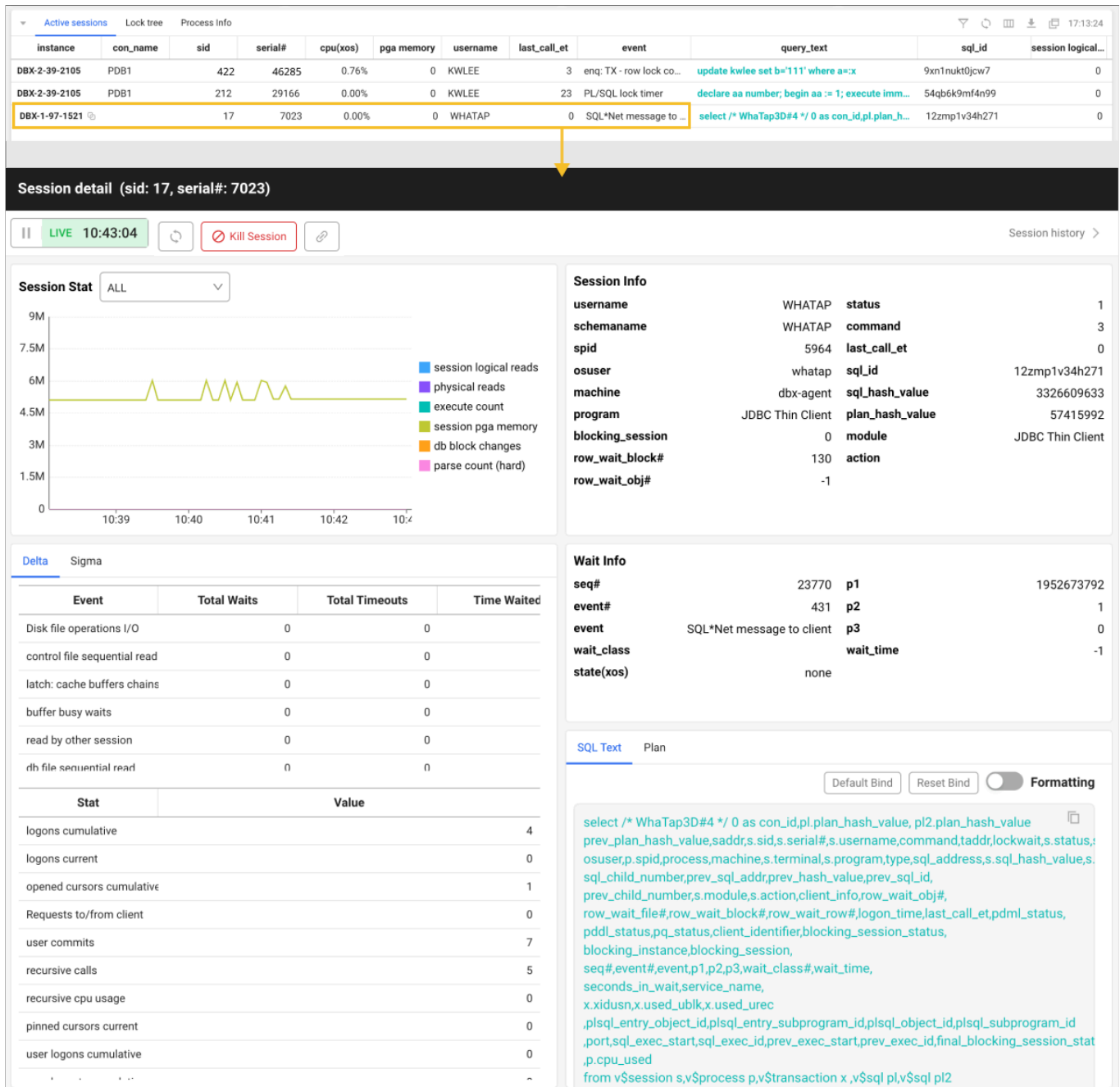
- **Lock tree**

You can see the lock session information, and holder and waiter information in real time. For more information about the table columns, see [the following](#).

ⓘ In the table list, the text colors are changed black → orange → Red, which means that the performance of the session is getting slower.

Checking the session information details

If you select a session in the **Active sessions** tab, the **Session detail** window appears. You can see the information details about the selected session.



- The session information is viewed in real time by default. If you select || in the time selector, you can stop a real-time view and see the data based on the paused time. To refresh data while pausing a real-time view, select ↻.
- Kill Session:** You can stop the selected session. If the **Session kill** window appears, enter the password

([Parameter Key](#)) and then select **Apply**.

! This feature is only available in real-time viewing mode.

- **Session history:** You can go to the **Session History** menu to see the records of the selected session.

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

- **Session Stat:** It provides various performance metrics and statistical information for the selected session. It includes the type and number of tasks performed in the session, resource usage, and more.
- **Delta / Sigma:** The change rate and statistical analysis information for the selected session appear. You can check the amount of change in session activities.
- **Session Info:** You can see the basic information of the selected session and the information details including the status.
- **Wait Info:** The waiting event and waiting time of the selected session are provided. You can also see information about the waiting events that occur when the database system is waiting to execute a specific task.
- **SQL Text / Plan:** You can see the SQL query statements and plans of the selected session. For more information, see [the following](#).

View SQL Details

Click the **View SQL Statistics →** button to move to **SQL statistics**, where you can view statistical information related to the selected SQL query.

ⓘ time. If the number of cases is too large, it displays up to 100.

AI Tuning Guide

The AI Tuning Guide analyzes SQL queries, plans, and statistical information to diagnose performance issues and suggest optimization strategies.

It helps developers and DBAs quickly identify bottlenecks and improve performance through efficient SQL optimization.

ⓘ Usage Conditions and Notes

For **PostgreSQL, MySQL, and SQL Server**, query plan retrieval is required.

If the plan is not retrieved, the AI Tuning Guide icon appears as 🛑 (disabled), and the feature cannot be used.

- Please note that AI-generated results are based on automated analysis and may not be 100% accurate.

1. Click the SQL you want to analyze and diagnose to move to the **SQL details** screen.
2. On the **SQL details** screen, click the AI Tuning Guide icon 🛑 at the bottom right to start the AI analysis.
3. Review the AI analysis results.

Result Item	Description
Query Plan and Summary	Provides the purpose and execution summary of the query. Analyzes execution count, cumulative execution time, and overall database load ratio to assess the query's impact on system performance.
Performance Analysis	Provides performance scores and diagnostic results based on overall analysis. Analyzes detailed resource usage such as CPU, disk, cache hit rate, and wait time to visually identify bottleneck areas during query execution.
Key Issues Found	Summarizes detected major issues.

Result Item	Description
Optimization Recommendations	Suggests optimized queries based on identified issues.

Viewing the default plan

When creating the **default plan user file**, you can configure the agent to automatically view the plan by using the predefined DB information. After checking the box for **Default plan inquiry**, search for the plan through the specified DB connection information as follows.

• Linux Windows

1. Add the `plan_db` option to the agent configuration ([whatap.conf](#)) file and then enter the DB name.

whatap.conf

```
plan_db=(dbname)
```

2. After configuring the DB, use the `planuid.sh` script to set the user and password.

```
./planuid.sh user pwd
```

❗ If the user (`user`) and password (`pwd`) include special characters, add backslashes (`\`) before each special character.

e.g. `user$example` → `user\$example`

1. Add the `plan_db` option to the agent configuration ([whatap.conf](#)) file and then enter the DB name.

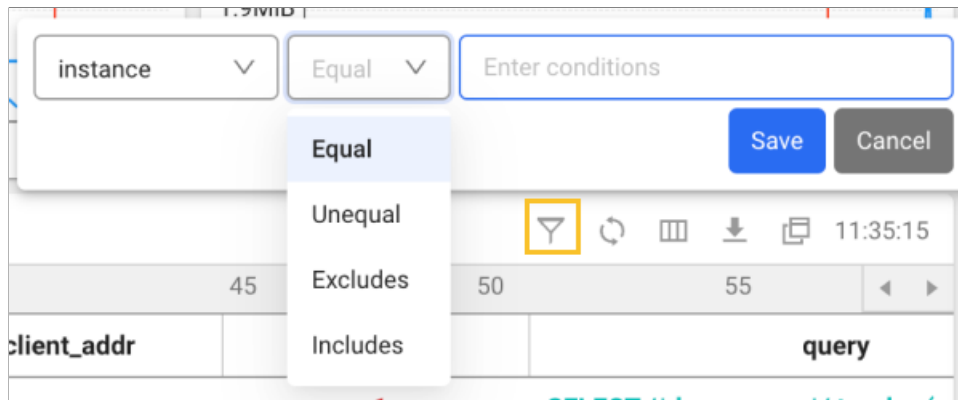
whatap.conf

```
plan_db=(dbname)
```

2. After configuring the DB, use the `planuid.bat` script to set the user and password.


```
planuid.bat user pwd
```

Filtering the table data



1. Select  on the upper right of the table.
2. Select a column header and a condition in the table.
3. Enter a desired value in the **Enter conditions** field.
4. Select **Save**.

Setting the table columns

You can hide the table header columns or add any of them. You can also change the column order. Select .

- After configuration, select **Confirm** to apply the settings in the table.
- In the  search bar, enter text to search the desired columns. Only the columns that meet the entered text are displayed.
- Images may differ depending on the product, project, or menu.

Adding columns

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

Deleting columns

From the ① list, unselect the columns to delete. Alternatively, select ✕ on the right of the item to delete from the ② list.

Changing the column order

Drag an item to reposition from the ② list, and then move it to the desired position.

Initializing the configuration

To cancel all changes and reset them, select ↺ **Initialized**.

Column information guide

For more information about columns, see [the link](#).

Item	Description
sid	Session ID
cpu(xos)	CPU utilization collected by the installed XOS
serial#	Value to increment to determine if the same session ID has been used.
username	DB user name.
command	Same as <code>COMMAND_TYPE</code> ; SQL command number
status	Session status
schemaname	Schema. Same as the username.
osuser	Username of the OS that accessed the session.
spid	OS process ID.

Item	Description
process	ID of the client process that accessed the session.
logon_time	Session created time point.
last_call_et	If the session is active, it is the elapsed time (sec) since activation.
sql_text	query, running statement, SQL statement that separates literal strings
sql_param	Literal value separated from query
prev_sql_text	Previously executed query
prev_sql_param	Previously executed query param
program	Program information executed by client
type	session type - can be adjusted in user background recursive (whatap.conf).
seq#	Number that uniquely identifies the current or last wait (incremented with each wait)
event	Number of the resource or event if the session is waiting.
p1	First waiting event parameter (decimal point)
p2	Second waiting event parameter (decimal point)
p3	Third waiting event parameter (decimal point)
wait_class	Class name of the waiting event
wait_time	The value is 0 if the session is waiting. It is the last waiting time (milliseconds) if the value is greater than 0.
seconds_in_wait	Time spent waiting for the wait if the session is waiting. Otherwise, it is the time elapsed since the last waiting.

Item	Description
state	Waiting state: WAITING , WAITED UNKNOWN TIME , WAITED SHORT TIME , WAITED KNOWN TIME
machine	Name of the client server that accessed the session (hostname)
port	Client port number
terminal	Client terminal name
module	Name of the module set by the user of the program executed on the client.
action	Action name set by the user of the running module.
client_info	Client information set by user
client_identifier	Client ID set by user
session logical reads	Number of times data was read into the buffer cache
physical reads	Number of data reads from the physical disk
execute count	SQL execution count
parse count (hard)	Total number of syntax analysis calls (actual syntax analysis)
parse count (total)	Total number of syntax analysis calls (hard, soft, describe)
opened cursors current	Number of open cursors
db block changes	Change count of all blocks in the database
session pga memory	Session's current PGA size
undo_segid	Undo information ID
undo_blk	Undo block count

Item	Description
<code>undo_rec</code>	Undo record count
<code>sql_address</code>	SQL statement identification data
<code>sql_hash_value</code>	SQL statement identification data
<code>sql_id</code>	SQL ID of the running SQL statement
<code>sql_child_number</code>	Sub number of the running SQL statement
<code>sql_exec_start</code>	Start time point of SQL running in the session
<code>sql_exec_id</code>	SQL running ID
<code>prev_sql_addr</code>	Information that identifies the last run SQL statement
<code>prev_child_number</code>	Sub number of the last SQL statement executed
<code>prev_exec_start</code>	Start time point of the last SQL statement executed
<code>prev_exec_id</code>	Start ID of the last SQL statement executed
<code>plsql_entry_object_id</code>	Object ID of the top PL/SQL subprogram on the stack.
<code>plsql_entry_subprogram_id</code>	ID of the top PL/SQL subprogram on the stack.
<code>plsql_object_id</code>	Subprogram ID of the running PL/SQL object
<code>taddr</code>	Transaction address
<code>lockwait</code>	lock wait address (lock address where the session is waiting)
<code>row_wait_obj</code>	Object ID of the table containing the row specified in <code>ROW_WAIT_ROW#</code>
<code>row_wait_file</code>	ID of the data file containing the row specified in <code>ROW_WAIT_ROW#</code>

Item	Description
row_wait_block	ID of the block containing the row specified in ROW_WAIT_ROW#
row_wait_row	Currently locked row
pdml_status	If the value is ENABLED , the session is in PARALLEL DML activation mode. If the value is DISABLED , the session does not support the PARALLEL DML activation mode. If the value is DISABLED , the session does not support the PARALLEL DML activation mode.
pq_status	If the value is ENABLED , the session is in PARALLEL QUERY activation mode. If the value is DISABLE , the session does not support the PARALLEL QUERY activation mode. If the value is FORCED , it indicates the session was changed to force the use of PARALLEL QUERY .
blocking_session_status	It provides details about whether there is a blocking session.
blocking_instance	Instance ID for the blocking session
blocking_session	Session ID for the blocking session
final_blocking_session_status	It provides details about whether there is a last block session.
final_blocking_instance	Instance ID for the last blocking session
final_blocking_session	Session ID for the last blocking session
service_name	Service name of the session. It is similar to the DB name.
saddr	Session address
con_id	ID of the container associated with the data (only for multi-database support)
rss(xos)	OS metrics collected by the XOS agent
pss(xos)	OS metrics collected by the XOS agent

Item	Description
ioread(xos)	OS metrics collected by the XOS agent
iowrite(xos)	OS metrics collected by the XOS agent

! WhaTap basically stores the client-related information.

Downloading the view list

You can download the current [Session Table](#) list in CSV format. Select  on the upper right of the [Session Table](#) section.

The naming format of the downloaded file is as follows:

- **Active sessions:** `activeSession_HH_MM_SS.csv`
- **Lock tree:** `lockTree_HH_MM_SS.csv`

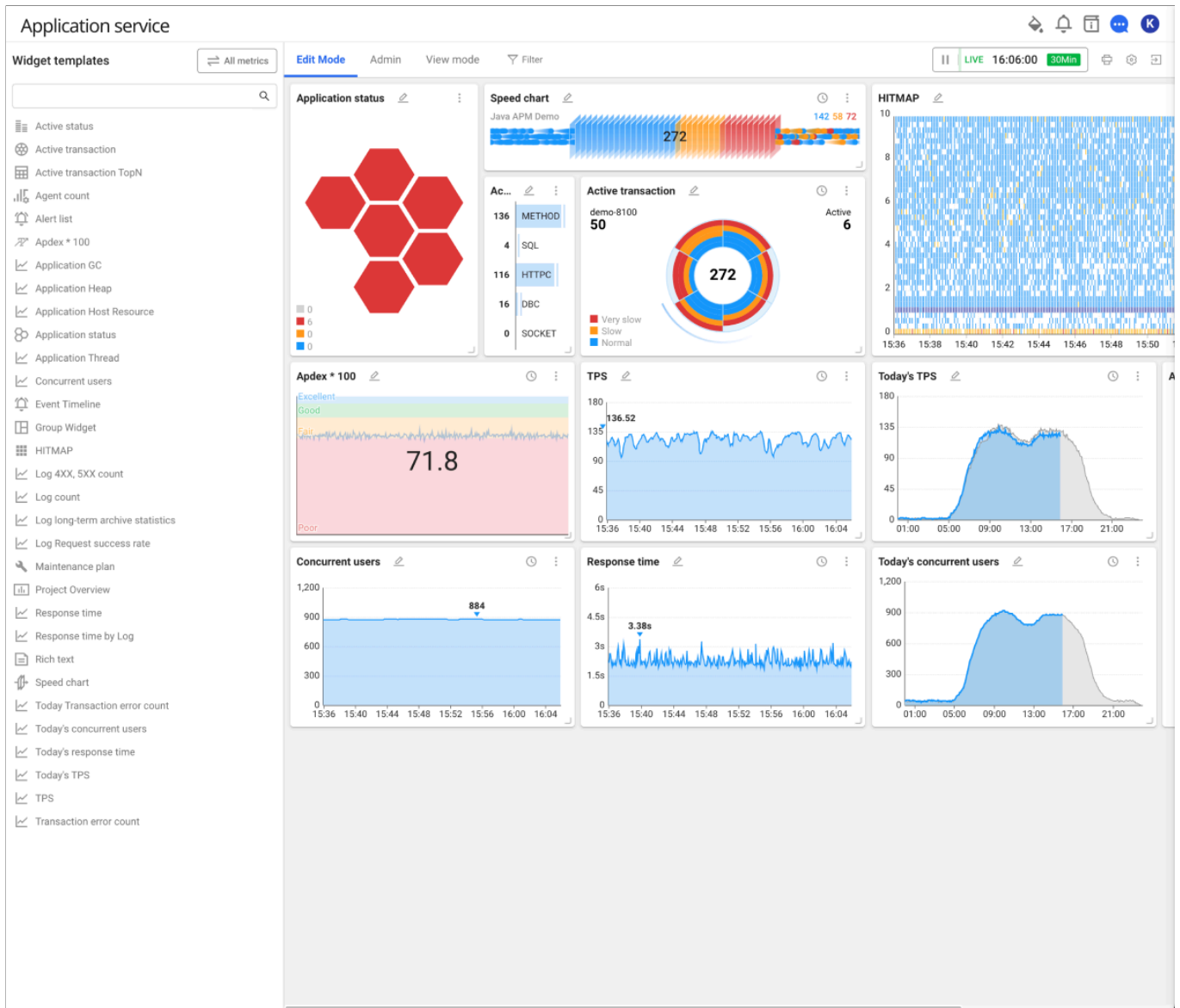
Viewing in a new window

You can see the [Session Table](#) list on a new wider screen. Select  on the upper right of the [Session Table](#) section.

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.



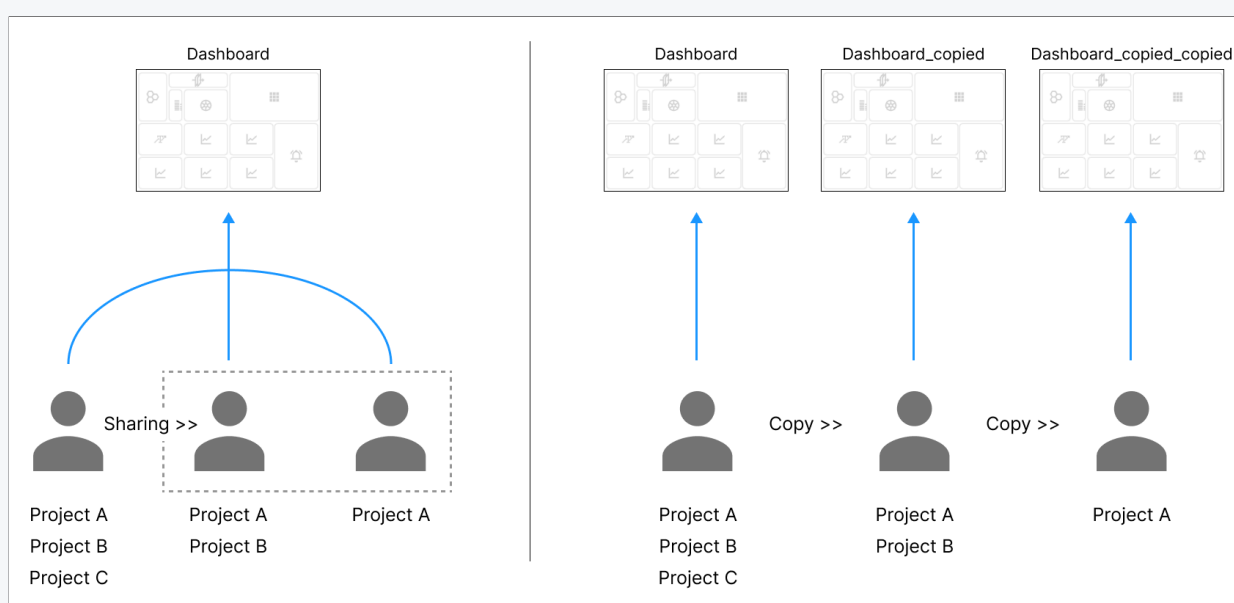
- If a user who received a shared dashboard deletes it, it is deleted only from that user's **Dashboard list**. The original dashboard remains.

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 to 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**

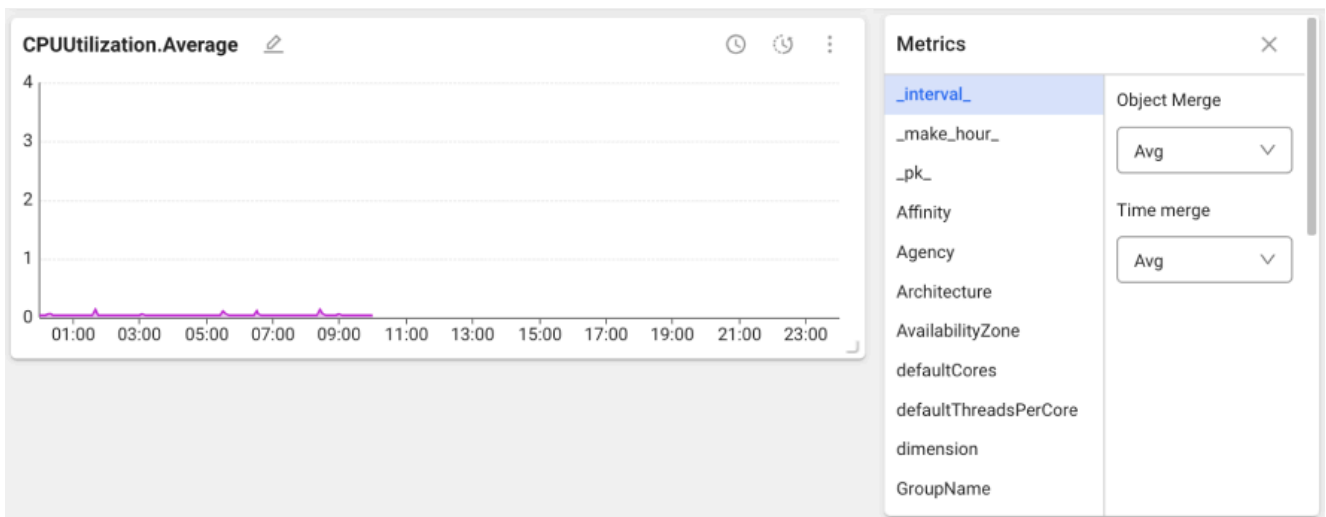
It is used to group the chart data based on the tag (e.g. agent name) selected on the left. At this time, if the tags match among the data with different fields (e.g. CPU), the corresponding data is merged.

- **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**.

Trend Comparison

Home > Select Project > **Analysis** > **Trend Comparison**

In **Analysis** > **Trend Comparison**, you can check key database metrics by time period and track performance changes.

By comparing metrics across different dates, you can analyze operational trends.

The Active sessions table allows you to view both currently running sessions and long-running sessions.

Basic screen guide



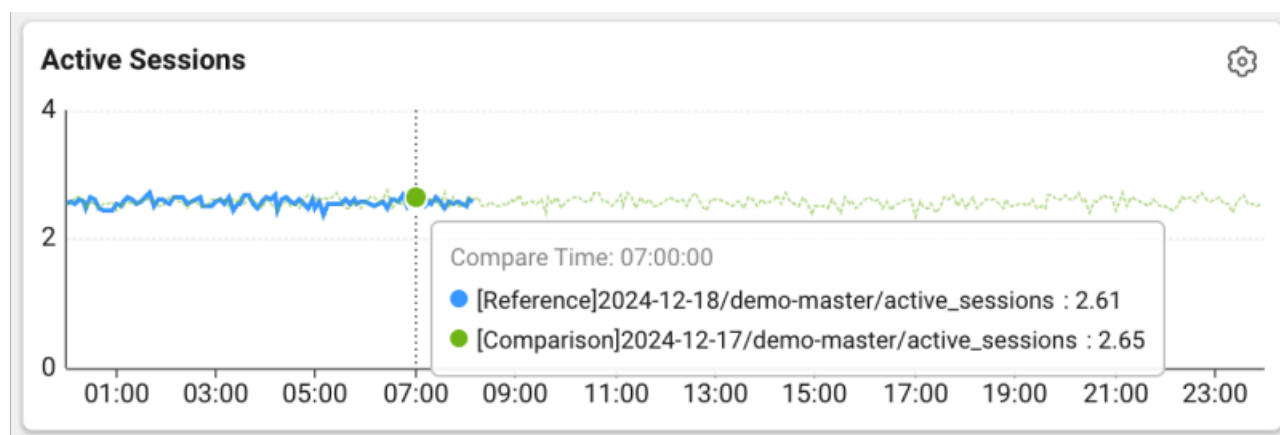
- If you select **1 Base date**, **2 Comparison date**, and **Instance**, the data is automatically applied to all widgets based on the selected items.
- To see the meaning of the metric displayed in the widget, select ⓘ next to the name or see [the following](#).

❗ The tooltip of the ⓘ button is not supported in multiple languages.

- To change the position with another widget, select and drag the upper part of the widget. However, the size of the widget cannot be changed.
- The left of the ⓘ Active sessions table displays the data for the **Base date**, and the right of the table displays the data for the **Comparison date**.
- In the ⓘ Active sessions table, the text colors are changed black → orange → Red, which means that the performance of the session is getting slower.

Selecting a comparison target

Select ⓘ **Base date** and **Instance**, and then select ⓘ **Comparison date** and **Instance**. You can check the data details at the time point through a tooltip that appears by hovering the mouse over the chart. The content of the tooltip is updated based on the position of the mouse.



- Each series is given a unique color for easy visual distinction. The solid line represents the data in **Base date**, and the dotted line represents the data in **Comparison date**.
- **Compare Time** displays the time value of the data where the mouse pointer is located.
- It displays the metric value at the time point of the selected data, and provides the data by series. Each series includes the data point date, agent (instance) name, metric name, and the value.

❗ • You can also select a desired database to compare from **Database** at the top of the screen.



- The color of **Database** at the top of the screen is displayed in the same color on the graph chart.

Comparing the active session data

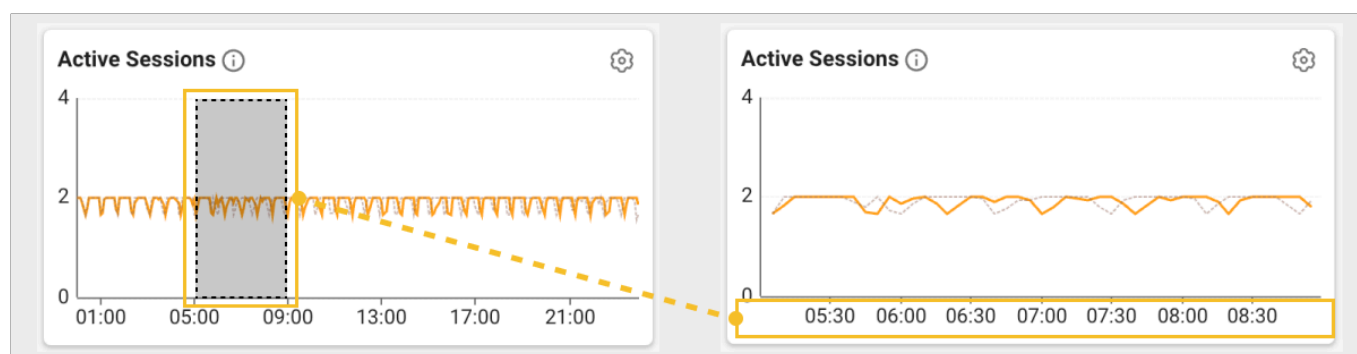
Reference (2024-12-08)																			Comparison (2024-12-06)																							
1	4	8	11	14	17	20	23	26	29	32	35	38	41	44	47	50	53	56	59	00:00:26	0	4	7	10	13	16	19	22	25	28	31	34	37	40	43	46	49	52	55	58	61	00:00:31
instance		con_name		sid		serial#		cpu(xos)		pga memory		username		last_call_		instance		con_name		sid		serial#		cpu(xos)		pga memory		username		last_call_												
DMX-ol9-23.localdior		FREEPDB1		219		23347		0.00%		1513656		KWLEE				DMX-ol9-23.localdior		FREEPDB1		216		5587		1.81%		1513656		KWLEE														
DMX-ol9-23.localdior		FREEPDB1		216		26370		0.00%		1513656		KWLEE				DMX-ol9-23.localdior		FREEPDB1		196		55263		3.38%		1513656		KWLEE														
DMX-ol9-23.localdior		FREEPDB1		213		17075		0.00%		1513656		KWLEE				DMX-ol9-23.localdior		FREEPDB1		57		32534		0.00%		1579192		KWLEE														
DMX-ol9-23.localdior		FREEPDB1		211		16582		0.00%		1513656		KWLEE				DMX-ol9-23.localdior		FREEPDB1		46		3972		0.00%		1513656		KWLEE														
DMX-ol9-23.localdior		FREEPDB1		162		46247		0.00%		1579192		KWLEE				DMX-ol9-23.localdior		FREEPDB1		35		63001		16.66%		1513656		KWLEE														
DMX-ol9-23.localdior		FREEPDB1		55		57240		0.00%		1513656		KWLEE																														
DMX-ol9-23.localdior		FREEPDB1		37		14205		0.00%		1513656		KWLEE																														

- The time zone for data retrieval can be checked in 4.
- Active session data is collected every 5 seconds. You can search data for the desired time by selecting the button that moves in 5-second increments on the table.
- If you select ◀ or ▶ on the upper right, you can see the data in 1-minute increments. To move in increments of 5 seconds, select ◀ or ▶.
- In the Active sessions table list, the text colors are changed black → orange → Red, which means that the performance of the session is getting slower.
- If you click a specific time point of the graph chart, the (3) area appears with a red line and the collected active sessions can be also seen.



- When a session is selected in the session table list, you can see **Session detail**.

Comparing with a specific time



You can compare the data by zooming in on a specific time point. Select a desired time zone to anywhere on the widgets and then drag the chart. It displays the data for a specific time zone dragged to all widgets and active session tables.

Preset

In **Monitoring a Database Instance**, you can load the custom widget settings and the sorted active sessions as presets.

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

Column information guide

For more information about columns, see [the link](#).

Item	Description
sid	Session ID
cpu(xos)	CPU utilization collected by the installed XOS

Item	Description
serial#	Value to increment to determine if the same session ID has been used.
username	DB user name.
command	Same as <code>COMMAND_TYPE</code> ; SQL command number
status	Session status
schemaname	Schema. Same as the username.
osuser	Username of the OS that accessed the session.
spid	OS process ID.
process	ID of the client process that accessed the session.
logon_time	Session created time point.
last_call_et	If the session is active, it is the elapsed time (sec) since activation.
sql_text	<code>query</code> , running statement, SQL statement that separates literal strings
sql_param	Literal value separated from <code>query</code>
prev_sql_text	Previously executed query
prev_sql_param	Previously executed query param
program	Program information executed by client
type	session type - can be adjusted in user background recursive (whatap.conf).
seq#	Number that uniquely identifies the current or last wait (incremented with each wait)
event	Number of the resource or event if the session is waiting.

Item	Description
p1	First waiting event parameter (decimal point)
p2	Second waiting event parameter (decimal point)
p3	Third waiting event parameter (decimal point)
wait_class	Class name of the waiting event
wait_time	The value is 0 if the session is waiting. It is the last waiting time (milliseconds) if the value is greater than 0.
seconds_in_wait	Time spent waiting for the wait if the session is waiting. Otherwise, it is the time elapsed since the last waiting.
state	Waiting state: WAITING , WAITED UNKNOWN TIME , WAITED SHORT TIME , WAITED KNOWN TIME
machine	Name of the client server that accessed the session (hostname)
port	Client port number
terminal	Client terminal name
module	Name of the module set by the user of the program executed on the client.
action	Action name set by the user of the running module.
client_info	Client information set by user
client_identifier	Client ID set by user
session logical reads	Number of times data was read into the buffer cache
physical reads	Number of data reads from the physical disk
execute count	SQL execution count

Item	Description
parse count (hard)	Total number of syntax analysis calls (actual syntax analysis)
parse count (total)	Total number of syntax analysis calls (hard, soft, describe)
opened cursors current	Number of open cursors
db block changes	Change count of all blocks in the database
session pga memory	Session's current PGA size
undo_segid	Undo information ID
undo_blk	Undo block count
undo_rec	Undo record count
sql_address	SQL statement identification data
sql_hash_value	SQL statement identification data
sql_id	SQL ID of the running SQL statement
sql_child_number	Sub number of the running SQL statement
sql_exec_start	Start time point of SQL running in the session
sql_exec_id	SQL running ID
prev_sql_addr	Information that identifies the last run SQL statement
prev_child_number	Sub number of the last SQL statement executed
prev_exec_start	Start time point of the last SQL statement executed
prev_exec_id	Start ID of the last SQL statement executed

Item	Description
<code>plsql_entry_object_id</code>	Object ID of the top PL/SQL subprogram on the stack.
<code>plsql_entry_subprogram_id</code>	ID of the top PL/SQL subprogram on the stack.
<code>plsql_object_id</code>	Subprogram ID of the running PL/SQL object
<code>taddr</code>	Transaction address
<code>lockwait</code>	lock wait address (lock address where the session is waiting)
<code>row_wait_obj</code>	Object ID of the table containing the row specified in <code>ROW_WAIT_ROW#</code>
<code>row_wait_file</code>	ID of the data file containing the row specified in <code>ROW_WAIT_ROW#</code>
<code>row_wait_block</code>	ID of the block containing the row specified in <code>ROW_WAIT_ROW#</code>
<code>row_wait_row</code>	Currently locked row
<code>pdml_status</code>	If the value is <code>ENABLED</code> , the session is in <code>PARALLEL DML</code> activation mode. If the value is <code>DISABLED</code> , the session does not support the <code>PARALLEL DML</code> activation mode. If the value is <code>DISABLED</code> , the session does not support the <code>PARALLEL DML</code> activation mode.
<code>pq_status</code>	If the value is <code>ENABLED</code> , the session is in <code>PARALLEL QUERY</code> activation mode. If the value is <code>DISABLE</code> , the session does not support the <code>PARALLEL QUERY</code> activation mode. If the value is <code>FORCED</code> , it indicates the session was changed to force the use of <code>PARALLEL QUERY</code> .
<code>blocking_session_status</code>	It provides details about whether there is a blocking session.
<code>blocking_instance</code>	Instance ID for the blocking session
<code>blocking_session</code>	Session ID for the blocking session
<code>final_blocking_session_status</code>	It provides details about whether there is a last block session.

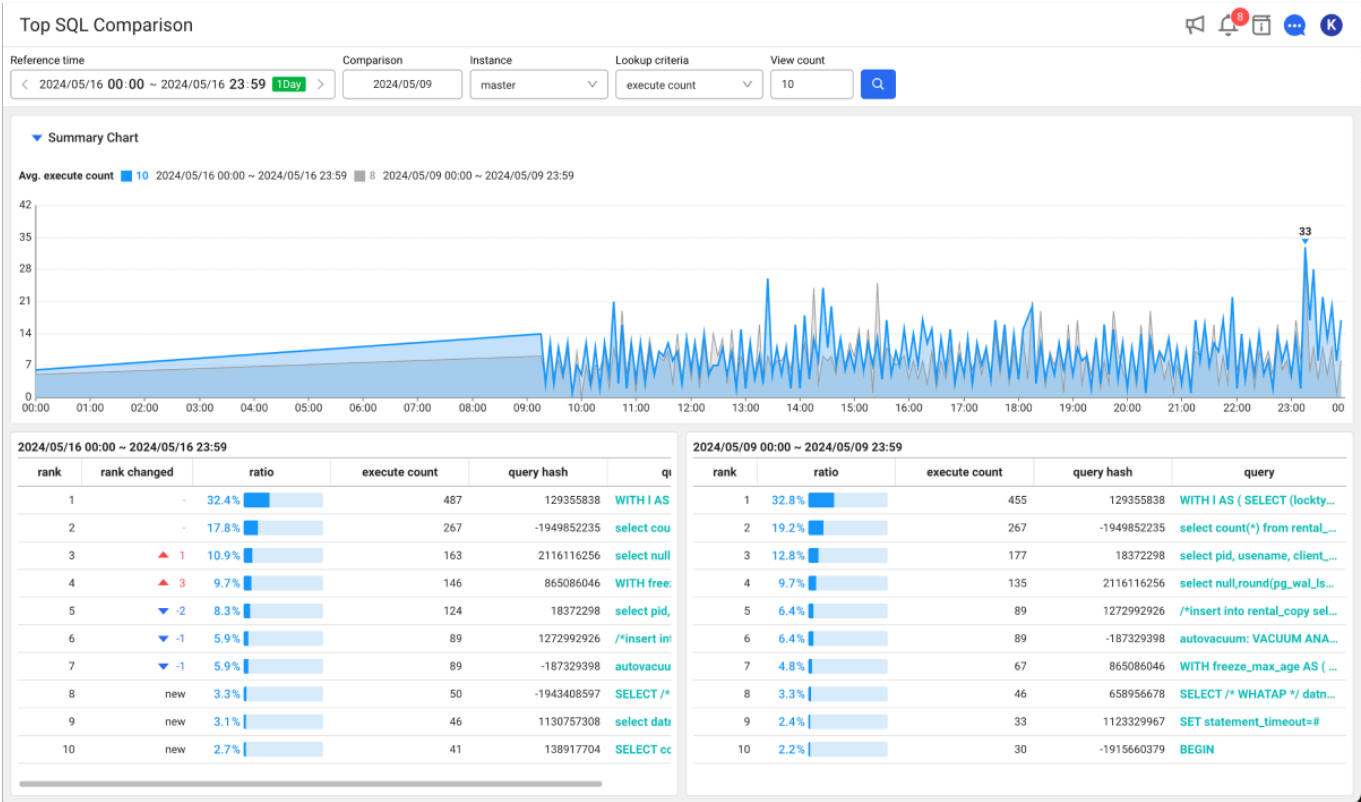
Item	Description
<code>final_blocking_instance</code>	Instance ID for the last blocking session
<code>final_blocking_session</code>	Session ID for the last blocking session
<code>service_name</code>	Service name of the session. It is similar to the DB name.
<code>saddr</code>	Session address
<code>con_id</code>	ID of the container associated with the data (only for multi-database support)
<code>rss(xos)</code>	OS metrics collected by the XOS agent
<code>pss(xos)</code>	OS metrics collected by the XOS agent
<code>ioread(xos)</code>	OS metrics collected by the XOS agent
<code>iowrite(xos)</code>	OS metrics collected by the XOS agent

ⓘ WhaTap basically stores the client-related information.

Top SQL Comparison

Home > Select Project > Analysis > Top SQL Comparison

You can compare the SQL trends in the time zone set based on a specific date. You can compare the performance of SQL by selecting multiple criteria and view it as a time series graph via **Summary Chart**. Additionally, the table list at the bottom of the screen allows you to compare the **Top SQL** rankings and fluctuations between the reference date and the comparison date.



- **Check SQL performance:** You can easily check the SQL performance through the **Top SQL Comparison** menu. You can quickly identify performance issues in your system by comparing the number of SQL executions, maximum elapsed time, total elapsed time, waiting time, and more.
- **Check trends over time:** You can visually check the performance changes of SQL over time through **Summary Chart**. This allows users to quickly identify and respond to time zones or sections of system load or performance degradation.

- **Top SQL comparison:** The table list at the bottom of the screen allows you to compare the Top SQL rankings for the reference date and the comparison date. This allows users to see and compare performance changes in SQL and take necessary actions to improve the performance.

Comparing Top SQLs

Set the conditions for Top SQL comparison on the screen and then select .

- **Reference time:** Set the reference time for comparison. It can be set up to 24 hours. By clicking the green button, the options to select a viewing time appear. Select the date and time area to set your own preferred time.

- 
- The maximum time range that can be viewed is 24 hours.
 - For more information on how to use the **Reference time** option.

- **Comparison:** Select a date or time to compare with the time set in **Reference time**. Select the desired conditions and then select **Confirm**.

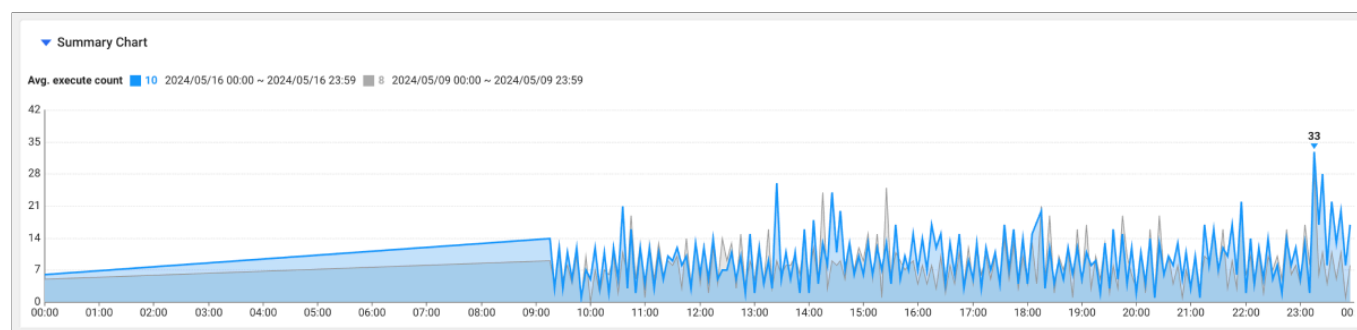
Date	Hour
Comparison 2024/05/16 Instance master Date Hour 1 Days Ago 2024/05/16 7 Days Ago 2024/05/10 30 Days Ago 2024/04/17 Custom 2024-05-16 Cancel Confirm	Comparison 2024/05/16 Instance master Date Hour (n)hours ago 1 Cancel Confirm

- **Instance:** Select an instance target for SQL comparison.

- **Lookup criteria:** Select the comparison criteria. It generates a chart and then sorts them by rank according to the selected items.
 - **excute count:** It sorts query results by the number of executions.
 - **elapsed max:** It sorts searched results by the maximum elapsed time.
 - **elapsed time:** It sorts searched results by the total elapsed time.
 - **elapsed wait:** It sorts the searched results by the waiting time.
- **View count:** Set the count to display in the SQL ranking table list at the bottom of the screen.

Summary Chart

Summary Chart provides a time series graph that allows you to compare SQL trends between the reference date and a comparison date. This allows users to easily check how SQL performance changes over time.



- **Blue graph:** It indicates the reference date and the number on the right is the average.
- **Gray graph:** It indicates the comparison date, and the number on the right is the average.

Comparing SQL rankings

The SQL ranking table at the bottom of the screen displays a list of the top SQLs. The displayed number can be set in **View count**.

2024/05/16 00:00 ~ 2024/05/16 23:59

rank	rank changed	ratio	execute count	query hash	qi
1	-	32.4%	487	129355838	WITH I AS
2	-	17.8%	267	-1949852235	select cou
3	▲ 1	10.9%	163	2116116256	select null
4	▲ 3	9.7%	146	865086046	WITH free:
5	▼ -2	8.3%	124	18372298	select pid,
6	-	5.9%	89	-187329398	autovacuu
7	▼ -2	5.9%	89	1272992926	/*insert ini
8	new	3.3%	50	-1943408597	SELECT /*
9	new	3.1%	46	1130757308	select datr
10	new	2.7%	41	138917704	SELECT cc

2024/05/09 00:00 ~ 2024/05/09 23:59

rank	ratio	execute count	query hash	query
1	32.8%	455	129355838	WITH I AS (SELECT (lockty...
2	19.2%	267	-1949852235	select count(*) from rental_...
3	12.8%	177	18372298	select pid, username, client_...
4	9.7%	135	2116116256	select null,round(pg_wal_ls...
5	6.4%	89	1272992926	/*insert into rental_copy sel...
6	6.4%	89	-187329398	autovacuum: VACUUM ANA...
7	4.8%	67	865086046	WITH freeze_max_age AS (...
8	3.3%	46	658956678	SELECT /* WHATAP */ datn...
9	2.4%	33	1123329967	SET statement_timeout=#
10	2.2%	30	-1915660379	BEGIN

The table on the left displays the SQL rankings for the reference date, and the table on the right displays the SQL rankings for the comparison date. You can check the changes in rank compared to the comparison date through the **rank changed** column in the reference date table.

For example, if a user compares based on the **elapsed time** and the elapsed time for the same SQL statement on the reference date is greater than the comparison date, the performance of the SQL statement needs to be reviewed. Try using query optimization or indexes to optimize the performance.

Previewing the query

Previewing the query

You can check the query of the corresponding SQL statement by hovering the mouse over the **query** column in the SQL ranking table.

❗ If you select **query** column from the SQL rank list, the **SQL details** window appears.

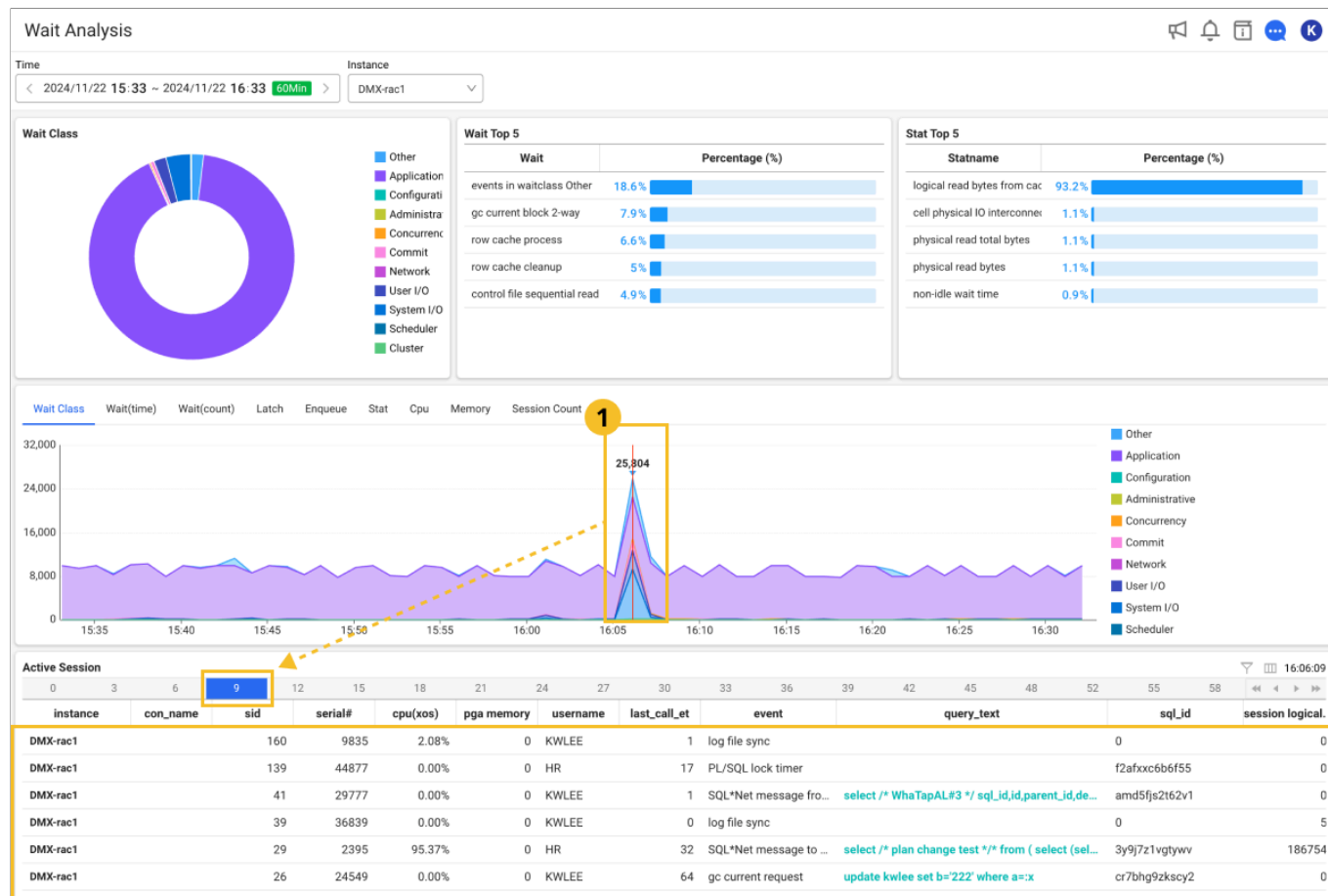
Wait Analysis

Home > Select Project > **Analysis** > **Wait Analysis**

You can check in which wait event the active session is waiting and which wait events occur frequently. Active sessions collected every 5 seconds can be analyzed from the wait perspective.

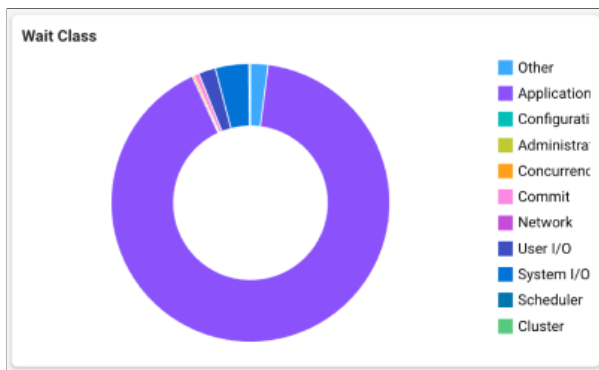
- ⓘ • This function is supported in DBX agent 1.9.0 or later.
- The collected wait and stat information is based on the `v$system_wait_class` , `v$system_event` , and `v$sysstat` views.
For detailed descriptions of each metric, see the [following](#).

Select the **Time** and **Instance** to view. You can see the wait data corresponding to the set time and instance. If there are active sessions that frequently wait (delay) during the query period, it can be used to analyze what types of wait events occurred.



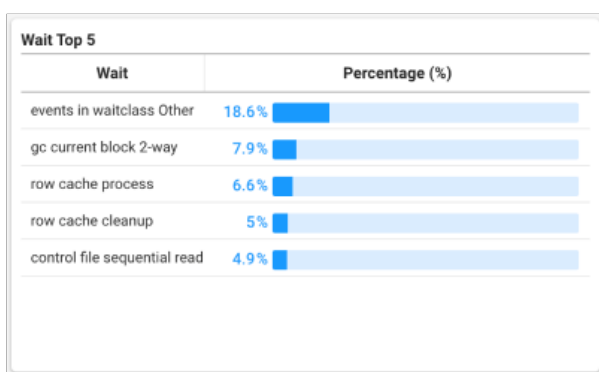
The center time-series chart visually displays the distribution of wait classes and number of sessions by time zone. This allows you to identify patterns of increased waiting events in a specific time zone. You can select analysis categories such as [Wait \(time\)](#), [Wait \(count\)](#), [Latch](#), [Enqueue](#), [Stat](#), [Cpu](#), [Memory](#), and [Session Count](#).

If you select the **1** detailed time in the time-series chart, you can view the active sessions in the time and check the SQL statements in which the wait events occurred.



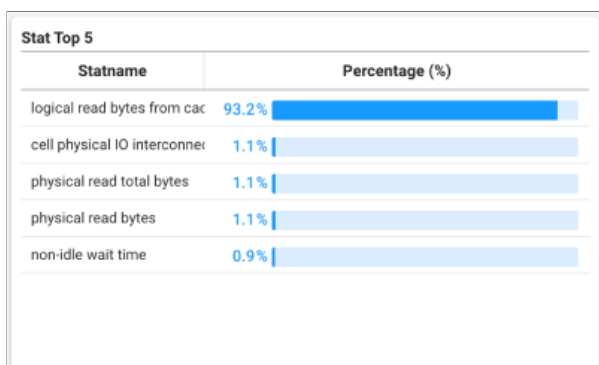
Wait Class

It visualizes wait events for each wait class with a pie chart. This allows you to quickly identify the causes of major wait events. If the wait class is abnormally high, focus on checking the resources related to the class.



Wait Event Top 5

It displays the top 5 frequent wait events. The percentage of each wait event can help you determine which events have the greatest impact on system performance, which is useful for identifying bottlenecks in the system.



Stat Top 5

The top 5 key stat data are displayed in percentage (%). You can quickly find the resource usage status and which operation is having the greatest impact on the database.

- The wait analysis results are used as a tool to help diagnose their causes, and problem resolution must be performed at the DBA's discretion.
- You need to adjust the time range to accurately find the time zone when the problem occurs.
- If you become familiar with the characteristics of the wait classes and events in advance, it can speed up the analysis.

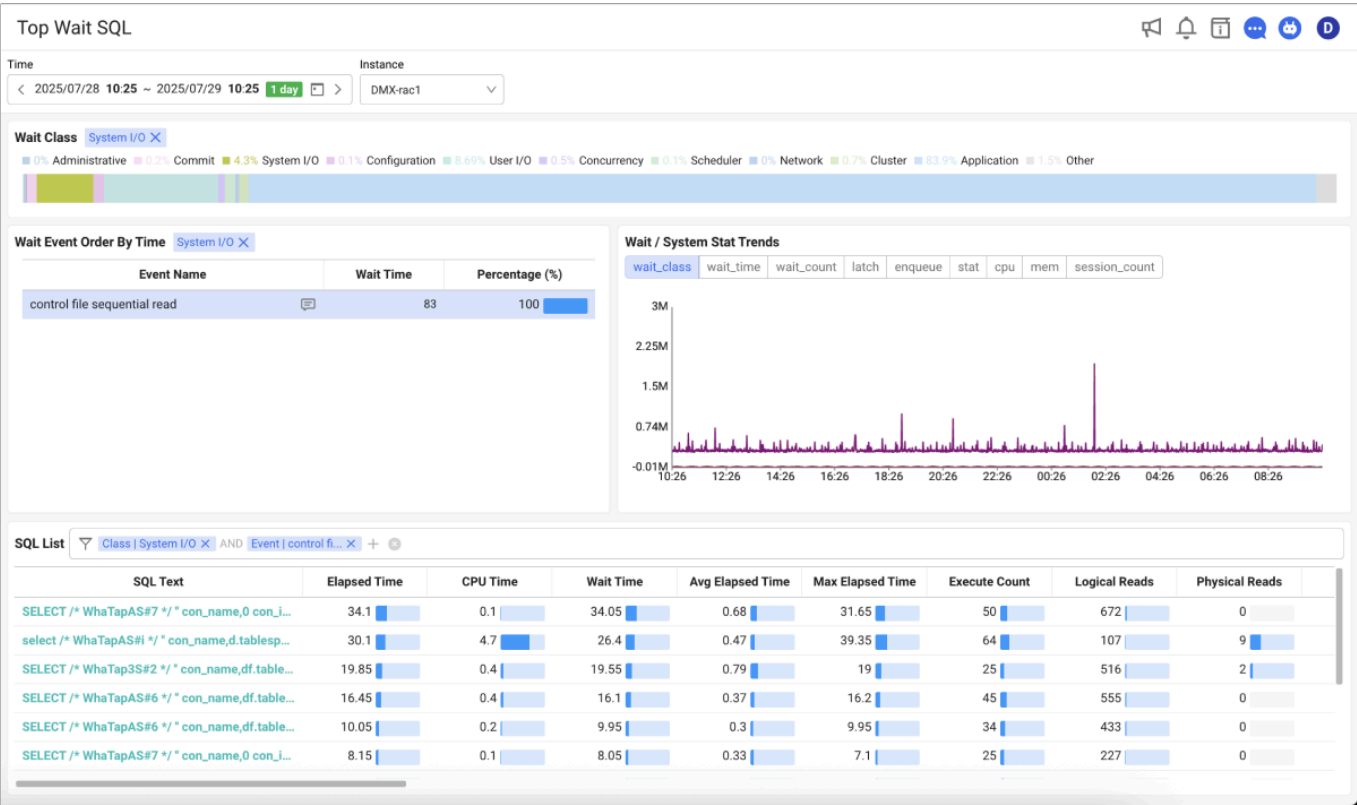
- When you select the [instance](#) column in the [Active Session](#) section list, the **Monitoring a Database Instance** menu appears. For more information about **Monitoring a Database Instance**, see [the following](#).
- In the [Active Session](#) list, if you select a session, the **Session detail** window appears. For more information about the **Session detail** window.
- When you select the [query_text](#) column in the [Active Session](#) list, the **SQL details** window appears. For more information about the **SQL details** window.
- You can hide the table header columns or add any of them. You can also change the column order.
- You can filter the table data to quickly find the desired item.

Top SQL By Wait

Home > Select Project > **Statistics** > **Top SQL by Wait**

Top SQL by Wait focuses on quickly identifying SQL tuning priorities. You can visually confirm the Wait Class and Wait Event with the most wait time during a specific time period through charts and tables, and compare execution count and average time in the Top SQL table.

Basic Screen Guide



Wait Class Chart

Clicking a class in the chart or legend will apply it as a filter. Clicking again will remove the filter.

- The Wait Class bar chart at the top visually displays the proportion of total wait time during the selected analysis period.
- The selected filters are shown as tags.

Wait Event List

If no class is selected, the most frequent events within the selected time range are displayed. When a Wait Class filter is applied, the event list for that class is shown.

When a **Wait Event** is selected from the list, it is added as a filter to the **SQL List**.

Wait / System Stat Trends

Visually displays the distribution of Wait Classes and session counts over time. You can identify spikes in wait events by time range. Analysis categories include: Wait(time), Wait(count), Latch, Enqueue, Stat, Cpu, Memory, Session Count.

SQL List

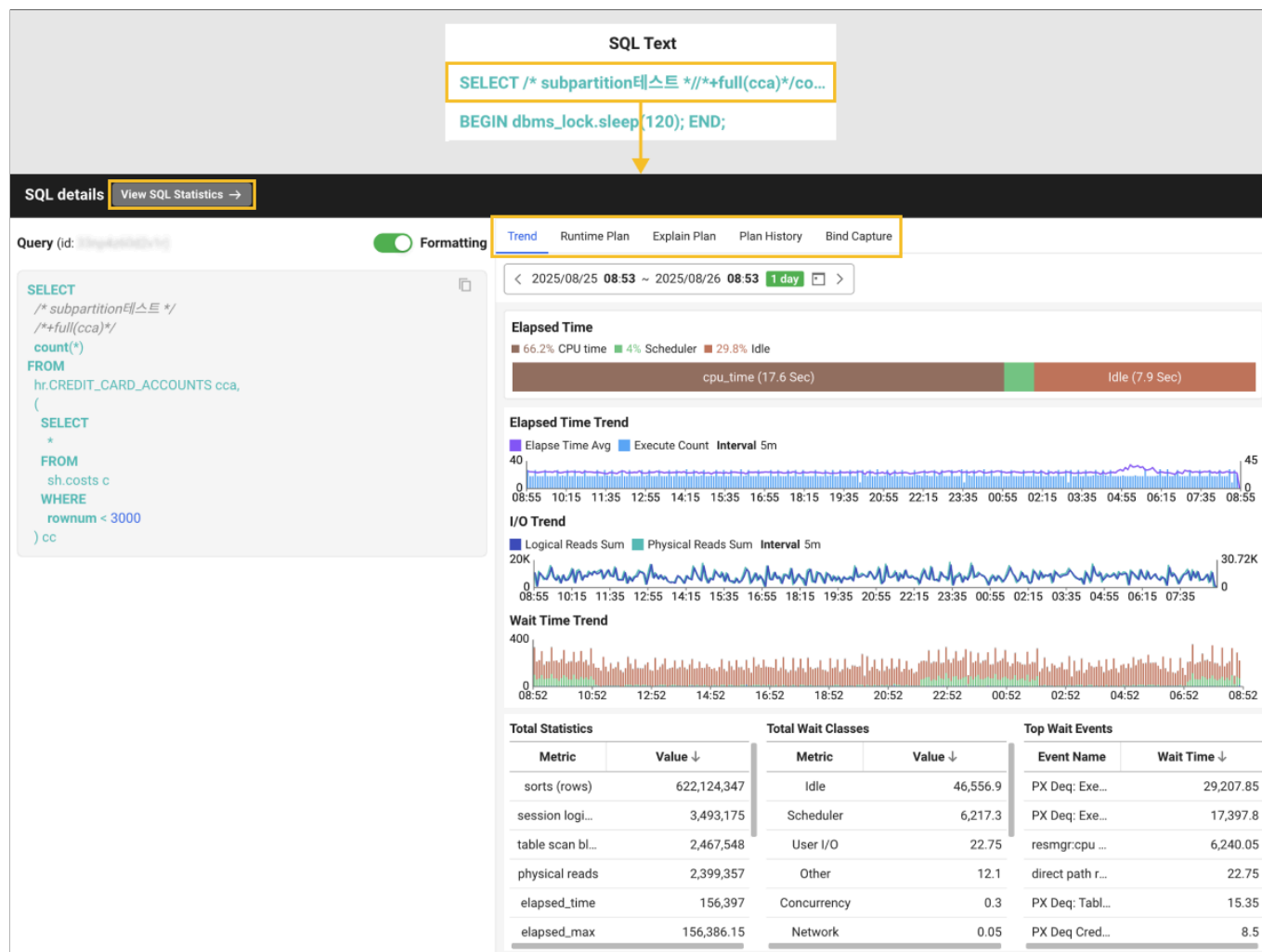
Key execution performance metrics are displayed for each SQL item, including Elapsed Time, CPU Time, Wait Time, Execute Count, and Logical Reads, enabling quick comparison and analysis.

- **Default (no filters selected):** Displays the top 50 queries by Elapsed Time.
- **When a class is selected:** Displays the top 50 queries by Wait Time for the selected class.

View Query Details

When you select the **query** column item in the retrieved result list, the **SQL Details** window appears.

Click **View SQL Statistics** → to move to the **SQL statistics** menu, where you can check the statistical information related to the SQL query.



The tabs of the **SQL Details** window are as follows:

- **Trend:** Analyze execution time, I/O, and wait events of a specific SQL comprehensively to quickly identify performance bottlenecks.
- **Runtime Plan:** Provides execution plan and runtime information for the selected SQL query. Includes details such as execution count, average execution time, and average physical reads.
- **Explain Plan:** Shows the execution plan predicted by the optimizer, including cost, operations, object names, and cardinality.
- **Plan History:** Check the history of execution plans for SQL queries that have been run in the database.
- **Bind Capture:** Displays the bind variable values used in executed SQL queries in the database. This allows you to confirm the actual contents of the query execution.

ⓘ These are not real-time bind values, but values captured in the database (`v$sql_bind_capture`). If there are many, up to 100 entries are displayed.

Trend Tab

The upper chart visualizes Elapsed Time, I/O, and Wait Time trends by time range.

The lower statistics table shows accumulated execution statistics, Wait Class ratio, and major Wait Events at a glance.

Elapsed Time

Displays the sum of **CPU time**, **Wait Class time**, and **Idle time** during the selected period with the time selector.

Example:

- CPU Time: Time actually spent using CPU for SQL execution
- Scheduler: Time the session waited for CPU
- Idle: Time not executed, waiting, or idle (intuitively confirms which factor consumed the most time)

Elapsed Time Trend

Aggregates **average Elapsed Time** and execution count at 5-minute intervals and shows the trend.

If the period is less than 3 hours, it aggregates by 1-minute intervals.

This helps quickly identify whether Elapsed Time spiked at a certain time or response time increased compared to execution counts, making it useful to pinpoint performance degradation timing.

I/O Trend

Shows **Logical Reads** and **Physical Reads** during query execution in 5-minute intervals.

If the period is less than 3 hours, it displays by 1-minute intervals.

- Logical Reads: Number of reads from buffer cache
- Physical Reads: Number of reads directly from disk. Useful for comparing cache efficiency and I/O load to identify bottlenecks

Wait Time Trend

Shows **Wait Event elapsed time** during query execution by time range.

If Wait Time spikes at a certain moment, it indicates DB resource contention or I/O latency. You can track which events (locks, I/O, parallel processing, etc.) caused the bottleneck.

Total Statistics

Displays **accumulated SQL execution statistics** for the selected range.

Example:

- sorts(rows): Total number of sorted rows
 - session logical reads: Session logical read count
 - table scan blocks gotten: Number of blocks read during table scans
 - elapsed_time: Total elapsed time
- Confirms aggregated data of how SQL execution used resources.

Total Wait Classes

Shows the sum of **Wait Time** that occurred during SQL execution, grouped by Wait Class.

The more time consumed by a specific Wait Class, the more likely that area is the bottleneck.

Example: Idle, Scheduler, User I/O, Concurrency

Top Wait Events

Displays the **top detailed Wait Events** during the selected range.

Provides a more fine-grained view than Wait Classes to see which events most impacted performance.

Example: PX Deq: Execution Msg, resmgr:cpu quantum, direct path read, etc.

Cube

The **Cube** menu shows the load distribution over a long period in a single hitmap and lets you pick the interval you are interested in and drill down into detailed metrics. It is provided only in the Oracle Pro product.

To track a performance issue, you must first find when it occurred. If you widen the query range of a metric chart to a full day, short spikes are flattened out and go unnoticed. If you narrow the range, you cannot tell which interval to look at. Cube divides the entire query period into 5-minute or 1-day cells and displays them by color intensity. You can locate the interval where load is concentrated by color first, then analyze only that interval in detail.

On this screen, you can do the following:

- Select a metric and check its distribution over the entire query period on a hitmap
- Select a cell with high load on the hitmap and check the OS, session, stat, and wait metrics for that interval
- Check the Top SQL for the same interval

Viewing the hitmap

Specify the query conditions for the hitmap in the left area of the screen.

1. In **Start date**, select the date to start the query from.
2. In **Query period**, select the unit of the hitmap. **Day (5 min)** displays a day in 5-minute cells, and **Month (daily)** displays a month in 1-day cells.
3. In **Aggregation**, select the method to apply when a single cell contains multiple values. You can select **Average** or **Max**.
4. In **Metrics**, select the metric to display on the hitmap.
5. In the **Instance** selector at the top right of the screen, select the instance to query.

Metrics

You can select any metric in the `db_oracle_dma_counter` category. The most commonly used metrics are as follows.

Item	Metric	Description
CPU	<code>cpu(xos)</code>	Server CPU usage collected by the XOS agent
Memory	<code>mem(xos)</code>	Server memory usage collected by the XOS agent
Active Sessions	<code>active_sessions</code>	Number of active sessions
DB Time	<code>DB time</code>	Time the database spent processing requests
Logical I/O	<code>session logical reads</code>	Number of session logical reads
Physical I/O	<code>physical reads</code>	Number of direct reads from disk
SQL Executions	<code>execute count</code>	Number of SQL executions
Lock Wait	<code>lock_wait_sessions</code>	Number of sessions waiting for a lock

Analyzing a detailed interval

When you select a cell on the hitmap, the detailed metrics for that interval are displayed on the right side of the screen. If you query by **Day (5 min)**, the 5-minute interval of the selected cell becomes the analysis target.

The tabs in the detail area are as follows.

Item	Description
OS	Trends for CPU, memory, disk I/O, network I/O, and CPU I/O wait collected by the XOS agent
Session	Trends for active sessions, connections, DB Time, lock wait sessions, long-running sessions, and enqueue waits
Stat	Trends for SQL executions, transactions (commit and rollback), logical I/O, physical I/O, block changes, and redo
Wait	Wait event distribution and latch and enqueue metrics for the selected interval

Item	Description
Top-N	Top SQL list for the selected interval

The **Wait** and **Top-N** tabs use the same widgets as the **Wait analysis** and **Top SQL** screens. For more information about the two screens, see [Wait analysis](#) and [Top SQL](#).

The **Top-N** tab includes the option bar and the analysis criteria tabs of the **Top SQL** screen as they are. The interval selected in Cube is applied as the query time.

In the **Session** tab, you can check the active session list and the process information for the selected interval. Because this screen queries a past interval, the session list is not refreshed in real time. The active session and process information are each available on their own subtab.

Lock Tree

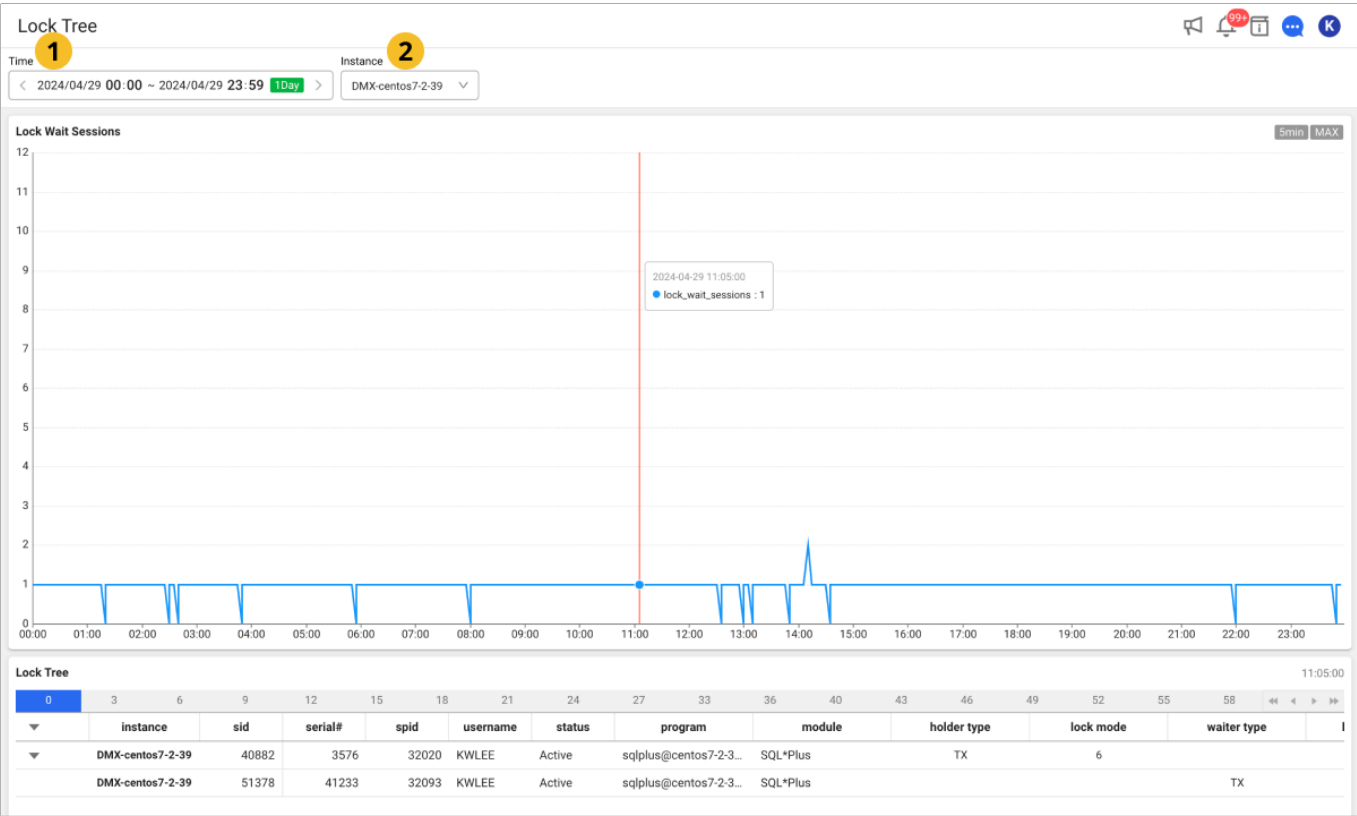
Home > Select Project > **Analysis** > **Lock Tree**

This tool analyzes the trend of locks that occurred during the day. This function allows you to visually check the tree structure of the session that generated the lock (lock holder) and the session waiting for the lock (lock waiter). You can analyze the relationship between the holder and waiter of the lock that occurred at a specific time point. Through the WhaTap's **Lock Tree** feature, the DB administrator can identify and resolve the DB lock issues and optimize the DB performance.

Key features are as follows:

- **Lock trend analysis:** The trends of locks that occurred during the set search period are traced over time and displayed in a graph. This allows you to visually check how the locks are created and resolved.
- **Check lock holder and waiter:** You can see the holder and waiter for each lock in a tree structure. Through this, when a specific lock occurs, you can check the relationship between the session that generated the lock and the waiting session.
- **Inter-session relationship analysis:** By analyzing the relationship between the session that generated the lock and the waiting session, you can identify the cause of the lock that occurred during the query execution and transaction handling.
- **Visual lock trend by time:** Through the **1-Min Lock Trends** and **Lock Wait Sessions** features, the lock occurrence data can be analyzed in time zones.

Basic usage guide



Set the desired 1 date and time, and then select an instance from the 2 list. If a lock occurs at the set time, lock wait sessions and the retrieved data appear in the table at the bottom of the screen.

Lock Wait Sessions

The Lock Wait Sessions feature allows you to analyze the lock wait sessions. Depending on the query period, the lock wait session data is provided at 5-minute intervals when the query is for 3 hours or more, and at 5-second intervals when the query is for less than 3 hours. You can also adjust the time range by dragging the chart flexibly.

1-Min Lock Trends

If you click a specific time zone in [Lock Wait Sessions](#), the lock occurrence status for 30 minutes, from 15 minutes before to 15 minutes after is visualized. This allows you to effectively analyze the lock occurrence trend at 30-minute intervals.

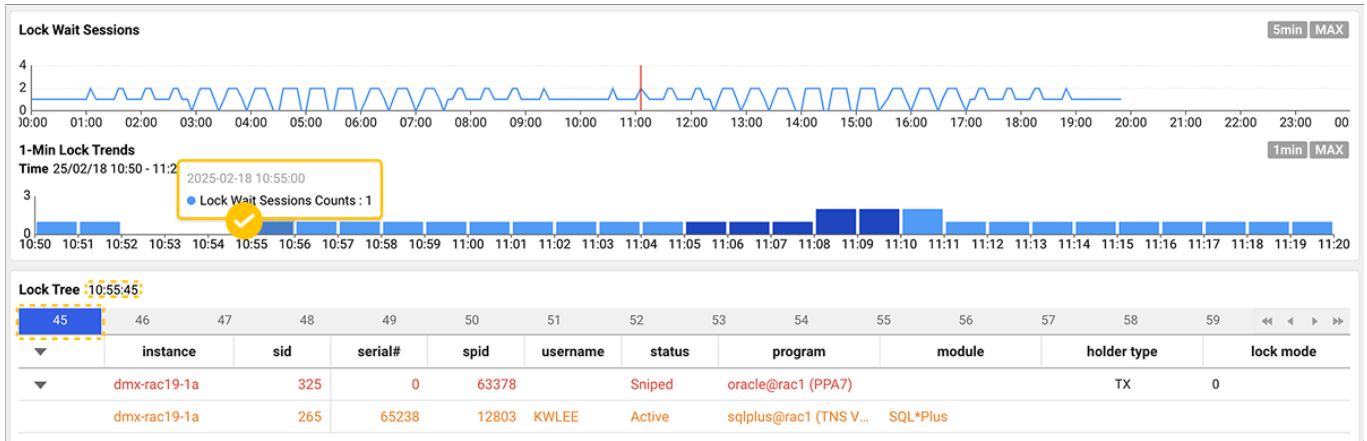
! **Total Locks** indicates the total number of locks generated in the time zone.

Viewing time

[1-Min Lock Trends](#) and [Lock Tree](#) tables are timed based on the maximum number of locks that occurred at the clicked time in [Lock Wait Sessions](#). This allows you to find the zone with the most locks.



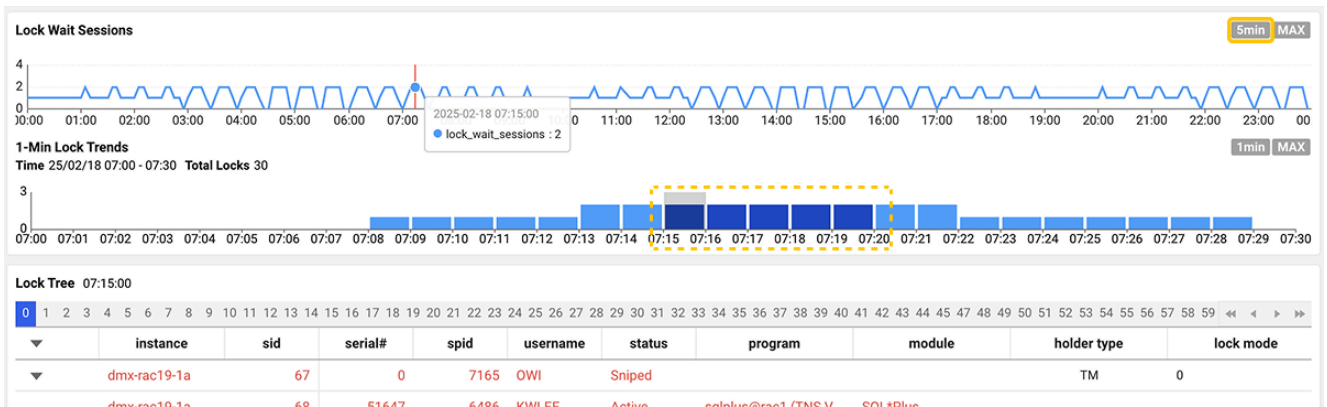
If you click the bar of a different time zone in the [1-Min Lock Trends](#) chart with the [Lock Wait Sessions](#) unchanged, the time range is synced to the [Lock Tree](#) table by using the seconds with the highest number of locks for the minute. This allows you to see the information reflected in the [Lock Tree](#) table when changing the time zone.



- **Viewing for 3 hours or more (using 5-minute data)**

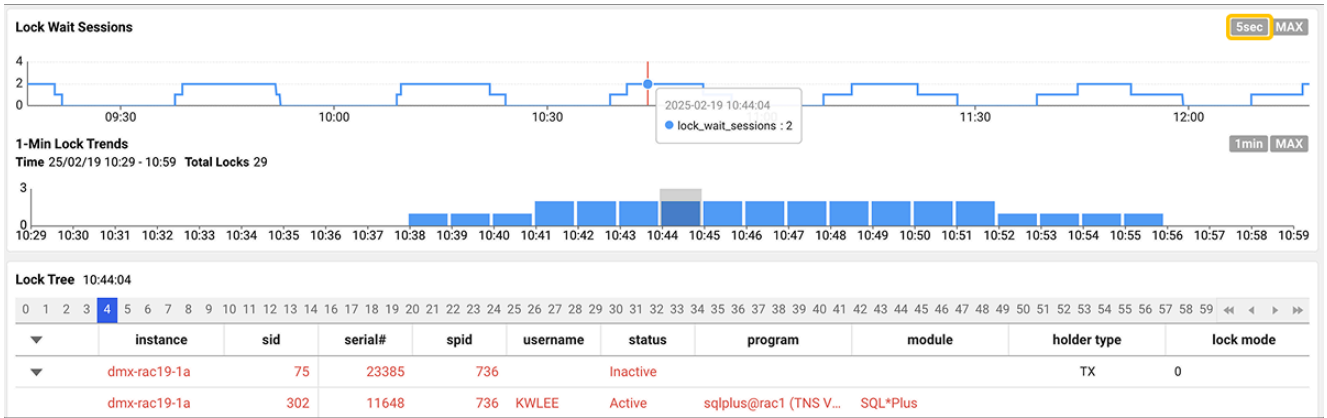
The bar color of the 5-minute zone from the time of inquiry appears more prominent.

When the 07:15:00 time point is clicked, the color of the 07:15:00-07:20:00 zone's bar chart appears more prominent.



- **Viewing for less than 3 hours (using 5-minute data)**

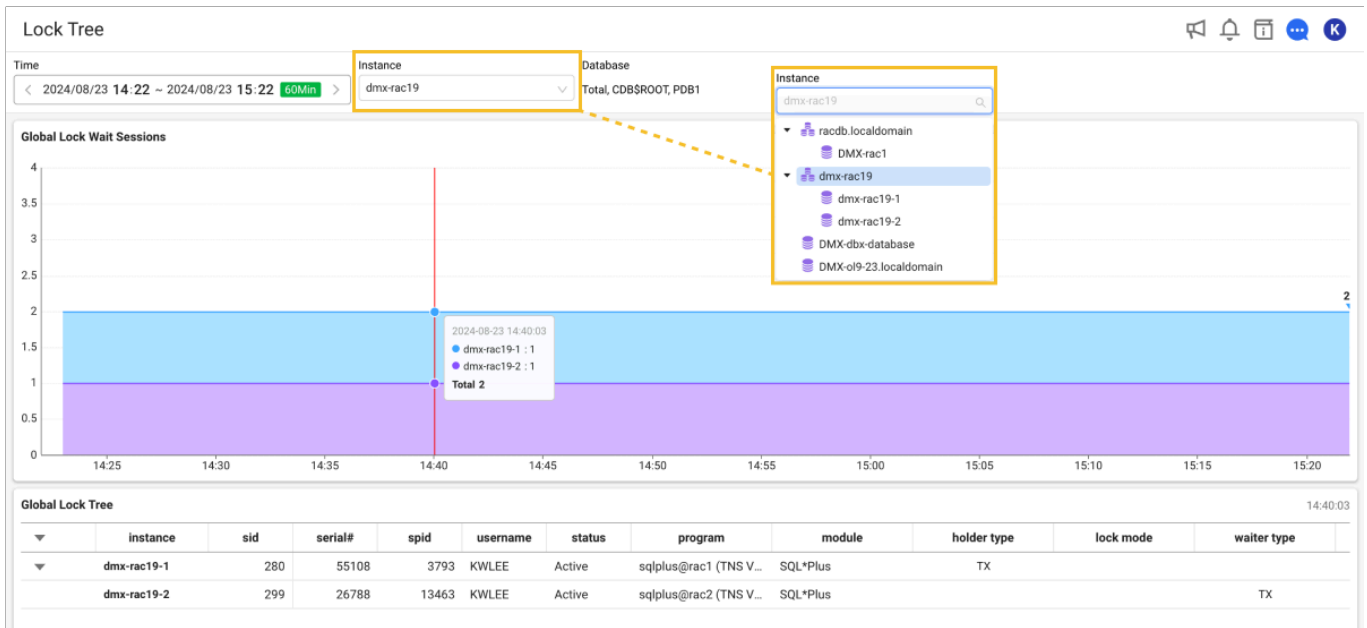
It is useful for analyzing the locks that occur within short time intervals.



Checking the global lock information

Checking the global lock information in the clustered Oracle database environment provides several important advantages. Especially in the Oracle Real Application Clusters (RAC) environment, this information is essential for performance optimization and troubleshooting.

If the environment consists of clusters, you can select a cluster in **Instance**.



- [Global Lock Wait Sessions](#)

It visualizes the number of global lock waiting sessions that occurred within a selected time range, by hour. You can see at a glance how many lock waits occurred at a specific time. Each color represents a different instance, and locks are displayed separately by the colored instance.

- [Global Lock Tree](#)

You can check the details related to the lock in a tree structure. It contains important information such as the lock-occurred instance, session ID (**sid**), and session status (**status**). You can quickly identify the cause and impact of a lock through the lock's holder type (**holder type**), lock mode (**lock mode**), and lock waiting type (**waiter type**).

The tree structure allows us to see which sessions hold locks and which are waiting.

- The maximum time range that can be queried with a cluster selected is 3 hours.
- When you select an item from the list of the [Global Lock Tree](#) section, you can see **Session detail**.

- In the RAC environment, multiple instances can access the same data simultaneously. Global Lock information is one of key mechanisms to ensure data consistency in concurrent access situations. By monitoring this, you can check that the data is consistent.
- Global Lock waiting time can have a significant impact on system performance. Monitoring global lock contention across multiple instances in the cluster can help you detect bottlenecks that cause performance degradation as early as possible.
- Global lock information provides important insights into how transactions are processed within the cluster. If concurrency control is not properly implemented, transaction latency may increase or deadlocks may occur.
- In the RAC environment, load balancing between instances is very important. Global lock information can help you understand which instances are encountering more contention, allowing for load redistribution or improvements in resource management based on this.

Column information guide

For more information about columns, see [the link](#).

Item	Description
sid	Session ID
cpu(xos)	CPU utilization collected by the installed XOS
serial#	Value to increment to determine if the same session ID has been used.
username	DB user name.
command	Same as <code>COMMAND_TYPE</code> ; SQL command number
status	Session status
schemaname	Schema. Same as the username.
osuser	Username of the OS that accessed the session.
spid	OS process ID.
process	ID of the client process that accessed the session.
logon_time	Session created time point.
last_call_et	If the session is active, it is the elapsed time (sec) since activation.
sql_text	<code>query</code> , running statement, SQL statement that separates literal strings
sql_param	Literal value separated from <code>query</code>
prev_sql_text	Previously executed query
prev_sql_param	Previously executed query param
program	Program information executed by client
type	session type - can be adjusted in user background recursive (whatap.conf).

Item	Description
seq#	Number that uniquely identifies the current or last wait (incremented with each wait)
event	Number of the resource or event if the session is waiting.
p1	First waiting event parameter (decimal point)
p2	Second waiting event parameter (decimal point)
p3	Third waiting event parameter (decimal point)
wait_class	Class name of the waiting event
wait_time	The value is 0 if the session is waiting. It is the last waiting time (milliseconds) if the value is greater than 0.
seconds_in_wait	Time spent waiting for the wait if the session is waiting. Otherwise, it is the time elapsed since the last waiting.
state	Waiting state: WAITING , WAITED UNKNOWN TIME , WAITED SHORT TIME , WAITED KNOWN TIME
machine	Name of the client server that accessed the session (hostname)
port	Client port number
terminal	Client terminal name
module	Name of the module set by the user of the program executed on the client.
action	Action name set by the user of the running module.
client_info	Client information set by user
client_identifier	Client ID set by user

Item	Description
session logical reads	Number of times data was read into the buffer cache
physical reads	Number of data reads from the physical disk
execute count	SQL execution count
parse count (hard)	Total number of syntax analysis calls (actual syntax analysis)
parse count (total)	Total number of syntax analysis calls (hard, soft, describe)
opened cursors current	Number of open cursors
db block changes	Change count of all blocks in the database
session pga memory	Session's current PGA size
undo_segid	Undo information ID
undo_blk	Undo block count
undo_rec	Undo record count
sql_address	SQL statement identification data
sql_hash_value	SQL statement identification data
sql_id	SQL ID of the running SQL statement
sql_child_number	Sub number of the running SQL statement
sql_exec_start	Start time point of SQL running in the session
sql_exec_id	SQL running ID
prev_sql_addr	Information that identifies the last run SQL statement

Item	Description
prev_child_number	Sub number of the last SQL statement executed
prev_exec_start	Start time point of the last SQL statement executed
prev_exec_id	Start ID of the last SQL statement executed
plsqli_entry_object_id	Object ID of the top PL/SQL subprogram on the stack.
plsqli_entry_subprogram_id	ID of the top PL/SQL subprogram on the stack.
plsqli_object_id	Subprogram ID of the running PL/SQL object
taddr	Transaction address
lockwait	lock wait address (lock address where the session is waiting)
row_wait_obj	Object ID of the table containing the row specified in ROW_WAIT_ROW#
row_wait_file	ID of the data file containing the row specified in ROW_WAIT_ROW#
row_wait_block	ID of the block containing the row specified in ROW_WAIT_ROW#
row_wait_row	Currently locked row
pdml_status	If the value is ENABLED , the session is in PARALLEL DML activation mode. If the value is DISABLED , the session does not support the PARALLEL DML activation mode. If the value is DISABLED , the session does not support the PARALLEL DML activation mode.
pq_status	If the value is ENABLED , the session is in PARALLEL QUERY activation mode. If the value is DISABLE , the session does not support the PARALLEL QUERY activation mode. If the value is FORCED , it indicates the session was changed to force the use of PARALLEL QUERY .
blocking_session_status	It provides details about whether there is a blocking session.

Item	Description
<code>blocking_instance</code>	Instance ID for the blocking session
<code>blocking_session</code>	Session ID for the blocking session
<code>final_blocking_session_status</code>	It provides details about whether there is a last block session.
<code>final_blocking_instance</code>	Instance ID for the last blocking session
<code>final_blocking_session</code>	Session ID for the last blocking session
<code>service_name</code>	Service name of the session. It is similar to the DB name.
<code>saddr</code>	Session address
<code>con_id</code>	ID of the container associated with the data (only for multi-database support)
<code>rss(xos)</code>	OS metrics collected by the XOS agent
<code>pss(xos)</code>	OS metrics collected by the XOS agent
<code>ioread(xos)</code>	OS metrics collected by the XOS agent
<code>iowrite(xos)</code>	OS metrics collected by the XOS agent

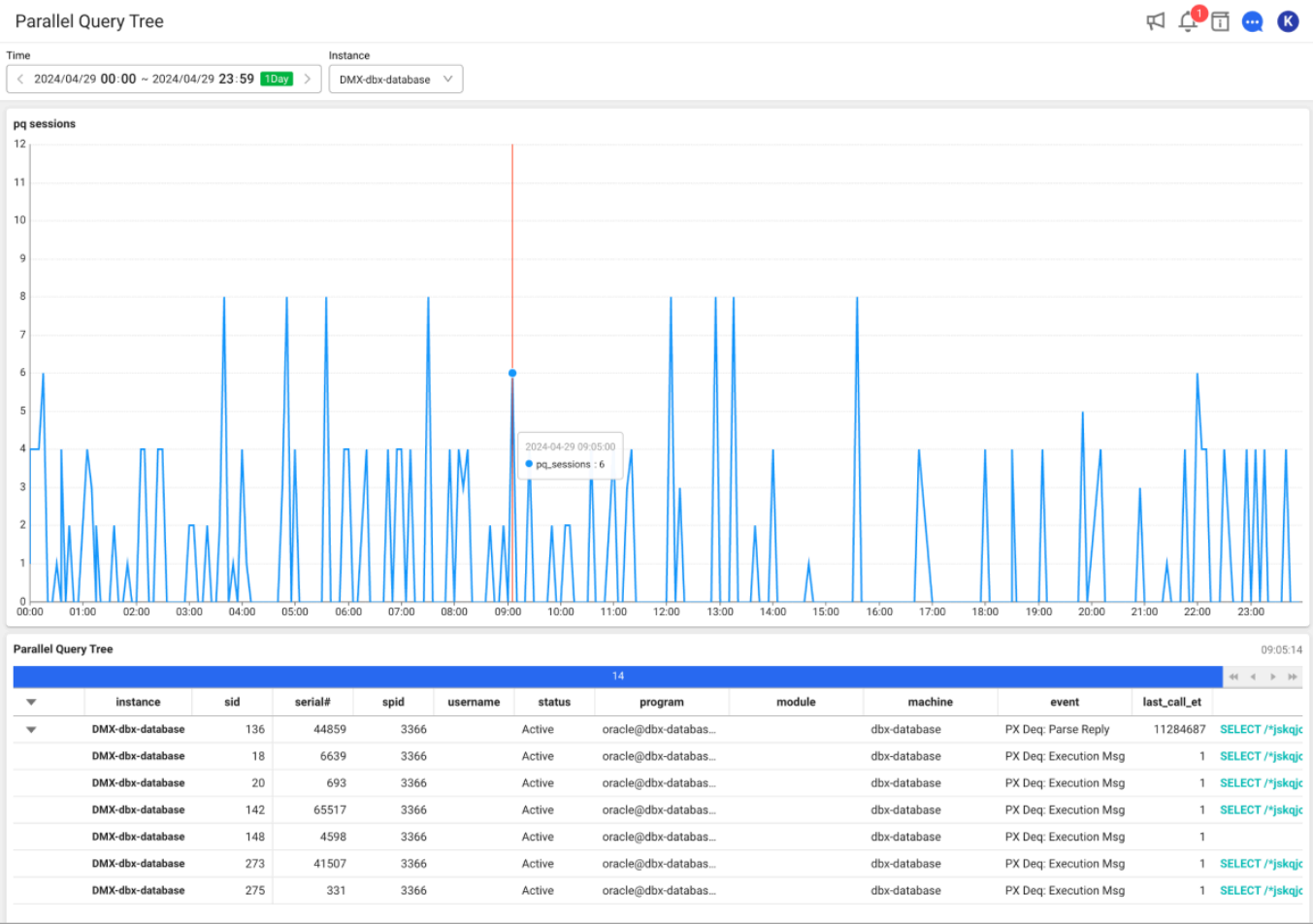
ⓘ WhaTap basically stores the client-related information.

Parallel Query Tree

Home > Select Project > Analysis > Parallel Query Tree

Parallel Query Tree provides the function to analyze the trend of parallel queries per day. The slave processes that perform actual tasks and the coordinator processes that assign tasks to them are distinguished, and overall executions are displayed in an easy-to-understand manner. It can help determine what purpose it was used for by providing information about parallel queries performed at a specific time point.

Basic screen guide



Set the time to view in the **Time** item, and then select the instance to view parallel query information in the **Instance** item.

You can check the trend of parallel queries that meet the set conditions through a graph chart, and the structure of coordinators and slaves is represented in the form of tree.

- When you select the [query](#) column in the table list of the [Parallel Query Tree](#) section, the **SQL details** window appears.

Column information guide

For more information about columns, see [the link](#).

Item	Description
<code>sid</code>	Session ID
<code>cpu(xos)</code>	CPU utilization collected by the installed XOS
<code>serial#</code>	Value to increment to determine if the same session ID has been used.
<code>username</code>	DB user name.
<code>command</code>	Same as <code>COMMAND_TYPE</code> ; SQL command number
<code>status</code>	Session status
<code>schemaname</code>	Schema. Same as the username.
<code>osuser</code>	Username of the OS that accessed the session.
<code>spid</code>	OS process ID.
<code>process</code>	ID of the client process that accessed the session.

Item	Description
logon_time	Session created time point.
last_call_et	If the session is active, it is the elapsed time (sec) since activation.
sql_text	query, running statement, SQL statement that separates literal strings
sql_param	Literal value separated from query
prev_sql_text	Previously executed query
prev_sql_param	Previously executed query param
program	Program information executed by client
type	session type - can be adjusted in user background recursive (whatap.conf).
seq#	Number that uniquely identifies the current or last wait (incremented with each wait)
event	Number of the resource or event if the session is waiting.
p1	First waiting event parameter (decimal point)
p2	Second waiting event parameter (decimal point)
p3	Third waiting event parameter (decimal point)
wait_class	Class name of the waiting event
wait_time	The value is 0 if the session is waiting. It is the last waiting time (milliseconds) if the value is greater than 0.
seconds_in_wait	Time spent waiting for the wait if the session is waiting. Otherwise, it is the time elapsed since the last waiting.
state	Waiting state: WAITING, WAITED UNKNOWN TIME, WAITED SHORT TIME,

Item	Description
	WAITED KNOWN TIME
machine	Name of the client server that accessed the session (hostname)
port	Client port number
terminal	Client terminal name
module	Name of the module set by the user of the program executed on the client.
action	Action name set by the user of the running module.
client_info	Client information set by user
client_identifier	Client ID set by user
session logical reads	Number of times data was read into the buffer cache
physical reads	Number of data reads from the physical disk
execute count	SQL execution count
parse count (hard)	Total number of syntax analysis calls (actual syntax analysis)
parse count (total)	Total number of syntax analysis calls (hard, soft, describe)
opened cursors current	Number of open cursors
db block changes	Change count of all blocks in the database
session pga memory	Session's current PGA size
undo_segid	Undo information ID
undo_blk	Undo block count

Item	Description
<code>undo_rec</code>	Undo record count
<code>sql_address</code>	SQL statement identification data
<code>sql_hash_value</code>	SQL statement identification data
<code>sql_id</code>	SQL ID of the running SQL statement
<code>sql_child_number</code>	Sub number of the running SQL statement
<code>sql_exec_start</code>	Start time point of SQL running in the session
<code>sql_exec_id</code>	SQL running ID
<code>prev_sql_addr</code>	Information that identifies the last run SQL statement
<code>prev_child_number</code>	Sub number of the last SQL statement executed
<code>prev_exec_start</code>	Start time point of the last SQL statement executed
<code>prev_exec_id</code>	Start ID of the last SQL statement executed
<code>plsql_entry_object_id</code>	Object ID of the top PL/SQL subprogram on the stack.
<code>plsql_entry_subprogram_id</code>	ID of the top PL/SQL subprogram on the stack.
<code>plsql_object_id</code>	Subprogram ID of the running PL/SQL object
<code>taddr</code>	Transaction address
<code>lockwait</code>	lock wait address (lock address where the session is waiting)
<code>row_wait_obj</code>	Object ID of the table containing the row specified in <code>ROW_WAIT_ROW#</code>
<code>row_wait_file</code>	ID of the data file containing the row specified in <code>ROW_WAIT_ROW#</code>

Item	Description
row_wait_block	ID of the block containing the row specified in ROW_WAIT_ROW#
row_wait_row	Currently locked row
pdml_status	If the value is <code>ENABLED</code> , the session is in <code>PARALLEL DML</code> activation mode. If the value is <code>DISABLED</code> , the session does not support the <code>PARALLEL DML</code> activation mode. If the value is <code>DISABLED</code> , the session does not support the <code>PARALLEL DML</code> activation mode.
pq_status	If the value is <code>ENABLED</code> , the session is in <code>PARALLEL QUERY</code> activation mode. If the value is <code>DISABLE</code> , the session does not support the <code>PARALLEL QUERY</code> activation mode. If the value is <code>FORCED</code> , it indicates the session was changed to force the use of <code>PARALLEL QUERY</code> .
blocking_session_status	It provides details about whether there is a blocking session.
blocking_instance	Instance ID for the blocking session
blocking_session	Session ID for the blocking session
final_blocking_session_status	It provides details about whether there is a last block session.
final_blocking_instance	Instance ID for the last blocking session
final_blocking_session	Session ID for the last blocking session
service_name	Service name of the session. It is similar to the DB name.
saddr	Session address
con_id	ID of the container associated with the data (only for multi-database support)
rss(xos)	OS metrics collected by the XOS agent
pss(xos)	OS metrics collected by the XOS agent

Item	Description
<code>ioread(xos)</code>	OS metrics collected by the XOS agent
<code>iowrite(xos)</code>	OS metrics collected by the XOS agent

! WhaTap basically stores the client-related information.

Session History

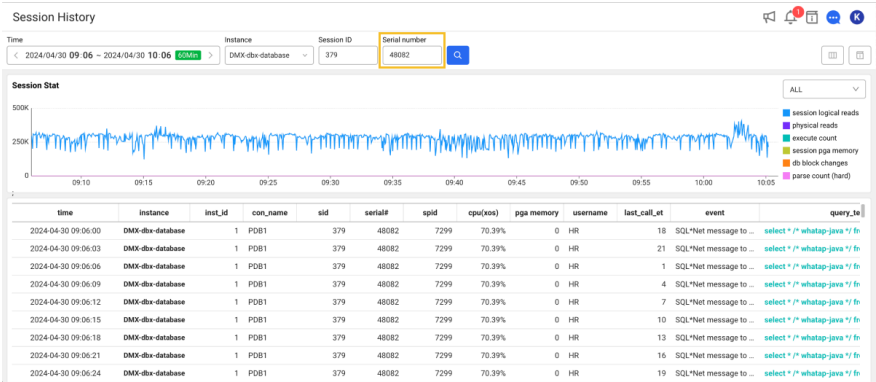
Home > Select Project > Analysis > Session History

In **Session History**, you can see what has been performed in a specific session during the query period. It provides information that allows you to check how much resource has been used for SQL for a long-running session and analyze the causes.

Basic usage guide

You must enter the session number (**Process ID**) to monitor as a required value, but it is more efficient to enter through the **Monitoring a Database Instance** menu.

- 1. Go to **Dashboard > Monitoring a Database Instance**.
- 2. In the **Active Session** tab, click a session from the ID column.
- 3. In **Session detail**, click **Session history** > to move to **Session History**.
- 4. You can view the session history collected for the selected session's process ID (PID).



Active session data collected every 5 seconds from the selected session is used to construct a chart and its list and provide information that can be used for analysis.

- **Wait:** You can see the percentage of waiting events through a chart.
- **Cpu:** You can see the change trend in CPU usage through the chart.
- **Memory:** You can see the change trend in memory usage through the chart.

- **Disk I/O:** You can see the change trend in disk usage through the chart.

If you select the **query** column on the utmost right of the list, the **SQL details** window appears where you can see the SQL statement. It contains full texts and plan information for the selected query.

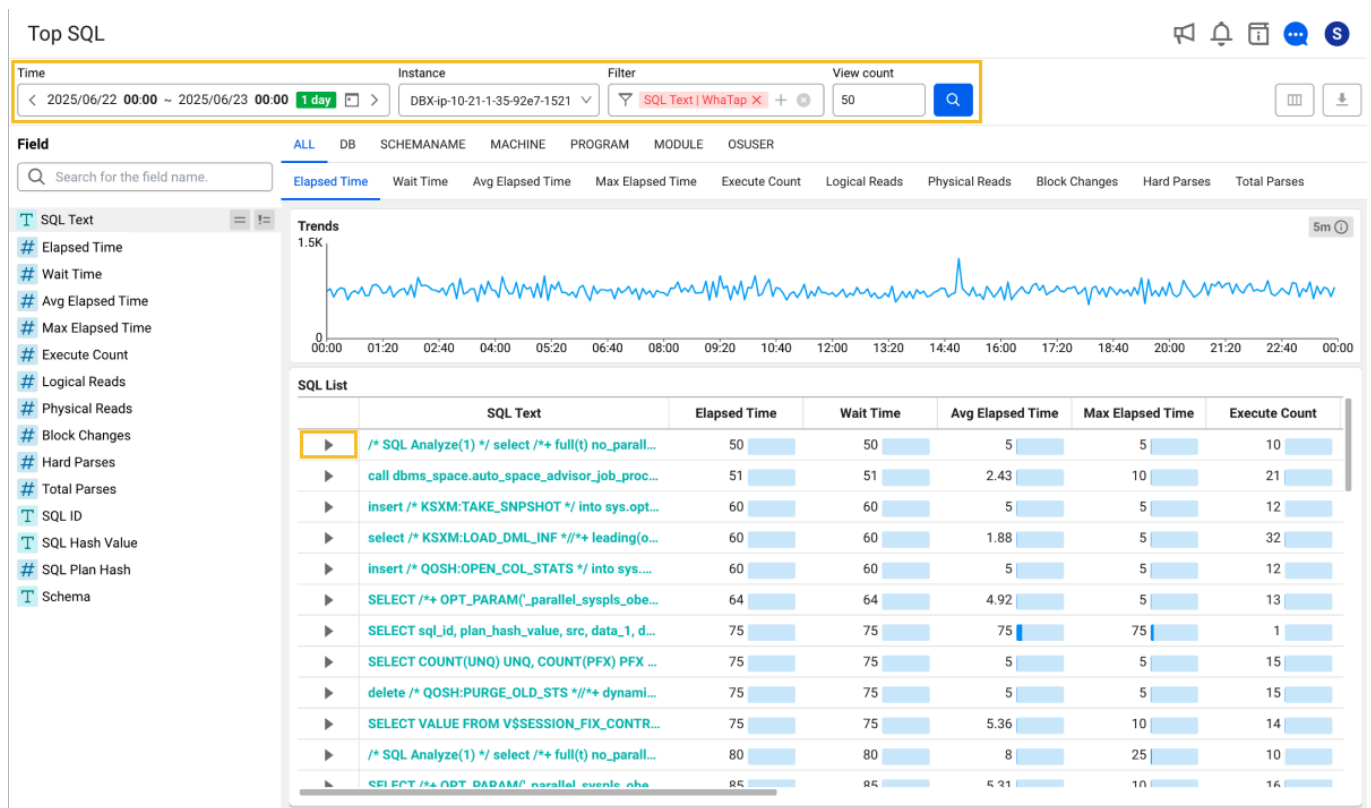
- If the [query_text](#) column is selected in the session table list, the **SQL details** window appears where you can see the SQL query statement and plan information.
- You can hide the table header columns or add any of them. You can also change the column order.

Top SQL

Home > Select Project > **Laboratory** > [Top SQL](#)

TOP SQL extends the existing SQL statistics functionality to allow you to query and analyze data over a long period of time. You can search data for 1 minute to 30 days intervals, and provides analysis metrics in tab format so you can see them at a glance. Additionally, you can add trend charts to easily identify SQL occurrence patterns over a specific time period.

Basic screen guide



1. In **1 Time**, select the time and date to view data. You can also select a lookup time by clicking the green button.
2. In **1 Instance**, select a target to view the data.

3. Select .

- If you click the text range for date and time, the option appears to select the date and time.
- If you select each column in the table header, you can sort the list by the selected item.
- Data is displayed in a table for 10 to 100 items. In **View count**, you can set the number of items to display in the table. After all settings are finished, select .
- To download the viewed data as a CSV file, select .

Checking the trends chart



The SQL statistics data collection period and data query range of the **3 Trends** chart may differ depending on the following criteria.

- **3 hours or less**
 - **If the SQL statistical data collection cycle option is set to 1 minute:** Data is collected at 1-minute intervals.
 - **If the SQL statistical data collection cycle option is not set to 1 minute:** Data is collected at 5-minute intervals.
- **1 day or less:** Data is collected at 5-minute intervals.
- **30 days or less:** Data is collected at 1-hour intervals.

Setting the collection cycle option for SQL statistical data (chart interval)

To display 1-minute summary data, it is required to check the agent version and its settings.



1. Update to Agent 2.4.5 or later.
2. Add the `sqlstat_send_interval = 1` setting in the `whatap.conf` file.

Checking the query-based execution trend

You can classify and search the desired data by usage type or change the sorting order. Select the desired tab from **2**.

On the utmost left for each item of the table, select **4** ►. A chart appears that displays the number of executions every 5 minutes and the average execution time of the query statement. If you hover your mouse over each bar graph, you can see the number of that time.

Sorting criteria	Description
elapsed avg (sec)	Sort by the longest average execution time
execute count	Sort by the greatest number of executions
elapsed sum (sec)	Sort by the longest total execution time
elapsed max (sec)	Sort by the longest maximum execution time
elapsed wait	Sort by the longest waiting time for queries
physical read	Sort by the greatest number of data blocks read from disk
session logical reads	Sort by the greatest number of data blocks read from memory

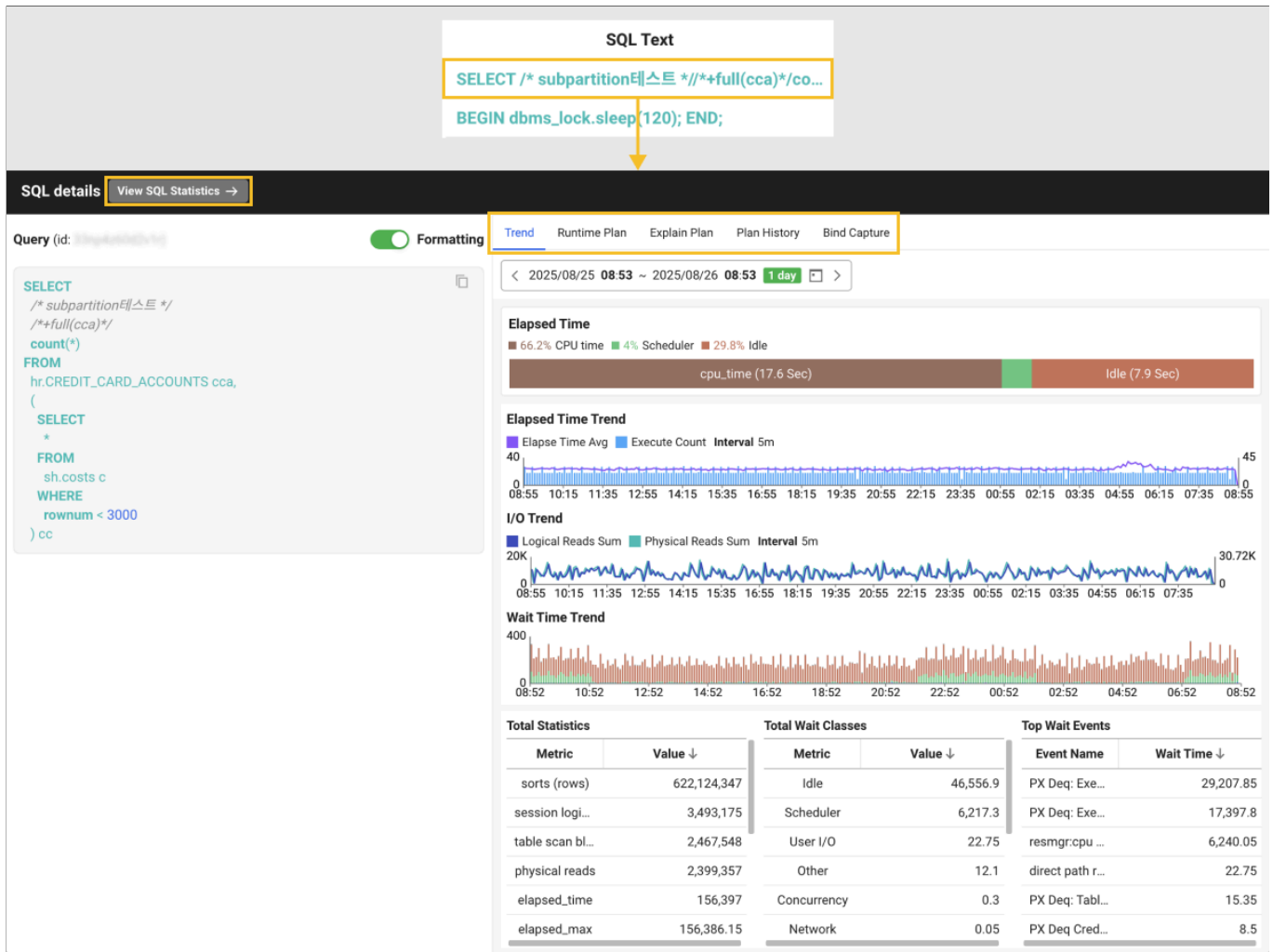
Previewing the query

sql	elapse avg (sec)	execute count ↓	elapse sum (sec)
select /* WhaTap6D#4 */ processlist_id,processlist_user,processlist_host,proces...			
<pre>select /* WhaTap6D\$4 */ processlist_id,processlist_user,processlist_host,processlist_db,processlist_command,processlist_state,thread_id,processlist_time,type,name,connection_type,thread_os_id,i.info from performance_schema.threads p left outer join information_schema.processlist i on p.processlist_id=i.id where type<>'S' and processlist_command<>'S'</pre>			

You can preview the query statement by moving the mouse pointer to the [sql](#) column in the search results list.

See query details

If you select [query](#) column from the searched result list, the **SQL details** window appears.



Click the **View SQL Statistics →** button to navigate to the **SQL statistics** menu, where you can view the statistical information related to the SQL query.

- **Trend:** Provides a comprehensive analysis of the execution time, I/O, and wait events of a specific SQL to quickly identify performance bottlenecks.
- **Runtime Plan:** Provides the execution plan and runtime information of the selected SQL query, including execution count, average execution time, and average physical reads.
- **Explain Plan:** Displays the execution plan predicted by the optimizer, showing cost, operations, object names, cardinality, and more.
- **Plan History:** Allows you to review the execution plan history of SQL queries executed in the database.
- **Bind Capture:** Displays the values of bind variables used in SQL queries executed in the database, allowing

you to check the actual query execution contents.

! These are not real-time bind values but those captured in the database (`v$sql_bind_capture`). Up to 100 entries are displayed if there are many.

Trend Tab

The top chart visualizes the trends of Elapsed Time, I/O, and Wait Time over time. The bottom statistics table provides a side-by-side comparison of cumulative execution statistics, Wait Class ratios, and major Wait Events.

Elapsed Time

Displays the total of **CPU time**, **Wait Class time**, and **Idle time** for the selected interval.

Example:

- CPU Time: Time actually spent using the CPU for SQL execution
- Scheduler: Time the session waited for CPU
- Idle: Time spent idle or waiting without execution (quickly see which factor consumed the most time)

Elapsed Time Trend

Shows the trend of **average Elapsed Time** and Execution Count aggregated in 5-minute intervals (1-minute intervals if under 3 hours).

This helps quickly identify performance degradation by checking if Elapsed Time spiked or if response time increased compared to execution count.

I/O Trend

Displays **Logical Reads** and **Physical Reads** during query execution in 5-minute intervals (1-minute intervals if under 3 hours).

- Logical Reads: Number of times data was read from the buffer cache
- Physical Reads: Number of times data was read directly from disk, allowing you to assess cache efficiency and I/O bottlenecks

Wait Time Trend

Displays the **Wait Event times** during query execution by time interval.

If Wait Time spikes at a certain point, it indicates DB resource contention or I/O delays. You can trace which events (lock, I/O, parallelism, etc.) caused the bottleneck.

Total Statistics

Shows **cumulative SQL execution statistics** for the selected interval.

Example:

- sorts(rows): Total number of sorted rows
- session logical reads: Number of session logical reads
- table scan blocks gotten: Number of blocks read during table scans
- elapsed_time: Total elapsed time
→ Provides aggregated data on how SQL execution used resources

Total Wait Classes

Displays the total **Wait Time** aggregated by Wait Class during SQL execution.

The higher the time for a particular Wait Class, the more likely it is the bottleneck.

Example: Idle, Scheduler, User I/O, Concurrency, etc.

Top Wait Events

Shows the **top detailed Wait Events** that occurred during the selected interval.

Provides more fine-grained detail than Wait Classes to identify which events most impacted performance.

Example: PX Deq: Execution Msg, resmgr:cpu quantum, direct path read, etc.

Checking the chart details

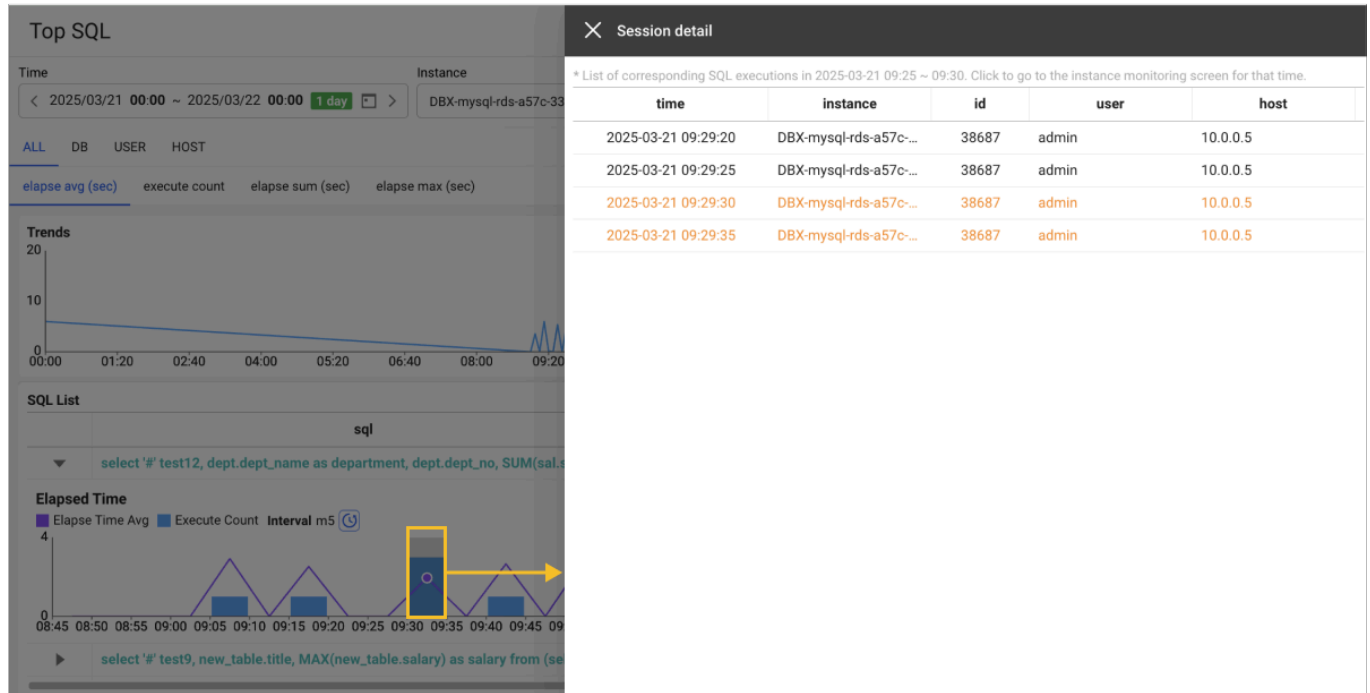
Check the detailed chart

Data is fetched at different intervals depending on the query time. The available time ranges are provided in

units of **1 minute (m1)**, **5 minutes (m5)**, and **1 hour (h1)**. To change back to the recent time, click the  icon. Data is automatically adjusted according to the set time.

Detailed charts have data cycles that differ depending on the query time, just like trend charts. You can adjust the query time range by dragging.

Checking the information details



When the bar graph is selected in the chart, the **Session detail** window appears where you can check statistical data for 5 minutes. The list of SQL executions appears for the selected time. When an item is selected, it goes to the **Instance monitoring** menu with a new window.

In **Instance monitoring**, you can check active sessions in the table and trends of various metrics in the selected time zone.

Filtering the searched results

Adding the filter conditions

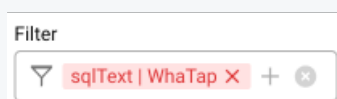
1. In the **Filter** option, click **+**.
2. In **Filter key**, select a desired filtering criteria.
 - If the value of the selected item is text, you can select any of **Includes** (blue) and **Excludes** (red).

example

- If the value of the selected item is number, you can select any condition of **==** (equal to), **>=** (greater than or equal to), and **<=** (less than or equal to).
3. In **Condition**, select a condition.
 4. Enter a string or number to match the condition.
 5. Select **Apply**.

! Adding and Deleting Filter Conditions

- To add a filtering condition, click the **Add** button and repeat steps 1~5. Added conditions are applied using the AND (&&) operator.
- To delete specific conditions while adding, click the  button on the right side of the filter condition. To delete all conditions, click the  **Delete All** button.
- To quickly remove all applied conditions from the **Filter** option, click the **×** button.
- The default setting is to exclude if `sqlText` contains 'WhaTap'.



Modifying the filter conditions

1. Click the applied item in the **Filter** option at the top of the screen.
2. In the **Edit filter** window, edit the desired item and click the **Apply** button.

Setting the table columns

You can hide the table header columns or add any of them. You can also change the column order. Select .

- After configuration, select **Confirm** to apply the settings in the table.
- In the  search bar, enter text to search the desired columns. Only the columns that meet the entered text are displayed.
- Images may differ depending on the product, project, or menu.

Adding columns

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

Deleting columns

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

Changing the column order

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

Initializing the configuration

To cancel all changes and reset them, select  **Initialized**.

SQL Plan Analysis

Home > Select Project > **Analysis** > **SQL Analysis**

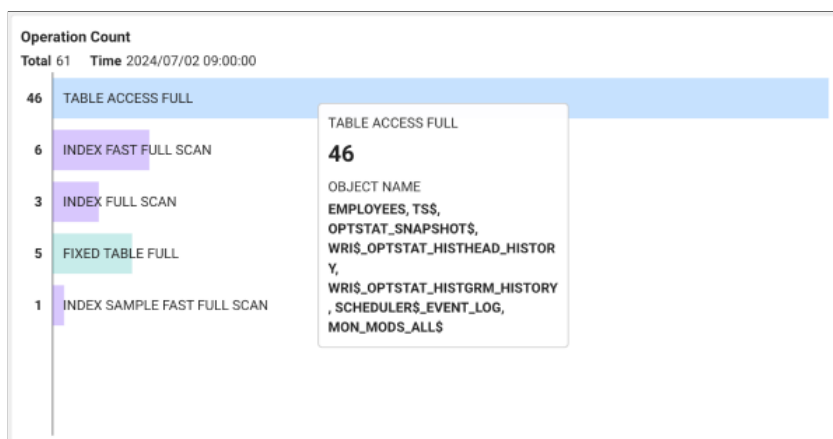
It provides useful data for diagnosing performance issues by analyzing SQL statements executed in the database.

SQL Plan Analysis consists of the [Access Statistics](#) tab and the [Plan Change History](#) tab.

Access Statistics

In the **Access Statistics** tab, you can check when each SQL performs a Full Scan, its type, and how many times it occurs. Queries that frequently trigger Full Scans can be optimized to use indexes, improving performance and reducing resource usage.

1. Go to the **Access Statistics** tab in **Analysis** > **SQL Plan Analysis**.
2. Select the query time and instance.
3. Set the **Time** and **Instance** options, then click the  button.
If needed, you can also configure the **Filter** option to filter the results.
4. Review the search results in the **Access Count** and **Operation Count** sections.
 - **Access Count:** A bar chart showing the number of Full Scans by time range.
Selecting a specific time displays the list of SQL statements executed during that time in the bottom list and updates the **Operation Count** chart on the right.
 - **Operation Count:** Displays the types and counts of Full Scans.
Hovering over each bar shows the object name.



- If the **query_text** column is selected from the table list, the **SQL details** window appears. For more information about the **SQL details** window.

Filtering the searched results

You can filter the retrieved results based on the following criteria:

- **Access Type:** Filter results by **Full Scan** or **Index Scan**.
- **Object Name:** Filter results by the object name.

Adding the access type filtering conditions

1. In the **Filter** option at the top of the screen, click the **+** button.
2. In **Edit filter**, select the **Access Type** item and the **Equal** condition, then choose either **FULL** or **INDEX**.
3. Click the **Apply** button.
4. To apply the selected conditions to the SQL list, click the  button.

- To search all for Full Scan and Index Scan, in **Edit filter**, select all options for **FULL** and **INDEX**. By selecting the **+** button, you add more filtering options.

Adding the object name filtering conditions

1. In the **Filter** option at the top of the screen, click the **+** button.
2. In **Add filter**, select the **Object Name** item and the **Equal** condition.
3. Click the empty input field and choose the desired object name from the displayed object list.
4. Click the **Apply** button.
5. To apply the selected conditions to the SQL list, click the  button.

! Adding and Deleting Filter Conditions

- To add a filtering condition, click the **Add** button and repeat steps 1~5. Added conditions are applied using the AND (&&) operator.
- To delete specific conditions while adding, click the  button on the right side of the filter condition. To delete all conditions, click the  **Delete All** button.
- To quickly remove all applied conditions from the **Filter** option, click the **×** button.

Modifying the filter conditions

1. Click the applied item in the **Filter** option at the top of the screen.
2. In the **Edit filter** window, edit the desired item and click the **Apply** button.

Plan Change History

In the **Plan Change History** tab, even if the SQL ID is the same, performance can be affected when the execution plan changes.

By detecting and monitoring plan changes made by the optimizer, you can prevent unnecessary changes and maintain consistent SQL performance.

1. Go to the **Plan Change History** tab in **Analysis > SQL Plan Analysis**.
2. Select the query time and instance.
3. Set the **Time** and **Instance** options, then click the  button.

4. In the **Plan Change Count** section, select a specific time period to view the list of plan changes that occurred during that time.
 - **Plan Change Count** section: A bar chart showing the number of plan changes that occurred by time period.

Checking the plan change details

1. In the **Plan Change History** tab of **SQL Plan Analysis**, click a specific change item from the list.
2. In the **Query** section, details before and after the plan change are displayed.
Compare the differences to identify the cause of performance changes.
 - In the **Query** section, click the  button in the upper-right corner to view the section in a new window.
3. To close the **Query** section, click the  button in the upper-right corner.

Filtering the searched results

You can filter the query results based on the following criteria.

- sql_id
- sql_hash_value
- after_plan_hash_value
- before_plan_hash_value

Adding the filter conditions

1. In the **Filter** option, click .
2. In **Filter key**, select a desired filtering criteria.
 - If the value of the selected item is text, you can select any of **Includes** (blue) and **Excludes** (red).

example

- If the value of the selected item is number, you can select any condition of  (equal to), .

(greater than or equal to), and $\leq$ (less than or equal to).

3. In **Condition**, select a condition.
4. Enter a string or number to match the condition.
5. Select **Apply**.

! Adding and Deleting Filter Conditions

- To add a filtering condition, click the **Add** button and repeat steps 1~5. Added conditions are applied using the AND ($\&\&$) operator.
- To delete specific conditions while adding, click the  button on the right side of the filter condition. To delete all conditions, click the  **Delete All** button.
- To quickly remove all applied conditions from the **Filter** option, click the $\times$ button.

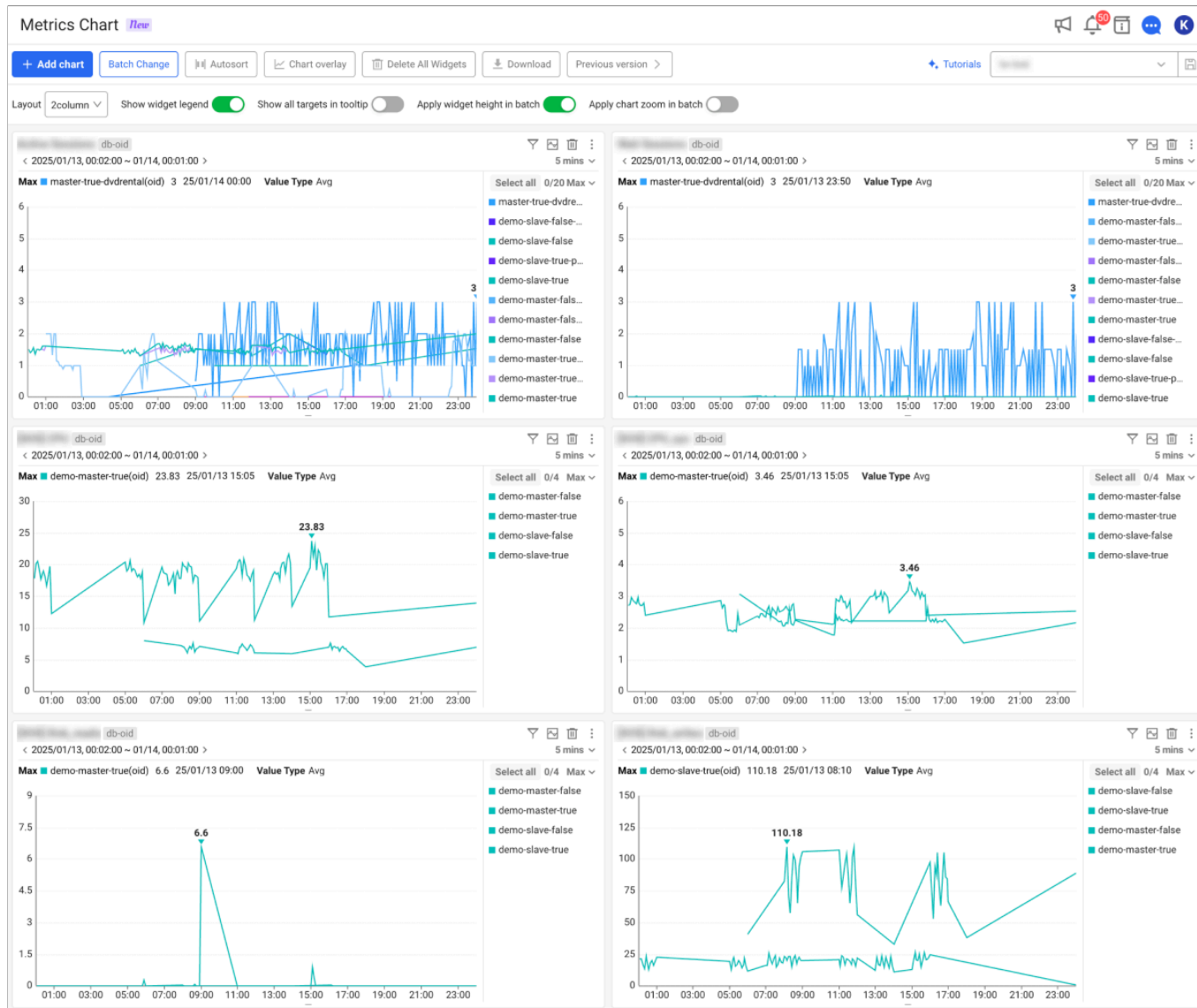
Modifying the filter conditions

1. Click the applied item in the **Filter** option at the top of the screen.
2. In the **Edit filter** window, edit the desired item and click the **Apply** button.

Metrics Chart

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

The **Metrics Chart** *New* menu is a core feature of the WhaTap monitoring solution, which provides the data collected from various monitoring targets in a visualized time-series chart. This allows users to effectively monitor the system performance, quickly identify anomalies, and gain insights for resolving issues.



- You can visually check various metrics via time series charts, and analyze multiple metrics upon failure at a time.
- You can select a specific time range to view past data and analyze the performance at that period.
- You can filter data based on specific time zones or conditions for more accurate analysis.
- Users can place their desired charts on the screen and freely adjust the layout.
- Users can arrange desired metrics and save them as presets for loading at any time.

! Difference from the **Metrics Chart** menu

The following are the changes since the [Service 2.8.0](#) update. To check the changes on the screen after the update, select **Tutorials** on the upper right.

- User experience (UX) has been improved so that users can set the time, target, and interval options, and then specify the metrics to monitor and add charts.
- You can change the time range of a chart view without deleting the widgets placed on the dashboard.
- It provides a batch **Batch Change** feature that allows you to apply options to all widgets placed on the dashboard at once.
- The preset feature allows you to save and load the widgets placed on the dashboard.
- You can adjust the layout of the dashboard and the functionality of the widgets through various options provided.

Adding a chart widget

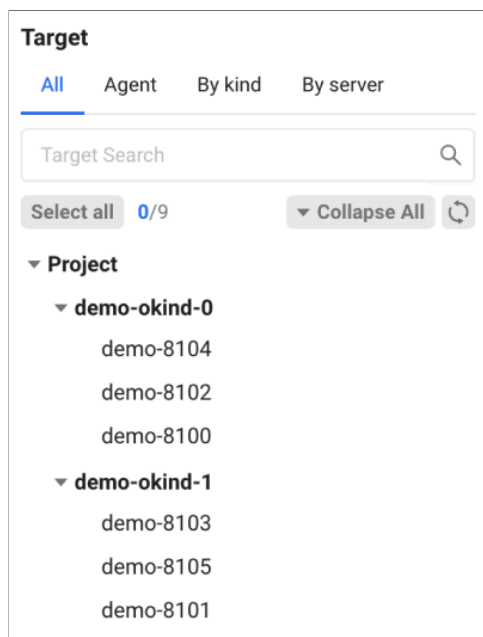
To add a metric as a widget to the dashboard, select **Add chart** on the upper left.

Setting the lookup time

In **Time**, set the date and time to view the metric.

Selecting a target

Select a desired target from the agents in the project. You can select an individual agent or a group classified by the agent configuration.

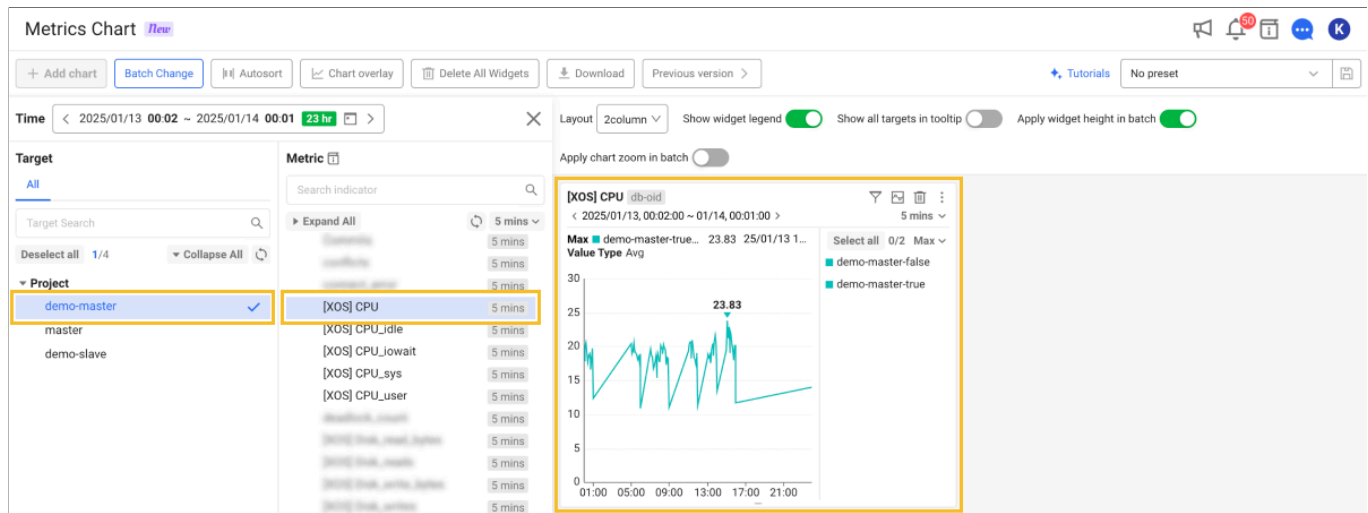


- **Agent:** You can select an individual agent or all agents.
- **By kind:** Agents can be monitored in each group classified by `whatap.okind` in the agent configuration.
- **By server:** Agents can be monitored in each group classified by `whatap.onode` in the agent configuration.

- Depending on the set time and date, available targets may change.
- You can select or deselect targets by clicking in the agent list. You can also select multiple agents at once by using the `Shift` key.
- By clicking **Select all** or **Deselect all**, you can select or deselect related sub-targets at once.
- If a desired agent does not appear in the list, select **Refresh**.
- By selecting **Collapse All** or **Expand All**, you can hide or expand the list by group.

Selecting a metric

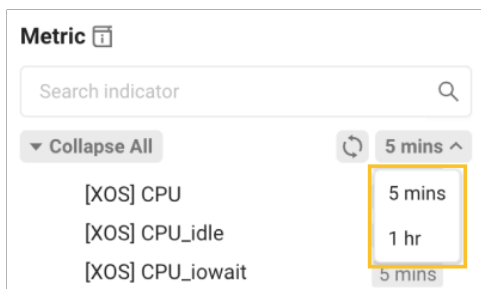
Select a target agent to monitor. Then the list that can be selected on the **Metric** panel appears. Select a desired metric in the list.



A chart widget for the selected metric is placed on the dashboard on the right. You can add chart widgets by selecting additional metrics. To enlarge the chart area, select **×** on the right of the **Time** option. The **Target** and **Metric** panels disappear, leaving only the chart area visible on the screen.

- If you enter a string in the **Search indicator** field, you can quickly search for a desired metric.
- You can also search by metric category and field name. For more information about the categories and fields, see [the following](#).
- By selecting **Collapse All** or **Expand All**, you can hide or expand the metrics list by category.
- If a desired metric does not appear in the list, select **Refresh**.

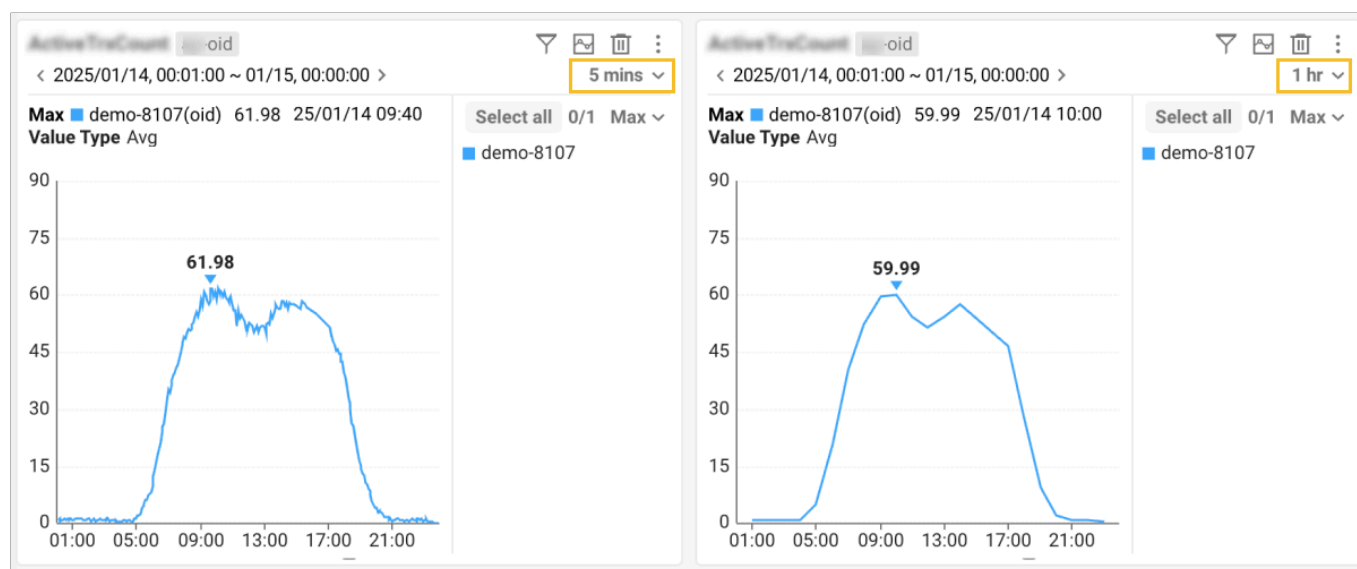
Setting the interval



You can set intervals for each chart, allowing you to adjust the aggregation unit of data to visualize at different time intervals. If you select ⌚ on the upper right of the **Metric** panel, you can select an interval option.

Each metric has a different default interval time, and the interval time changes depending on the query time range.

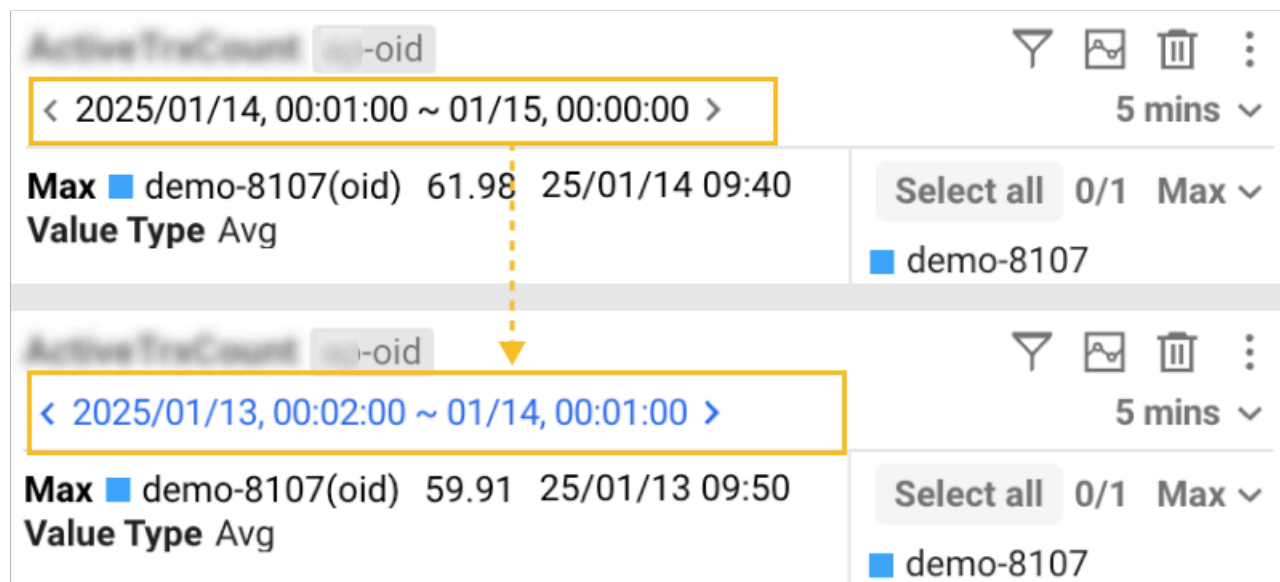
For example, the **5 min** interval displays average values for 5-minute intervals, and the **1 hour** interval displays average values for every one hour.



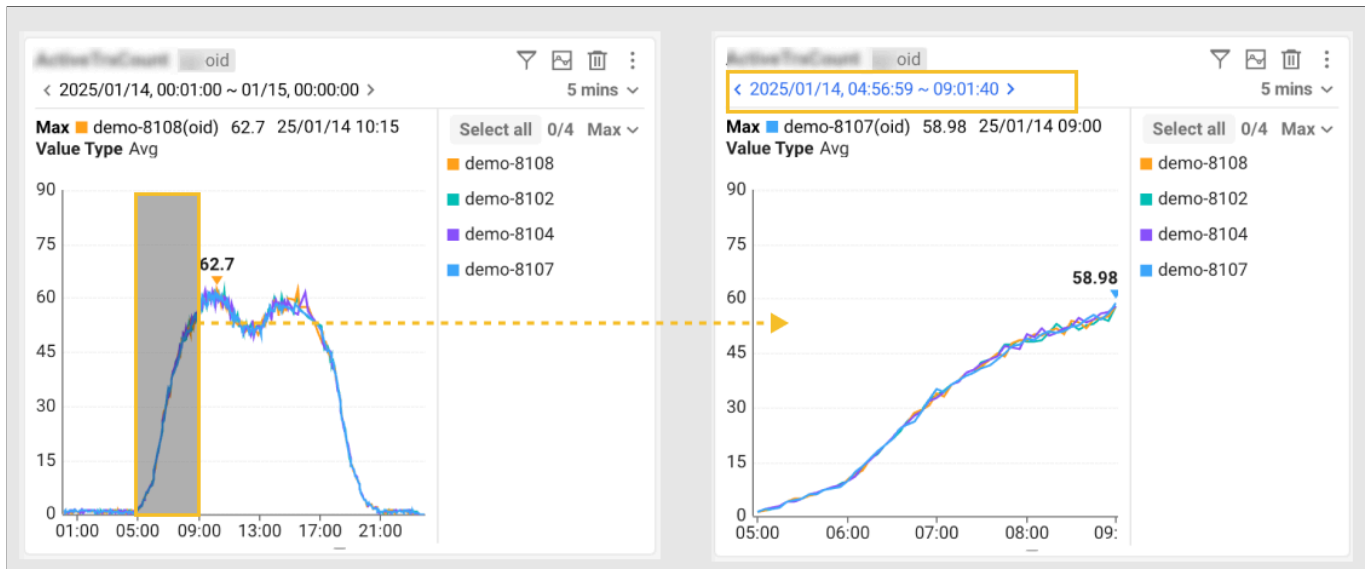
- ✓ By adjusting the interval option, you can analyze the data from different perspectives, from detailed data over a short time to overall trend over a long time.

Changing the query time

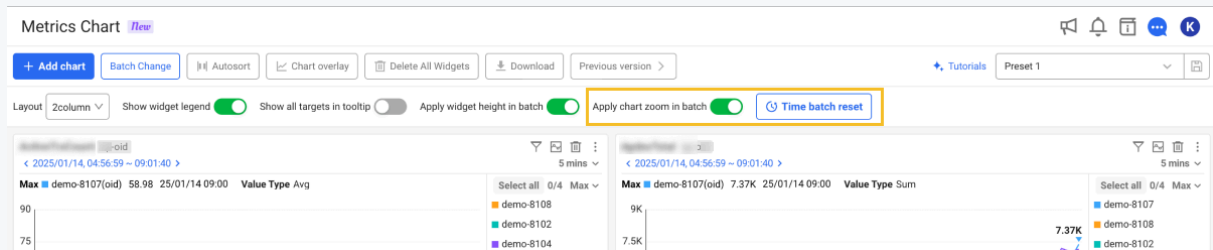
To change the chart query time range for an individual widget, select **<** or **>** at the top of the widget. The time change range is the same as the time reference set in the **Time** option. For example, if the **Time** option is set to **1 day** and you select **<** or **>**, you can change the time to a daily interval.



Alternatively, by dragging a specific area of the chart, you can change the time according to the dragged range for a view.



- ✓ You can change the time for all widgets by changing for each widget with the **Apply chart zoom in batch** option enabled at the top of the screen.



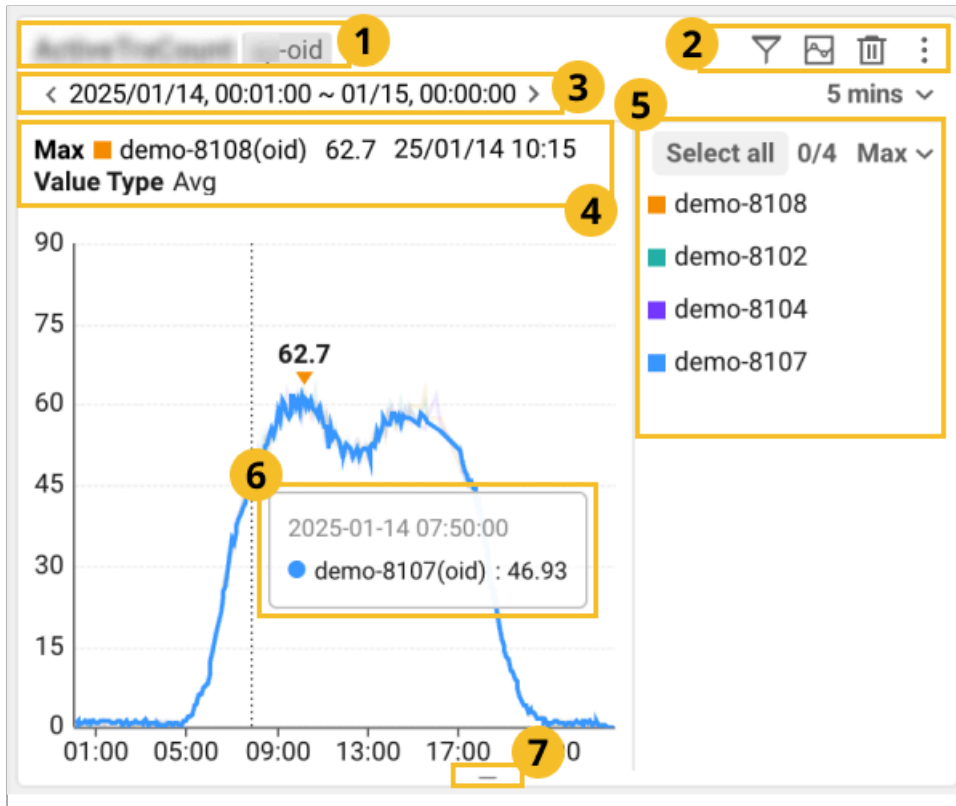
- To change the query time range and then change it back to a recent time, select .

If the **Apply chart zoom in batch** option is enabled at the top of the screen, the  **Time batch reset** button appears. By selecting this button, you can reset the time for all widgets.

- ! By using the **Batch Change** feature, you can change the time for all widgets and change the monitoring target and interval. For more information, see [the following](#).

Chart details

See details for each item displayed in the chart widget.



- **1 Metric name and category:** It displays the metric name of the chart widget. On the right of the metric, the categories of the query targets are displayed in tag format. The category name may differ depending on the product.
 - `*oid` : It is displayed when an individual agent is selected.
 - `project` : It is displayed when the target is selected on a project basis.
 - `okind` : It is displayed when a group classified as `whatap.okind` is selected in the agent configuration.
 - `onode` : It is displayed when a group classified as `whatap.onode` is selected in the agent configuration.
- **2 Widget option:** Option menu of the chart widget. For more information, see [the following](#). For more information about the interval options, see [the following](#).
- **3 Time:** It displays the time range for which data in the chart is viewed. By selecting `<` or `>`, you can adjust

the time range for a query.

- **4 Data:** You can check the maximum value (**Max**) of the data viewed from the monitoring targets and the data merge method (**Value Type**).
- **5 Legend:** It displays the monitoring targets as a legend. For more information about the display of legend, see [the following](#).
- **6 Chart:** It displays the viewed data in a time series graph. If you hover your mouse pointer over the graph, you can see detailed information through a tooltip. By dragging a specific area of the chart, you can change the time according to the dragged range for a view.
- **7 Resize:** You can adjust the size by dragging the center of the bottom of the chart widget.

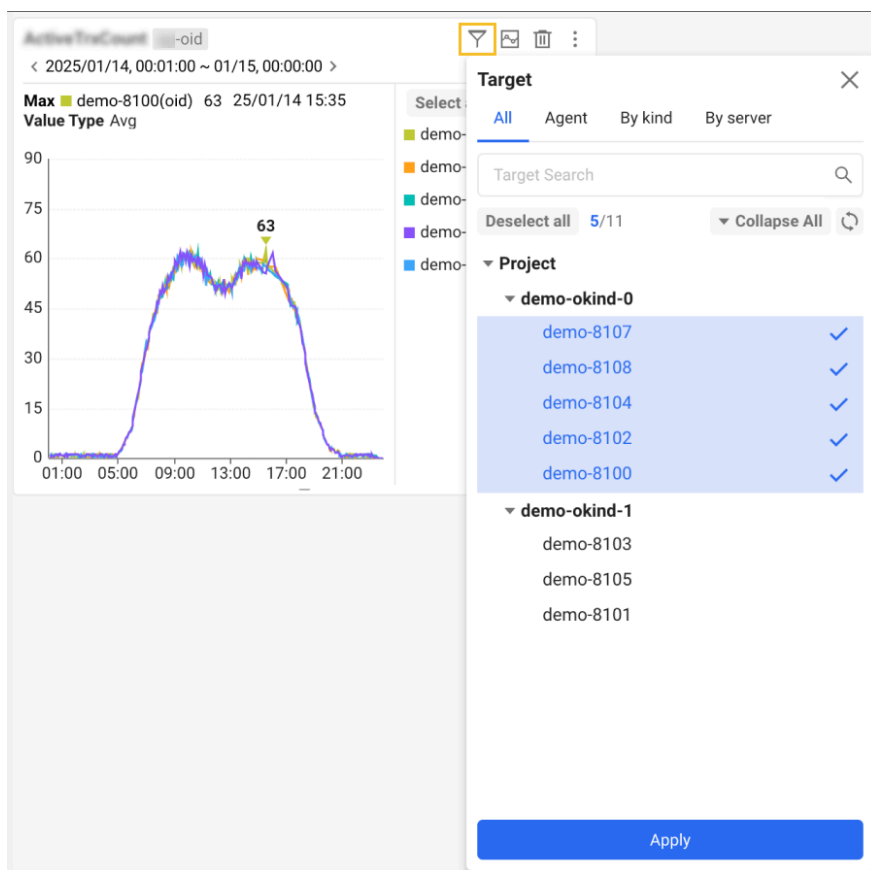
Using the chart widget options

Various features can be used through the option buttons on the upper right of the chart widget.

Filtering targets

You can add or remove monitoring targets to/from the chart widget added to the screen.

1. If you select , a pop-up menu appears for you to select monitoring targets.



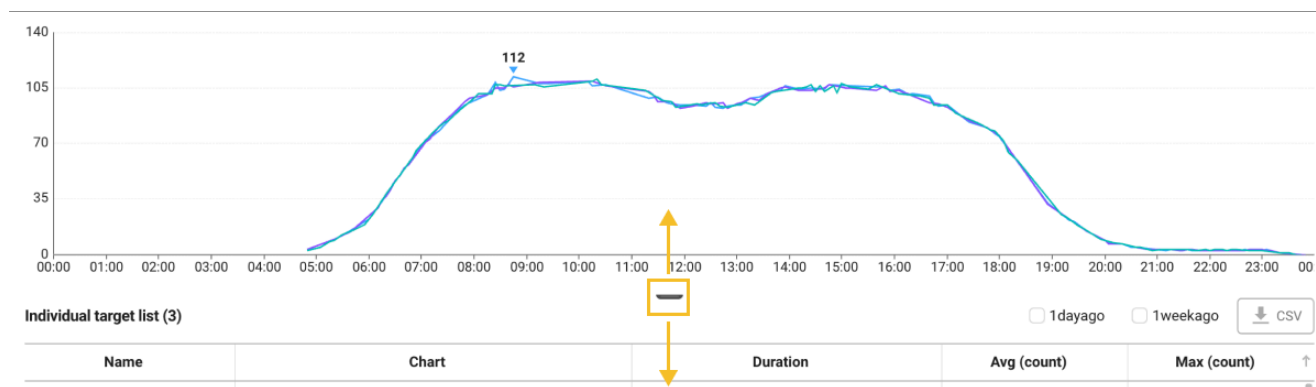
2. Select the desired targets according to the instructions on the screen.
3. Select **Apply**.

Viewing the details

To check the detailed information about the chart data in a wider screen, select . The **View details** window appears. You can view charts, averages, and maximum values for individual targets.



- **1dayago:** You can compare the data with that from the previous day.
- **1weekago:** You can compare the data with that of the previous week.
- **↓ CSV:** The viewed data can be downloaded as a CSV file.
- You can resize the chart by dragging the icon at the bottom of the graph.

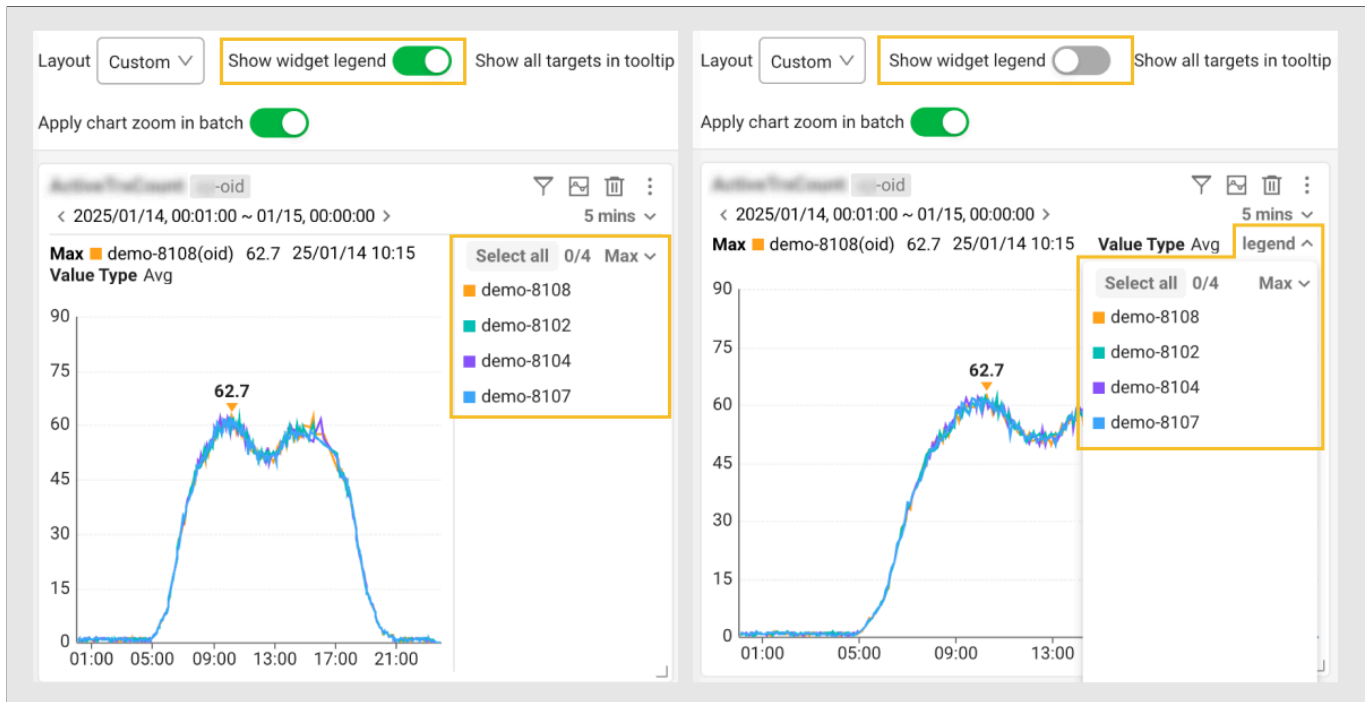


Removing the chart widget

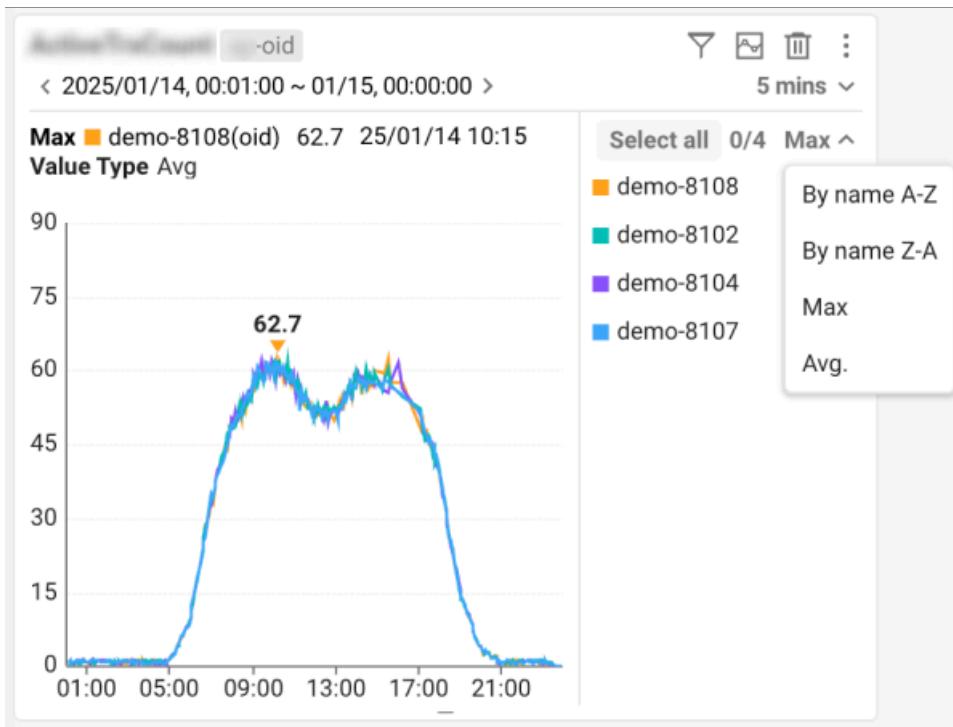
To remove any chart widgets placed on the screen, select .

Using the legend

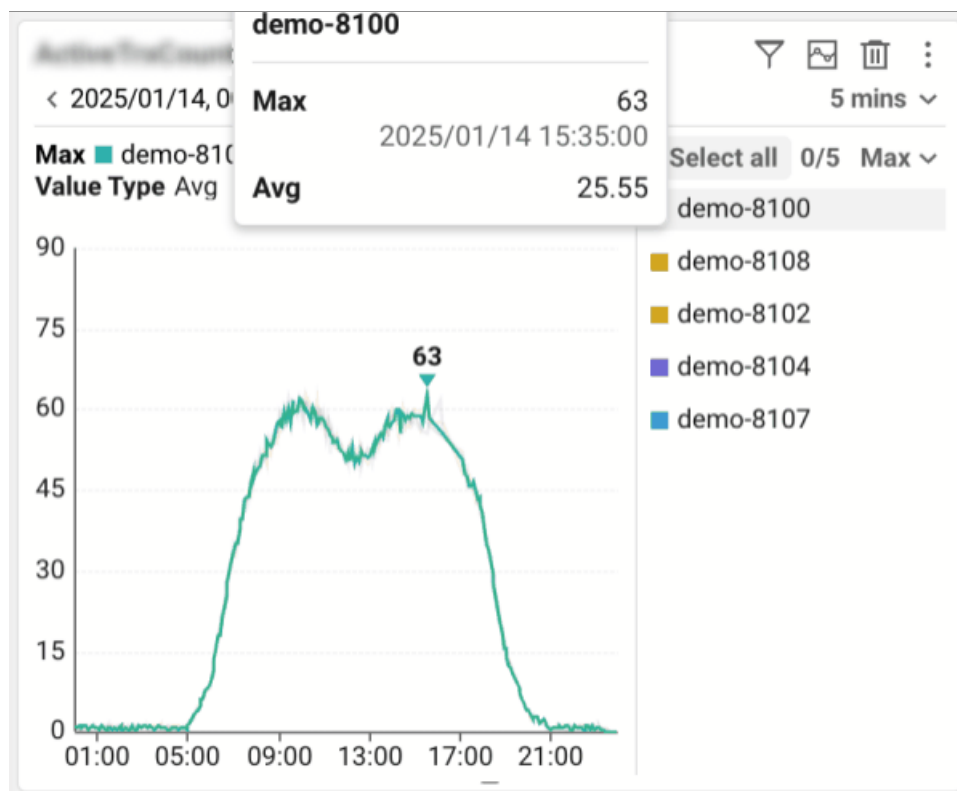
You can check the monitoring targets displayed on the chart through the legend and then sort or filter them by desired criteria. You can show or hide the legend on the right of the widget's chart by turning on or off the **Show widget legend** option at the top of the screen. With the option turned off, you can check and select the monitoring targets by clicking **legend**.



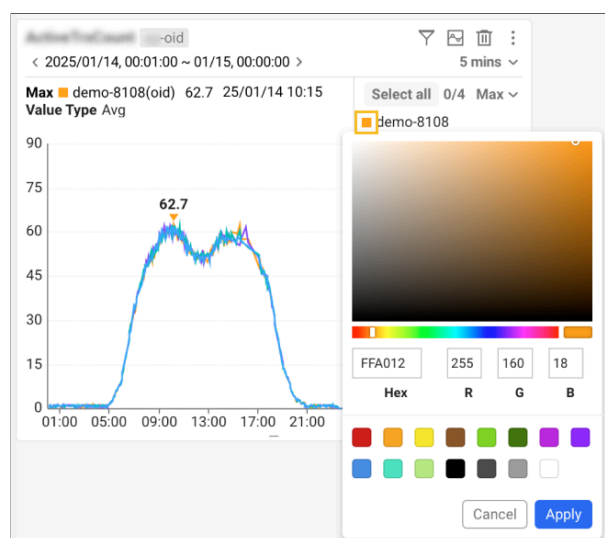
- You can select or unselect monitoring targets by selecting **Select all/Deselect all**.
- In the legend list, you can sort monitored objects by name, maximum value, or average value.



- If you hover the mouse over each item in the legend list, the highlighted graph for the monitoring targets appears and a tooltip indicating the specific time at which the maximum value is recorded. Alternatively, you can filter by selecting a target from the legend list.



Changing the legend color



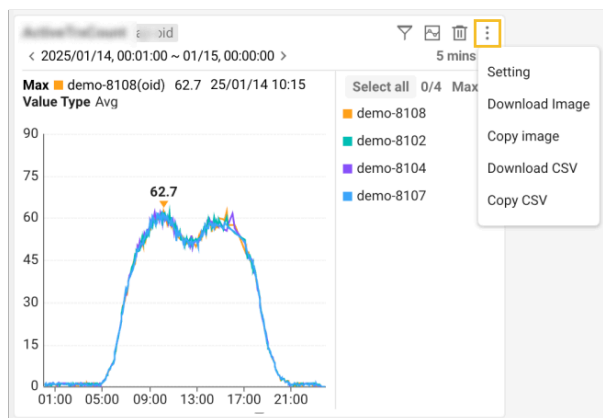
You can change the graph color corresponding to a monitoring target. In the legend list, select the square box

on the left of the target to change its color. Select a desired color and then click **Apply**.

! The set colors are not saved via the preset feature. This feature is to be provided in a future update.

Using additional options

If you select  on the upper right of the chart widget, you can select the following additional options.



- **Setting**
 - **Use Y axis automatically:** The size of the Y-axis can be automatically adjusted within the searched range.
 - **Show maximum value:** You can display the maximum value in the graph area of the chart.
- **Download Image:** You can download the chart as an image file.
- **Copy image:** You can copy the chart to the clipboard as an image format.
- **Download CSV:** You can download the data searched from the chart as a CSV file.
- **Copy CSV:** You can copy the data viewed in the chart to the clipboard in a CSV format.

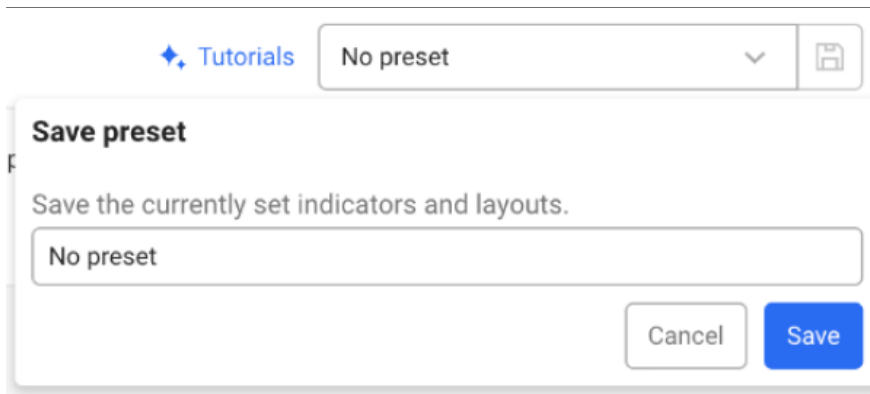
Setting presets

You can save and load the settings and layouts for widgets on the dashboard. You can create a new preset by adjusting its size and placing it on the desired position.

- ! You can save or modify the presets with the **Edit project** role. For more information about the member roles, see [the following](#).

Creating a new preset

1. Place the widgets in the desired format on the dashboard. You can also resize and place only the widgets to be frequently checked.
2. On the upper right of the screen, select .
3. Enter a new preset name.



4. Select **Save**.

You can see the newly saved preset from the preset list.

- !
- If any changes are made to the newly created preset, save the preset again. Select  and then save the preset with the same name. Any changes are overwritten in the existing preset.
 - If you go to another menu without saving changes on the dashboard, the changes are not saved.
 - Presets are saved on a project basis and can be shared with other users.
 - The targets saved in the preset are the layout of the dashboard, the widgets placed on the dashboard, and the options (**Target**, **Interval**) modified in the widgets. The **Time** option at the top of the screen is not saved.
 - To change the widget query time after loading a preset, use the **Batch Change** feature. For more information, see [the following](#).

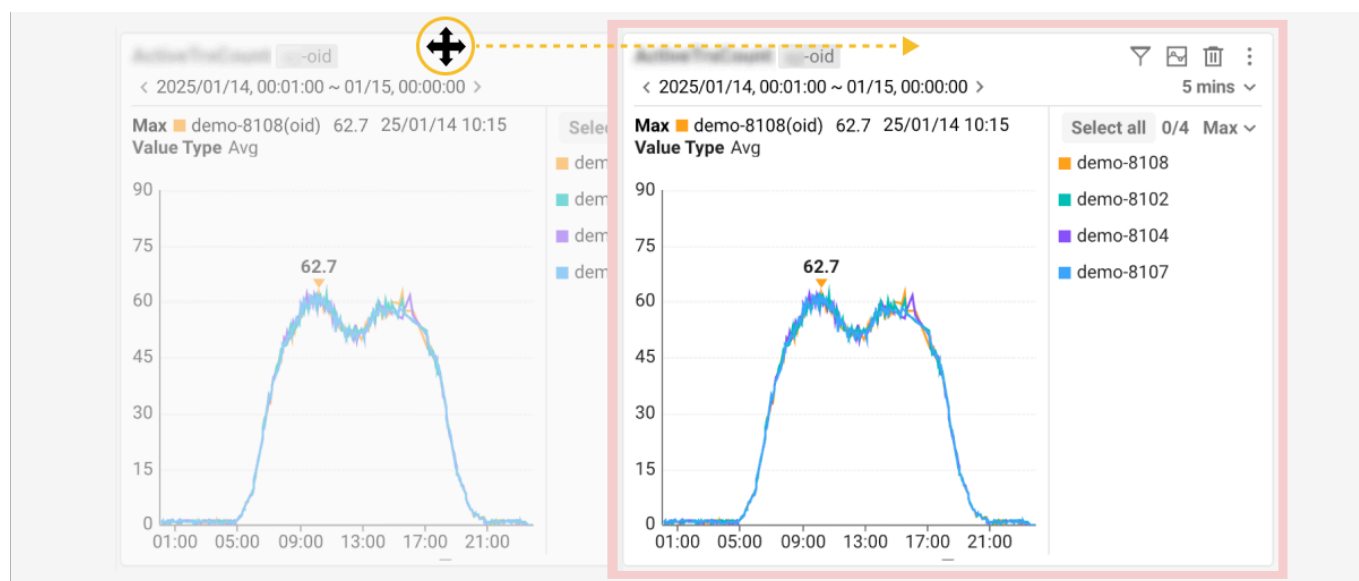
Deleting presets

If you have unused presets, you can delete them from the preset list. Select **×** on the right of the item to delete from the preset list.

Editing a dashboard widget

The widgets placed on the dashboard can be placed and resized to the user's desired location.

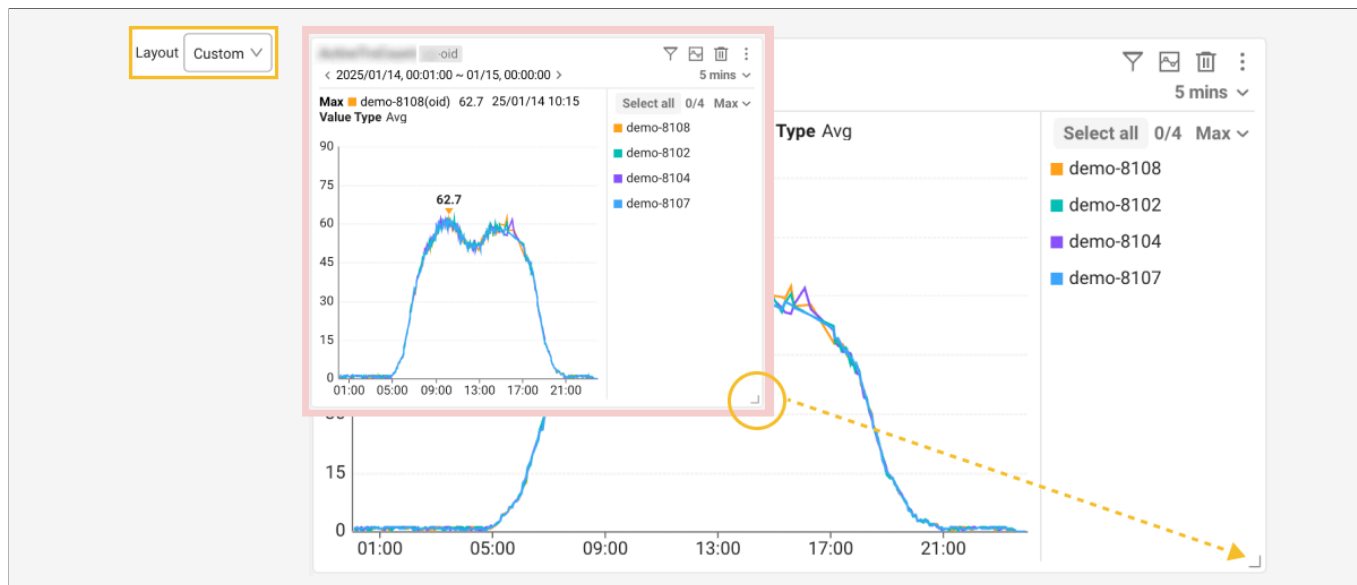
Moving a widget



When you move the top part of the widget by using the mouse pointer, it changes to the **+** shape. At this time, you can move the widget by dragging with your mouse to the desired position.

Resizing a widget

If **Layout** among the dashboard options is set to **Custom**, you can resize the widget.

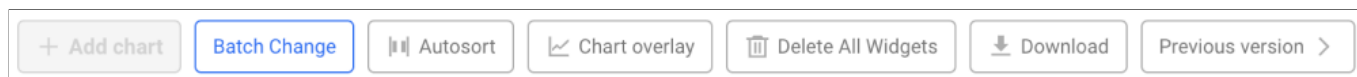


Click and hold the mouse on the  element at the lower right of the widget, and then drag it to the desired size.

ⓘ If **Layout** is not set to **Custom**, you can adjust only the height of the widget. Resize the widget with the bottom center of the widget clicked with your mouse.

Using the dashboard options

You can apply various settings to the chart widget at once using the Option menu located at the top of the screen. You can also customize the dashboard layout to your desired format.



Batch update of the chart widget options

It provides the feature to apply detailed options to the chart widgets on the screen at once. On the upper left of the screen, select **Batch Change**. If the **Batch Change** window appears, set the desired options and then select **Apply**.

Batch Change

Time

<

2025/01/15 00:00 ~ 2025/01/15 23:59

23 hr

>

Target

All

Target Search

Deselect all 1/4

Collapse All

▼ Project

demo-master ✓

master

demo-slave

Interval

no change

5 mins

1 hr

Apply widget

Select all

☐ Active Sessions

2025/01/13, 00:02:00 ~ 01/14, 00:01:00

db-oid 5 mins

☐ Wait Sessions

2025/01/13, 00:02:00 ~ 01/14, 00:01:00

db-oid 5 mins

☐ [XOS] CPU

2025/01/13, 00:02:00 ~ 01/14, 00:01:00

db-oid 5 mins

☐ [XOS] CPU_sys

2025/01/13, 00:02:00 ~ 01/14, 00:01:00

db-oid 5 mins

☐ [XOS] Disk_reads

2025/01/13, 00:02:00 ~ 01/14, 00:01:00

db-oid 5 mins

☐ [XOS] Disk_writes

2025/01/13, 00:02:00 ~ 01/14, 00:01:00

db-oid 5 mins

Cancel

Apply

- **Time:** You can set the data query range for the chart.
- **Target:** Select a monitoring target. You can select an individual agent or a group classified by the agent configuration.
 - **Agent:** You can select an individual agent or all agents.
 - **By kind:** Agents can be monitored in each group classified by `whatap.okind` in the agent configuration.
 - **By server:** Agents can be monitored in each group classified by `whatap.onode` in the agent configuration.
- **Interval:** You can adjust the data aggregation unit to various time intervals.

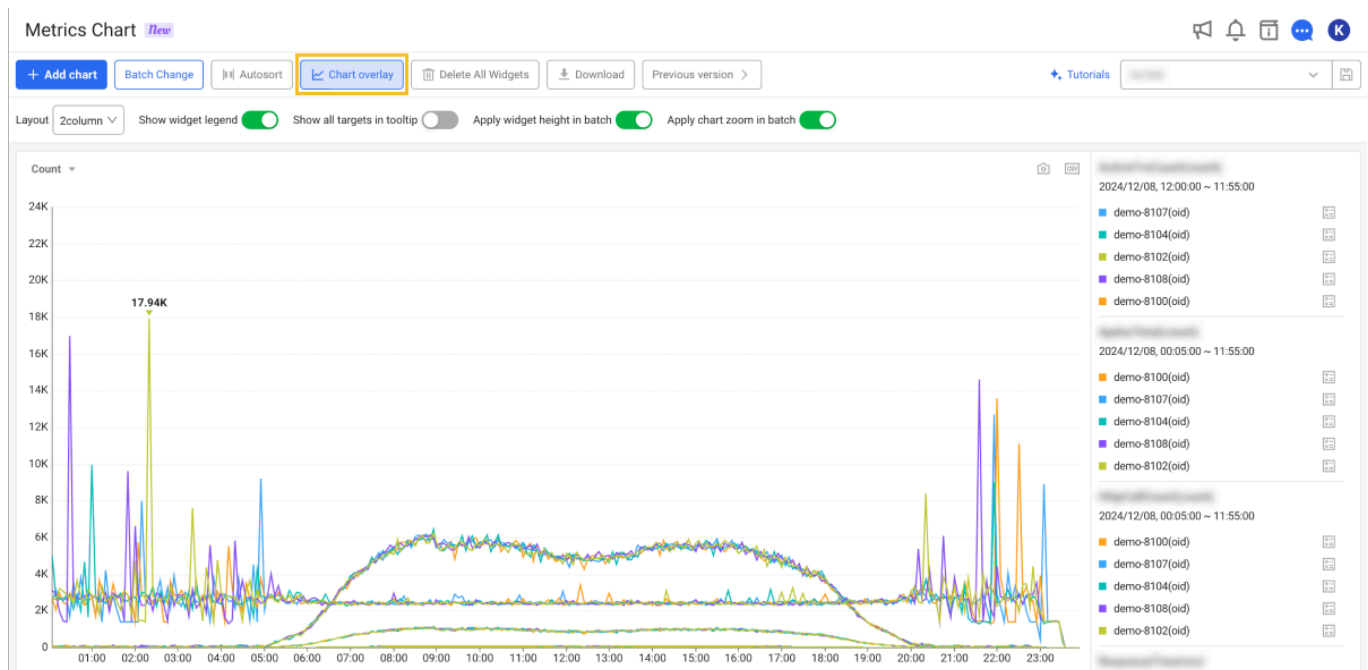
- **Apply widget:** Select a chart widget to apply the options to change. You can apply changed options to the selected widget at once.

Auto-aligning the dashboard

To auto-align the widgets placed on the dashboard, select **Autosort**. It auto-aligns widgets in order of their width and height, from smallest to largest. If the widths are the same, it aligns them in order of height, from smallest to largest.

Comparing multiple charts

This feature allows you to view data from multiple charts placed on the dashboard in a single chart. This feature allows you to compare multiple metrics from the same time period with others. Select **Chart overlay**. You can place overlay charts on the screen.



- If you select **Count** on the upper left of the overlay chart, a list appears where you can change the data criteria for the x-axis chart. You can compare the data of the chart by selecting a desired item.
- : You can download the chart as an image file based on the current view.
- : You can download the chart as a CSV file based on the current view.

- In the right section, you can see a list of agents for each metric. The colors of individual agents match the graph colors on the chart.

Batch deletion of placed widgets

You can delete multiple widgets placed on the dashboard all at once. Select **Delete All Widgets**. When the confirmation message appears, select **Delete**.

Downloading as a CSV file

You can download data from the widgets placed on the dashboard as a CSV file. Select **Download > Download CSV**. You can check the data of metrics for each time zone.

Downloading as an image file

You can download all the widgets placed on the dashboard as a single image file. Select **Download > Download Image**. You can save a captured image of the dashboard screen as is.

Using the previous version

To use the **Metrics Chart** menu of the previous version, select **Previous version**. Alternatively, you can see in the  **Sitemap > Analysis > Metrics Chart *Old*** path.

 For more information about **Metrics Chart *Old***, see [the following](#).

Using other options

Other options below allow you to adjust the layout of the dashboard or easily and conveniently batch set the applied options to the widgets.



- **Layout:** You can set the number of columns in the dashboard (or the number of widgets that can be placed in

a row). You can set up to 4 columns. If **Custom** is selected, widgets can be placed regardless of the number of columns.

❗ In **Custom**, you can freely adjust the width and height of the widget. If you select another option, you can only adjust the height of the widget.

- **Show widget legend:** Turning the option on or off shows or hides the legend on the right of the widget's chart. With the option turned off, you can check and select the monitoring targets by clicking **legend**. For more information about the use of legend, see [the following](#).
- **Show all targets in tooltip:** When you hover your mouse pointer over the chart, you can see the value for the point on the graph through a tooltip. Turn on the option to view data for all targets in the time zone where the mouse pointer is located through the tooltip.
- **Apply widget height in batch:** Enabling this option forces all widgets to have the same height.
- **Apply chart zoom in batch:** When changing the data query time of an individual chart widget, turn on this option to also change the chart times for other widgets.

Using the tutorial

When you select **Tutorials** on the upper right of the screen, you can check the changes after the [Service 2.8.0](#) update.

Metrics Trend Analysis

Metrics trend analysis provides a feature that visualizes metric changes within the past 30 days, allowing you to compare and analyze trends.

You can view key database performance metrics by period or time, helping you quickly identify patterns at specific moments or time intervals.

Default Screen

Metrics trend analysis consists of a **top option bar** and a **bottom chart area**.

When you select the viewing range and metrics in the option bar, trend graphs that match those conditions are automatically displayed.

Option Bar

The option bar provides functions such as time range selection, metric selection, and preset management.

Time Window

Use this option when you want to view data only for a specific time period.

When the toggle is enabled, the time range input fields become active. The default state is disabled.

For example, to analyze data during business hours (09:00–18:00), enable the Time Window and set that range.

Instance

Select the database instance to view.

A list of instances connected to the project is displayed, and you can quickly search for an instance using the search box.

Metrics

Select the metrics you want to view.

Metrics are categorized by field, allowing you to choose specific items for display.

Preset

Save and load frequently used chart configurations.

- The click states of the top date charts are saved in the preset.
- The click states of the lower date trend charts are **not** saved.

Save a Preset

Save the configured metrics and layout as a preset.

1. Set **Time Window**, **Metrics**, and other options as desired.
2. Click the  icon at the top right of the screen.
3. In the **Save as Preset** window, enter a name and click **Save**.

Load a Preset

1. Select a saved preset from the preset dropdown list.
2. The saved **Time Window**, **Metrics**, and **Chart** settings are applied automatically.

✔ Saving presets allows you to view charts immediately without repeating the same setup.

Chart

The chart area in **Metrics trend analysis** visualizes changes in the selected metrics.

You can compare trends, averages, and maximum values over specific periods.

You can add new charts as needed or remove unnecessary ones.

Add a Chart

1. (Optional) Set the **Time Window** in the top option bar.
2. Select the desired field under **Metrics**.
3. Click the + **Add Chart** button at the top right to add a chart.

Delete a Chart

Click the  icon at the top right of the chart you want to delete.

Widget

The bottom chart area in **Metrics trend analysis** includes **Interval** and **Open Link** functions.

These allow for detailed data analysis of specific periods or direct navigation to the detailed monitoring view for that time range.

Interval

Change the aggregation unit of the displayed data.

You can view data in hourly or 5-minute intervals.

By default, the data is displayed hourly; to see more detailed data, switch to the 5-minute interval view.

Open Link

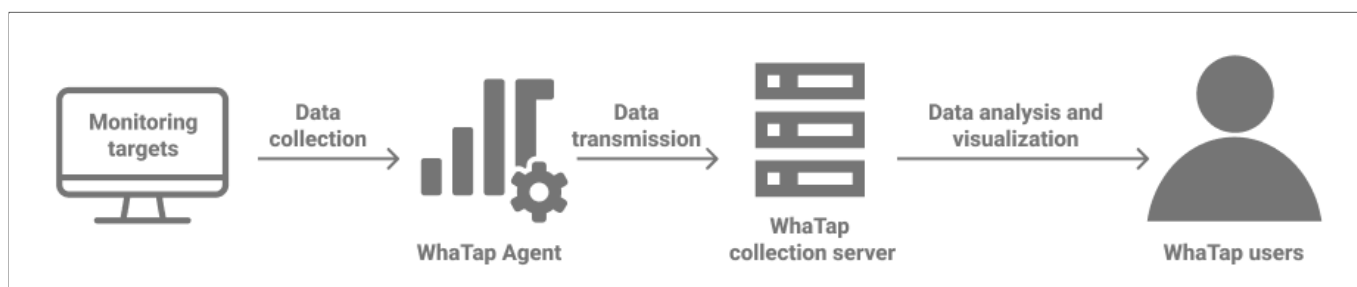
Click a bar in the chart representing a specific time range to move directly to that time's detailed monitoring page.

- ⓘ The **Open Link >** button becomes active only when a specific time bar is selected.
You can then view detailed metrics for that selected time range.

Metrics

What is the metrics?

WhaTap collects data from monitoring targets and provides them to users. The data collected from agents is called **metrics**.



Metrics consists of the following components:

- **Category**: A group of related metrics
- **Tags**: Data containing unique identifiers to find the targets to collect
- **Fields**: Metrics collected from the agent
- **Time**: Time when the metric has been collected
- **Oid**: Unique number of the agent that has collected metrics

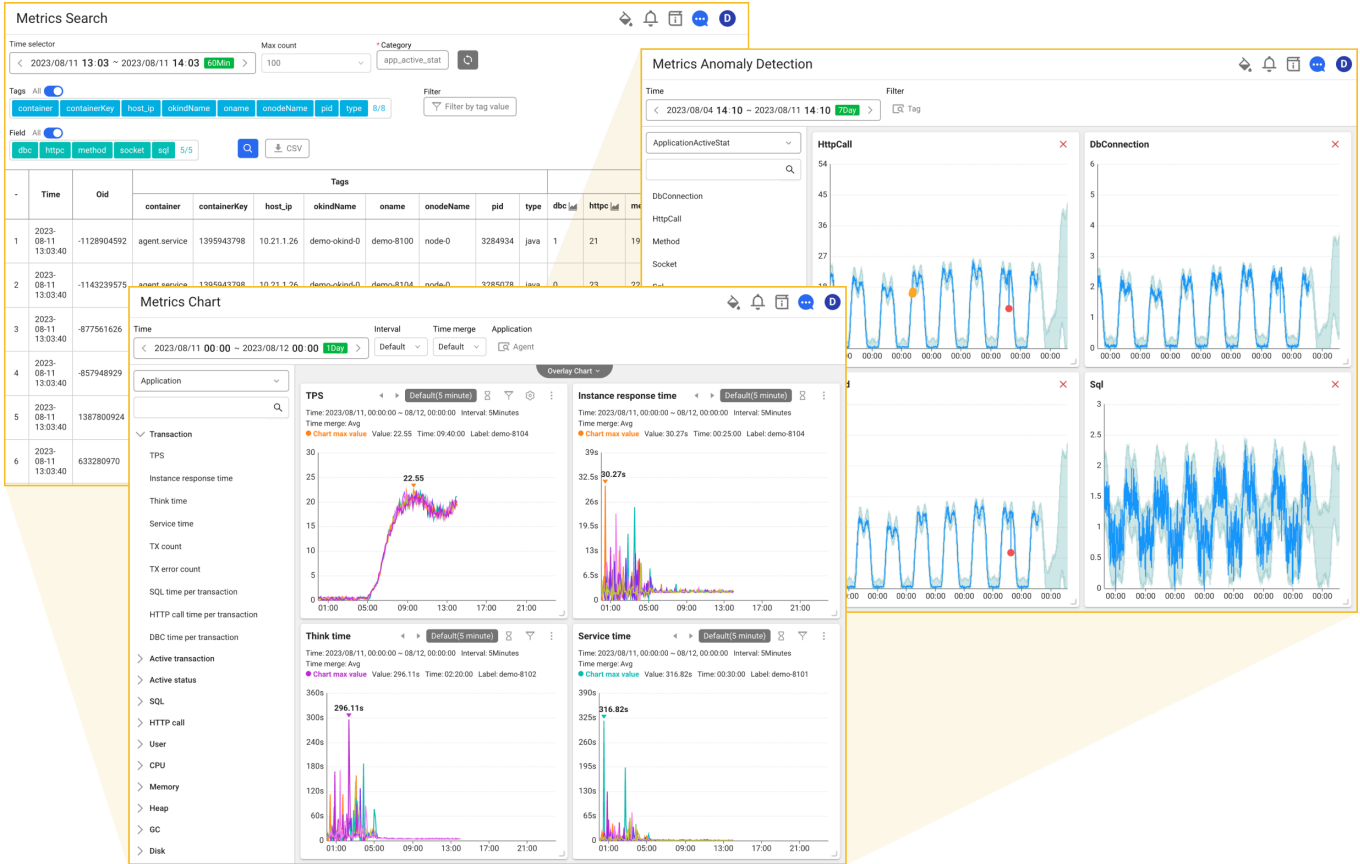
To view collected data, select a project on the Home screen, and then select [Sitemap](#) > [Metrics Search](#).

For example, select a category named `db_oracle_dma_counter` and view the related metrics. You can see the **(tags)** and **(fields)** that correspond to the category as follows.

Viewing and visualizing the metrics data

WhaTap provides a list of original data collected according to the conditions specified by user and various charts

visualized for convenience as follows. See the menus such as **Metrics Search** that can look up the original data of metrics, **Metrics Chart** that can look up metrics data through visualized charts, and **Metrics Anomaly Detection** that can detect out-of-pattern anomalies by comparing with the patterns of metrics learned by AI.



Oracle Pro Metrics

Database metrics are a comprehensive collection of database workloads and resource usages. The WhaTap monitoring service continuously collects metrics to identify bottlenecks in the database and provides data to help identify the root causes of performance issues.

For more information about columns, see [the link](#).

agent_status_summary

This category collects metrics related to agent status every 10 seconds.

Fields

Field	Type	Unit	Description
inActTime	-	Millisecond(ms)	Amount of time the agent remains inactive
isActive	Boolean	-	Whether the agent is active or not
isRestart	Boolean	-	Whether the agent was restarted recently (<code>true</code> / <code>false</code>)
lastActTime	-	Millisecond(ms)	Last time when the agent was activated (<code>0</code> : Disabled)
oid	-	-	Unique IDs for each agent in the project
startTime	-	Millisecond(ms)	Timestamp when the agent was started.

db_oracle_dma_counter

This category is useful for collecting overall performance metrics of the Oracle database to monitor the status

and performance of the system.

Tags

The following lists the tags that are collected in the `db_oracle_dma_counter` category.

Tag	Description	Remarks
<code>dbName</code>	Database name.	-
<code>oname</code>	Agent name	Unique value

Field

The following lists the fields that are collected in the `db_oracle_dma_counter` category.

Metrics	Description	Category
<code>cpu_used_by_this_session</code>	CPU time (in 10 milliseconds) used by the session, from the user call start time to its end time.	Performance
<code>db_time</code>	Elapsed time (in microseconds) to perform user-level calls to the database.	Performance
<code>active_sessions</code>	Number of active sessions.	Connection
<code>connect_error</code>	Number of connection errors.	Connection
<code>consistent_gets</code>	Number of consistent data block read operations.	I/O
<code>cpu</code>	CPU utilization used in the database.	Performance
<code>cpu_cores</code>	Number of available CPU cores	Performance
<code>data_blocks_consistent_reads_undorecords_applied</code>	Number of Undo records applied in consistent data block reads.	I/O

Metrics	Description	Category
db_block_changes	Number of blocks changed in SGA.	I/O
db_block_gets	Number of direct read operations from blocks in the SGA.	I/O
db_file_scattered_read_time_waited	Time waited for scattered read events.	I/O
db_file_scattered_read_total_timeouts	Number of timeouts occurred during the scattered read events.	I/O
db_file_scattered_read_total_waits	Total number of waits occurred during the scattered read events.	I/O
db_file_sequential_read_time_waited	Time waited for sequential read events.	I/O
db_file_sequential_read_total_timeouts	Number of timeouts occurred during the sequential read events.	I/O
db_file_sequential_read_total_waits	Total number of waits occurred during the sequential read events.	I/O
direct path read time waited	Time waited for direct path reads.	I/O
direct path read total timeouts	Number of timeouts occurred during direct path reads.	I/O
direct path read total waits	Total number of waits occurred during direct path reads.	I/O
direct path write time waited	Time waited for direct path writes.	I/O
direct path write total timeouts	Number of timeouts occurred during direct path writes.	I/O
direct path write total waits	Total number of waits occurred during direct path writes.	I/O

Metrics	Description	Category
enq:TX_-_row_lock_contention_time_waited	Time waited for row lock contention.	I/O
enq:TX_-_row_lock_contention_total_timeouts	Number of timeouts occurred for row lock contention.	I/O
enq:TX_-_row_lock_contention_total_waits	Total number of waits occurred for row lock contention.	I/O
enqueue_waits	Total number of waits during queue conversion or import due to queue import delays.	I/O
execute_count	Total number of commands that executed SQL statements.	Performance
fatal_count	Number of fatal errors.	Info
iowait_cpu	Percentage of CPU time spent in I/O wait state.	Performance
latch free time waited	Time waited in latch-free state.	I/O
latch free total timeouts	Number of timeouts occurred in latch-free state.	I/O
latch free total waits	Total number of waits occurred in latch-free state.	I/O
library cache lock time waited	Time waited in library cache lock state.	I/O
library cache lock total timeouts	Number of timeouts occurred during library cache locks.	I/O
library cache lock total waits	Total number of waits occurred during library cache locks.	I/O
library cache pin time waited	Time waited in library cache pin.	I/O
library cache pin total timeouts	Number of timeouts occurred in library cache pin.	I/O

Metrics	Description	Category
library_cache_pin_total_waits	Total number of waits occurred in library cache pin.	I/O
lock_wait_sessions	Number of waits to acquire the current row lock.	Connection
logons_current	Number of active logons.	Connection
long_running_sessions	Number of long running sessions.	Connection
memory	Percentage of memory in use.	MySQL Resource
opened_cursors_current	Number of open cursors.	Connection
parse_count_(hard)	Total number of syntax analysis calls (actual syntax analysis).	Performance
parse_count_(total)	Total number of syntax analysis calls (hard, soft, describe).	Performance
physical_reads	Number of data blocks read from disk.	I/O
physical_writes	Number of blocks in which data has been written to disk. Sum of <code>physical_writes_direct</code> and <code>physical_writes_from_cache</code> .	I/O
pq_sessions	Number of parallel query sessions.	Connection
redo_entries	Number of Redo entries created.	I/O
rollback_changes_-_undo_records_applied	Number of Undo records applied due to rollback changes.	I/O
session_logical_reads	Number of data blocks read from memory. Sum of <code>db_block_gets</code> and <code>consistent_gets</code> .	I/O
sorts_(disk)	Number of sort operations occurred on disk.	I/O

Metrics	Description	Category
<code>sorts_memory</code>	Number of sort operations occurred in memory.	I/O
<code>sys_cpu</code>	Sum of CPU time used by the session.	Performance
<code>total_elapsed_time</code>	Cumulative total elapsed time of all requests since the last database instance restart.	Performance
<code>total_sessions</code>	Total number of sessions.	Connection
<code>transaction_tables_consistent_reads_undo_records_applied</code>	Number of Undo records applied to consistent read operations in the transaction table.	I/O
<code>txn_sessions</code>	Number of transaction sessions.	Connection
<code>user_commits</code>	Number of user commit operations per transaction.	I/O
<code>user_cpu</code>	CPU % used for user tasks.	Performance
<code>user_rollbacks</code>	Number of user rollback operations.	I/O
<code>wait_sessions</code>	Number of sessions waiting for events.	Connection
<code>warning_count</code>	Number of all warnings.	Info

The following describes the metrics collected via the `ora_ratio` option in the agent settings. For more information about the agent option configuration, see [the following](#).

Metrics	Description
<code>buffer cache hit ratio</code>	The number of times the data can be found directly in memory without reading it from the disk . A high number indicates a good database performance, while a low number indicates the increased disk I/O, which may cause poor performance.

Metrics	Description
soft parse ratio	SQL statements reuse frequency, Soft Parse is that a SQL statement that is in the cache is reused. A high value indicates that SQL statement are being reused frequently.
hard parse ratio	Parsed frequency of the SQL statements from scratch. A hard parse occurs when a SQL statement is run for the first time or when it does not exist in the cache. If this figure is high, it may increase the CPU usage of the database and result in poor performance.
library cache hit ratio	The percentage of successful search of SQL and PL/SQL statements from the Library Cache. A high number indicates that SQL statements and PL/SQL code are being reused effectively in the cache.
library cache pin hit ratio	Percentage of successful Library Cache Pin requests. The Library Cache Pin protects running SQL statements from being changed by other sessions. A high number indicates that this protection is effective.
library cache get hit ratio	If the rate of successful retrieval of objects from the Library Cache is a high number, it indicates that SQL statements and PL/SQL code are being reused effectively in the Library Cache.
latch hit ratio	The rate of successful latch requests , latch is a locking mechanism for controlling access to shared memory in the database. A high number indicates a low latch latency and good concurrency and performance of the system.
shared pool free mem ratio	The percentage of Free Memory in the Shared Pool . The shared pool is a memory area that stores SQL statements, PL/SQL code, and the data dictionary cache. If an appropriate amount of free memory is maintained, it can ensure efficient operation of the shared pool and optimal performance.

db_oracle_dma_tablespace

This category collects tablespace usage data of the Oracle database every one hour. Metrics in this category help you monitor the status and capacity of tablespaces to efficiently manage storage resources and proactively prevent database performance issues.

Tags

The following lists the tags that are collected in the `db_oracle_dma_tablespace` category.

Tag	Description	Remarks
<code>con_name</code>	It is the database name of the connected container. It is useful for distinguishing each Pluggable Database (PDB) in the multi-tenant environment.	-
<code>datfile</code>	Path and file name of the data file linked to each tablespace.	-
<code>oname</code>	Agent name	Unique value
<code>tablespace</code>	It is the name of tablespace.	-

Field

The following lists the fields that are collected in the `db_oracle_dma_tablespace` category.

Metrics	Description
<code>autoextensible</code>	It indicates whether the data file is automatically extensible with <code>YES</code> or <code>NO</code> .
<code>increment_by</code>	Number of blocks that increase at a time when the data file is extended automatically.
<code>maxsize</code>	Maximum size set when the data file is automatically extended.
<code>size</code>	Current allocated capacity of the tablespace.

Metrics	Description
<code>used</code>	Size of the tablespace in use.

XOS-related metrics

The following describes the metrics collected from the XOS agent.

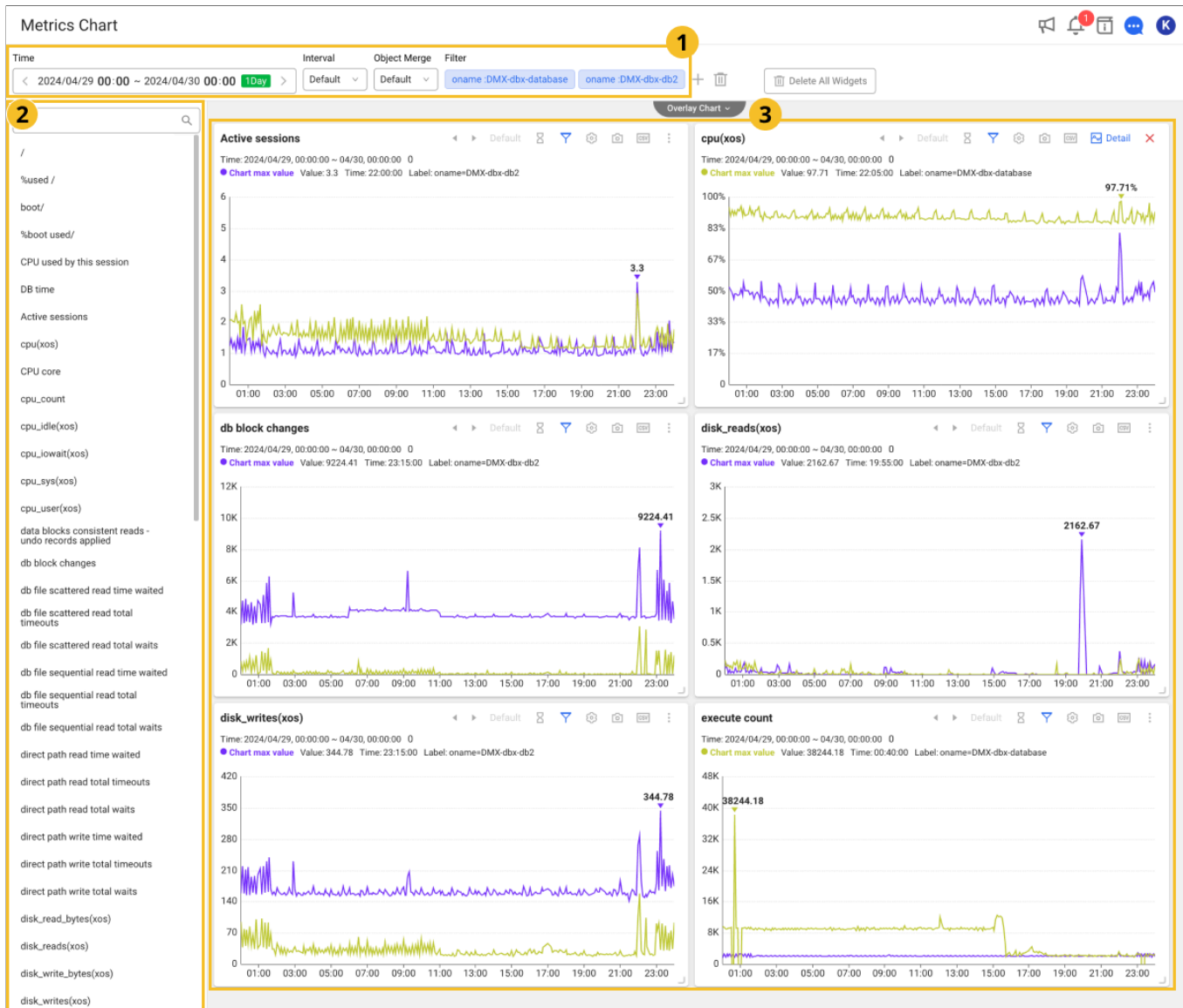
Metric	Unit	Description
<code>cpu(xos)</code>	pct	CPU utilization <code>cpu_user</code> + <code>cpu_sys</code> + <code>cpu_iowait</code>
<code>cpu_idle(xos)</code>	pct	CPU idle time
<code>cpu_iowait(xos)</code>	pct	CPU's I/O waiting time
<code>cpu_sys(xos)</code>	pct	CPU's kernel code execution time
<code>cpu_user(xos)</code>	pct	CPU's user code execution time
<code>disk_read_bytes(xos)</code>	byte	Number of data blocks read from disk
<code>disk_reads(xos)</code>	block	Number of buffers read from disk
<code>disk_write_bytes(xos)</code>	byte	Number of pages written to disk
<code>disk_writes(xos)</code>	block	Number of buffers written to disk
<code>mem(xos)</code>	pct	Memory utilization
<code>mem_available(xos)</code>	MB	Shared memory size
<code>mem_buffcache(xos)</code>	MB	Amount of memory used by the buff/cache
<code>mem_free(xos)</code>	MB	Memory not in use

Metric	Unit	Description
<code>mem_swapfree(xos)</code>	MB	Unused swap space size Because swap uses disk space, it may cause performance degradation.
<code>mem_swaptotal(xos)</code>	MB	Total swap size It is used when physical memory is insufficient.
<code>mem_total(xos)</code>	MB	Total memory size
<code>mem_used(xos)</code>	MB	Memory size in use
<code>net_rcv_bytes(xos)</code>	byte	Amount of data transmitted over the network
<code>net_rcv_packets(xos)</code>	count	Number of data packets transmitted over the network
<code>net_send_bytes(xos)</code>	byte	Amount of data transmitted over the network
<code>net_send_packets(xos)</code>	count	Number of data packets transmitted over the network

Metrics Chart

[Sitemap](#) > [Analysis](#) > [Metrics Chart](#) Old

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.
- **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.

 For more information on how to use the time selector.

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 data via the chart widgets in the **3** area.

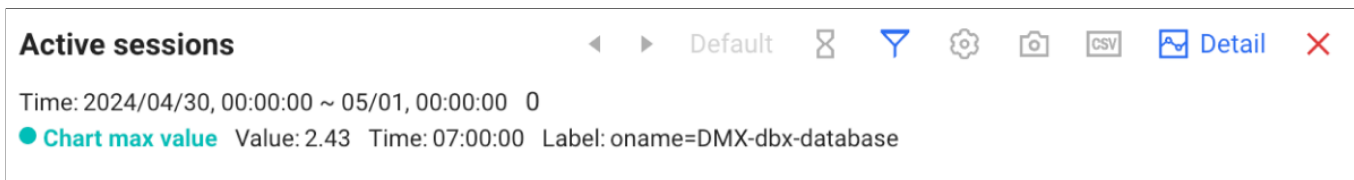
Data Merge

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 ①, 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: 📄 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 > **Analysis** > **Metrics Search**

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

Metrics Search

1 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.

Metric Anomaly Detection

You can compare the patterns of various metrics with the expected patterns learned by the AI. Anomaly detections that deviate from the expected patterns can be viewed through the graph chart. You can check recurring patterns based on the past data and use them to predict future metrics values.

- ❗ The widgets placed on the screen are not saved and reset when you go to another menu.
- It is similar to **Analysis > Metrics Chart** except for the pattern and anomaly display.
- The alert, **Anomaly detection** is based on the **Metric anomaly detection**. For more information about the **Anomaly detection** alert setting, see [the following](#).

Checking the widgets



- ❶ The bright color areas of the graph are the estimated patterns analyzed by the AI.
- ❷ The blue graph is the metrics trend of the project.
- If it deviates from the expected pattern analyzed by the AI, it displays dots on the graph in Orange and Red steps. The values that are far from the expected pattern range appear in red.

Placing the widgets



1. In **1 Time**, set a desired time interval. You can set the interval up to 1 month.
2. In **2 Filter**, select a range of the metrics.
3. In the following list, select a metric for monitoring.

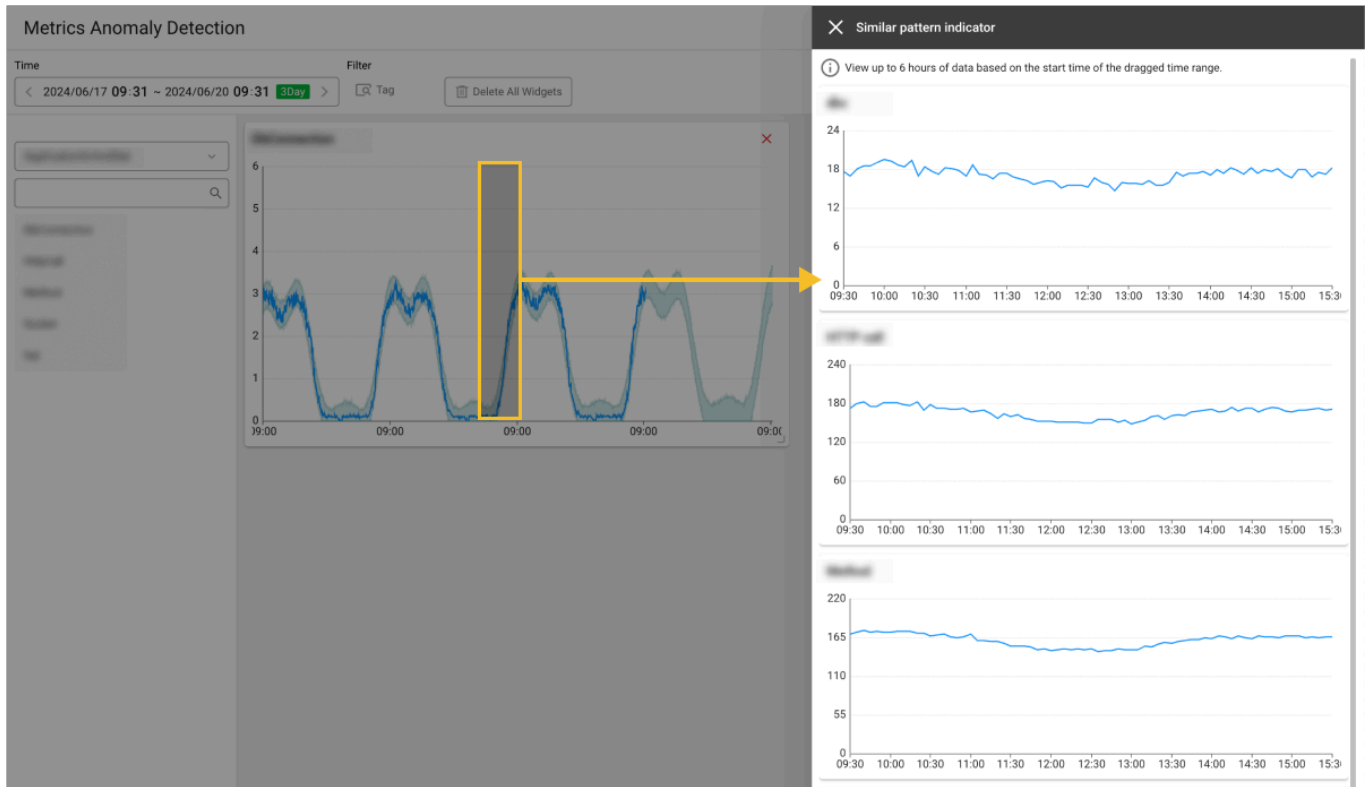
Place the selected metric to the right of the screen.

- Select **2 Tag** to check the metrics separately by selecting a monitoring target. A tag selection list appears on the right of the screen. Select desired items, and then add widgets.
- The widgets placed on the screen are not reflected in the chart even if the **Time** or **Tag** values have been changed.
- Select **X** on the upper right of the widget on the screen to delete.
- You can reposition a widget by dragging it while selecting the upper left part of the widget.
- You can resize a widget by selecting the upper right part of the widget and then dragging it.

Deleting a widget

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

Checking the related metrics



Drag part of the chart in the widget. The **Similar pattern indicator** window appears on the right of the screen. You can search the related metrics for up to 6 hours based on the start time of the time range you dragged.

Tablespace Size

Tablespace Size is a screen for monitoring and managing the usage of database tablespaces. You can quickly check the size change trend of each tablespace using a line chart, and detect space shortages in advance through daily changes and **Remaining Day**.

- **Tablespace Size Trend:** Use the line chart to track the capacity change trend of major tablespaces.
- **Disk Usage:** View disk usage and remaining days per mount point. Available when the XOS agent is installed.
- **Tablespace Size Detail:** Analyze the latest usage, daily changes, and remaining days for individual tablespaces.

Prerequisites

To display data on the Tablespace Size screen, [DB Configuration](#) and [Agent Configuration](#) are required.

DB Configuration

Grant permissions to the monitoring account so the agent can query tablespace and segment information.

```
GRANT SELECT ANY DICTIONARY TO whatap;
```

Or grant permissions to individual views.

```
GRANT SELECT ON dba_segments TO whatap;
```

Agent Configuration

This configuration is for collecting segment size (TOP 50) data. Based on the conditions below, the agent collects segment size data once a day at the configured time. Data may not be collected due to conditions such as `segments_min_size`. Without this configuration, segment size information will not be available.

Option	Type	Default	Description
segments	boolean	true	Enable/disable segment information collection
segments_hour	int	5	Time to collect segment information. Default is 5, which starts collection at 5 AM.
segments_min_size	int	10	Minimum segment size (MB) for collection
segments_exclude_sys	int	1	Exclude sys account segments (1: exclude, 0: include)
segments_max_count	int	50	Number of segments to collect per tablespace

Tablespace size collection options

- ⓘ Depending on the DB environment, performance issues may occur during tablespace size collection. Use the following options to adjust the collection method and interval.

Option	Type	Default	Description
tablespace	int	1	Select the tablespace size collection method - 0: Disable collection - 1: Query based on dba_free_space (most common) - 2: Query based on free_blocks for LMT (Locally Managed Tablespace) - 3: Query based on tablespace_usage_metrics view (per-datafile query not available, no Auto Ext info)
tablespace_split	boolean	false	Applied when the tablespace option is 1 - true: First retrieves the list of datafiles, then queries free space for each file individually to distribute the load. Use this when collection query load is high in large datafile environments.
tablespace_interval	int	60	Tablespace size collection interval (minutes) - Default: 60 minutes (1 hour)



✓ If collection is not smooth due to large DB size, try setting `tablespace=1` with `tablespace_split=true` to distribute the load, or switch to `tablespace=2` .

Tablespace Size

Basic Options

In the top filter area of the **Tablespace Size** screen, you can select the time range and target database to query.

- **Time:** Select the query time range. Up to 3 weeks can be queried; queries over 3 days are displayed as daily averages.
- **Instance:** Select the target DB instance to query.
- **Data Unit:** Select from MiB/GiB/TiB. All table size values are displayed in the selected unit.
- **Database:** Select the target database to query.

Tablespace Size Trend

Displays the size change of each tablespace as a **line chart**. Each tablespace is color-coded and can be identified in the legend at the bottom.

- Hover over the chart to see detailed values by time period.

Item	Description
Top 5 / 10 / 15 / 20	Select the number of top tablespaces to display on the chart
Size / Used	Switch between Size (total size) and Used (usage)

Disk Usage

! Available when the XOS agent is installed.

Displays disk usage per mount point. The **Latest Data** section shows the most recently collected data within the query range selected in the time selector. You can check the size, Usage (%), Increase AVG, and Remaining Day at that point in time.

Latest Data

Column	Description
Mount Point	Disk mount path
Size	Total disk size
Used	Used size
Free Size	Remaining free space
Usage(%)	Usage rate (visualized as a progress bar)
Increase AVG	Average daily increase over the query period
Remaining Day	Remaining available days based on the current growth rate

Daily Columns

Displays **Size** and daily change (**Inc**) for each date in the query period.

- **Search:** Filter by Label or Path.
- **CSV:** Export in CSV format.

In an ASM environment, both the **Disk Usage** tab and the **ASM Disk Usage** tab are displayed together.

ASM Disk Usage

The daily columns, search, and CSV features are the same as Disk Usage.

Column	Description
Label	ASM disk label
Path	ASM disk path
Size	Total disk size
Used	Used size
Free Size	Remaining free space
Usage(%)	Usage rate (visualized as a progress bar)
Increase AVG	Average daily increase over the query period
Remaining Day	Remaining available days based on the current growth rate

Tablespace Size Detail

Displays detailed usage for individual tablespaces. The **TOTAL** row at the top shows the overall total. The **Latest Data** section shows the most recently collected data within the query range selected in the time selector. You can check the size, Usage (%), Increase AVG, and Remaining Day at that point in time.

Latest Data

Column	Description
Container	Container (PDB) name. Displayed only in CDB environments; hidden in Non-CDB environments.
Tablespace	Tablespace name

Column	Description
Auto Ext	Auto extension status (YES/NO). Column visibility can be toggled in settings.
Inc By	Increment unit for auto extension. Column visibility can be toggled in settings.
Max Size	Maximum size limit. Column visibility can be toggled in settings.
Total Size	Total tablespace size
Used Size	Used size
Free Size	Remaining free space
Usage(%)	Usage rate (visualized as a progress bar)
Increase AVG	Average daily increase over the query period
Remaining Day	Remaining days based on current growth rate, calculated as $\text{Free Size} \div \text{Increase AVG}$

Daily Columns

Displays **Size** and daily change (**Inc**) for each date in the query period. Increases are shown as positive values and decreases as negative values, color-coded for distinction.

Additional Features

-  **Magnifier icon**: Displays a popup of the top 50 segments by space usage in the tablespace.
-  **Expand**: Expand the list of data files belonging to the tablespace.
- **Search**: Filter by container or tablespace.
- **CSV Download**: Export in CSV format. When downloading, you can select **Tablespace** or **Datafile** unit.

Table Settings

Click the  **Settings** button in the top right to configure table display options.

Item	Description
Column: Auto Ext / Inc By / Max Size	Toggle the display of <code>Auto Ext</code> , <code>Inc By</code> , and <code>Max Size</code> columns.
Warning(%)	Usage rate (%) for warning level; displayed in orange when exceeded. Default: 80
Critical(%)	Usage rate (%) for critical level; displayed in red when exceeded. Default: 100
Reset Settings	Reset all settings to default values

Segment Size (TOP 50)

Click the  **magnifier icon** in the tablespace row to display a popup showing the top 50 segments occupying the most space in that tablespace.

Click the  **Download** button at the top right of the popup to export the list shown in the popup to a CSV file. The column configuration and display format visible on the screen are reflected in the file as they are, and the saved file name is `table_size_top50.csv`.

 The `segments` option must be enabled in the Agent configuration. The agent collects segment size data once a day, and without this configuration, data will not be available.

Column	Description
<code>Tablespace Name</code>	Tablespace the segment belongs to
<code>Owner</code>	Schema that owns the segment
<code>Segment Name</code>	Segment name. Click to view object details in the Object Detail popup.
<code>Segment Type</code>	Segment type (TABLE, INDEX, LOBSEGMENT, etc.)
<code>Partition Name</code>	Partition name (if a partitioned table)

Column	Description
Size	Segment size. Size changes can be tracked via date columns.

Object Detail

Click a segment name in the Segment Size (TOP 50) popup to send a real-time query request to the agent and display detailed information about that object.

- **Object Basic Info:** owner, table_name, num_rows, blocks, avg_row_len, tablespace_name, partitioned, last_analyzed, compression, status, etc.
- **Column/Index:** Column list (column_id, column_name, nullable, data_type, data_default, num_distinct, num_buckets) and index information.

Data Interpretation Guide

Key Checkpoints

- **Tablespace with high Usage(%)**
Risk of space shortage → Consider cleaning up unnecessary data or expanding the tablespace.
- **Tablespace with high Increase AVG**
Rapidly growing → Check Remaining Day and take proactive action.
- **Short Remaining Day**
Available space will be exhausted soon based on current growth rate → Immediate action required.
- **Tablespace with Auto Ext set to NO**
Auto extension is disabled; errors may occur when space is exhausted → If usage is high, consider manual expansion or enabling Auto Ext.
- **Sudden change in daily Inc value**
Large-scale data load or deletion occurred on a specific date → Investigate the cause.

- Check disk **Usage(%)** and tablespace **Usage(%)** together

Even if a tablespace has free space, expansion is not possible if the disk is full.

Practical Use Scenarios

Scenario 1. Proactive Space Shortage Detection

1. Sort by **Usage(%)** in descending order in **Tablespace Size Detail**
2. Check **Remaining Day** for tablespaces with high usage
3. If remaining days are few, expand the tablespace or clean up unnecessary data

Scenario 2. Capacity Surge Root Cause Analysis

1. Check for rapidly increasing tablespaces in the **Tablespace Size Trend** chart
2. Identify the date when the increase occurred in the daily **Inc** column of that tablespace
3. Click the  **magnifier icon** for that tablespace to analyze segments occupying the most space
4. Review segment details for unnecessary indexes or ways to reduce data size; if none found, consider disk expansion

Scenario 3. Disk Capacity Management

1. Check mount points with high **Usage(%)** in **Disk Usage**
2. Use **Remaining Day** to estimate when the disk will be full
3. Compare with **Tablespace Size Detail** to identify which tablespaces are consuming the most disk space

Scenario 4. Auto Extension Configuration Check

1. Enable **Auto Ext / Inc By / Max Size** columns in settings
2. Among tablespaces with **Auto Ext** set to NO, check those with high **Usage(%)**
3. If necessary, enable auto extension or manually expand the tablespace

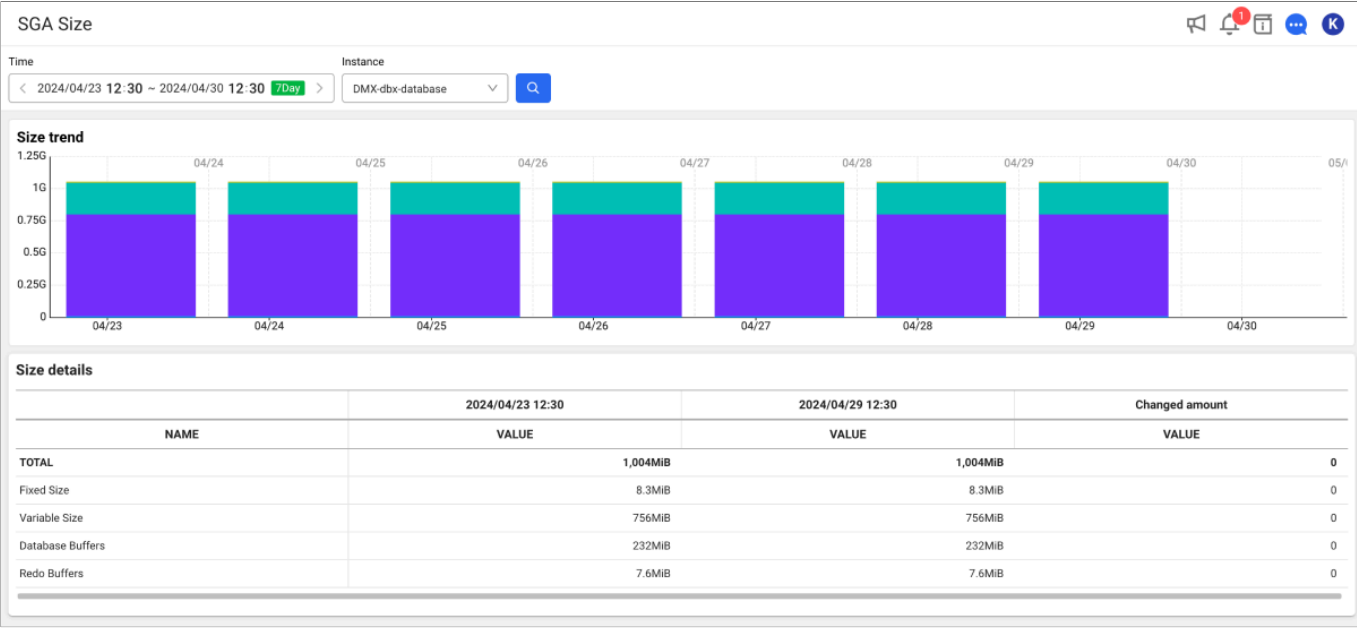
SGA Size

Home > Select Project > Stat/Report > SGA Size

You can monitor timely changes by saving summaries of System Global Area (SGA) that is an server memory area.

- **Performance optimization:** By monitoring the sizes and usage of various memory areas within the SGA to proactively prevent performance issues and optimally manage memory resources.
- **Bottleneck detection:** By early detecting the bottlenecks occurring in each memory area, the availability and stability of the system can be improved.
- **Resource management:** You can check whether memory resources are allocated appropriately to reduce unnecessary memory usage and efficiently distribute memory resources.
- **Troubleshooting:** By identifying which part of the SGA area are overused so you can quickly troubleshoot the cause of poor performance.

Basic usage guide



1. Set the time to search in **Time**.
2. Select an instance name to view the trend data of the instance.
3. Select .

 • The query time can be set up to 3 weeks.

Database Parameter

Home > Select Project > **Analysis** > **Database Parameter**

Using the database configuration inquiry, you can view the database settings of the past. You can compare parameter values in the database by date through the comparison function. By default, you can see the data of the current time.

If a problem occurs due to a parameter in the database, the cause can be quickly recognized through the database parameter comparison function.

Database Parameter

Reference date

Comparison date

Instance

2024/06/14

2024/06/13

DMX-dbx-database

View Changes

	2024/06/14 13:55:10	Today	2024/06/13 00:00:00
Key	Value		Value
shrd_dupl_table_refresh_rate	60		60
pdb_lockdown			
default_sharing	metadata		metadata
containers_parallel_degree	65535		65535
disable_pdb_feature	0		0
remote_recovery_file_dest			
cdb_cluster_name	ORA122		ORA122
cdb_cluster	FALSE		FALSE
max_pdb	4098		4098
target_pdb	1		1
common_user_prefix	C##		C##
noncdb_compatible	FALSE		FALSE
pdb_file_name_convert			
pdb_os_credential			
enable_pluggable_database	TRUE		TRUE

- 1. Set values for **Reference date** and **Comparison date**.
- 2. Select a desired instance in **Instance**.
- 3. Click .

To check the changes between the current time and the selected past date at a glance, select **View Changes**.

Database Parameter

Reference date

2024/06/14

Comparison date

2024/06/13

Instance

DMX-dbx-database

View Changes

	2024/06/14 13:55:10 Today	2024/06/13 00:00:00
Key	Value	Value
resource_manager_plan	SCHEDULER[0x4AC6].DEFAULT_MAINTENANCE_PLAN	SCHEDULER[0x4AC5].DEFAULT_MAINTENANCE_PLAN

Report

Home > Select Project > **Stat/Report** > **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.

Report types

The basic report is a form that is actually used by large enterprises, government agencies, and IT service companies. If you have any desired form, please send a request to support@whatap.io. The requested report template is to be added in the **Stat/Report** > **Report** list.

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

Daily DB Report

The daily database report allows you to check the database performance for the day in figures and charts. When creating a report, you can set the start date, start time, and end time if necessary, and you can select and view only specific instances. When all is selected, all instances are viewed at once.

Trend Chart

You can see metrics charts for each database instance. The maximum value of each metric is displayed in circle, and the time when the maximum value occurred is also displayed.

- Metrics drawn as charts may differ depending on the configuration.
- On the upper right of each chart, use [CSV](#) to download the data in an Excel file format.

SQL Top 10

You can check the statistics of SQL queries during the query period for all or selected instances. It retrieves only the top 10 sorted items in descending order based on the maximum value of query execution time.

Weekly DB Report

The weekly database report displays the database performances for the week with figures and charts. You can

enter the start date and see from 7 days prior to the day of inquiry.

Trend Chart

You can compare weekly metrics for each database instance with figures and charts. The maximum value during the query period is displayed in circle and the occurrence time is also displayed.

- ! • Metrics drawn as charts may differ depending on the configuration.
- On the upper right of each chart, use [CSV](#) to download the data in an Excel file format.

SQL Top 10

You can check the statistics of SQL queries during the query period for all or selected instances. It retrieves only the top 10 sorted items in descending order based on the maximum value of query execution time.

Monthly DB Report

This report allows you to check the database performance for a month with numbers and charts. You can set the start date and search up to 30 days from the start date. You can check the change amount in each metric over a month by instance, and you can also check the SQL query statistics of the top 10 items performed during the query period.

SQL Statistics

Home > Select Project > **Stat/Report** > **SQL Statistics**

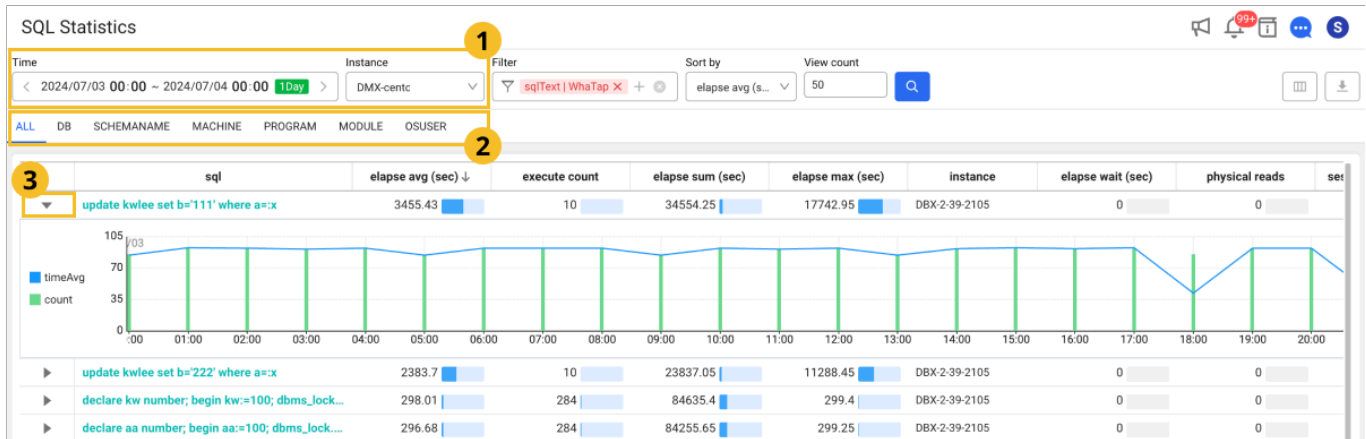
Statistics are created based on the SQL execution count, execution time, and waiting time within the active sessions collected by the agent, and a list of top SQLs for the search period is provided. It provides a basis for analyzing frequently performed SQL performance through the average and maximum execution time.

For quick performance analysis, you can search top SQLs for each frequently analyzed category (DB, USERNAME, MACHINE, PROGRAM). The main search criteria are as follows:

- **execute count**: Execution count of the SQL
- **elapsed sum (sec)**: Total execution time (secs) of the SQL
- **elapsed avg (sec)**: Average SQL execution time (sec), which is the result of **elapsed sum** / **execute count**
- **elapsed max (sec)**: Maximum execution time (sec) of the SQL
- **elapsed wait (sec)**: Total waiting time (sec) of the SQL
- **instance**: Instance name
- **physical reads**: Number of times the SQL reads data from the disk
- **session logical reads**: Number of times the SQL reads data from memory
- **cpu time**: CPU time used while the SQL is running
- **redo size**: Size of the transaction log (redo log) generated by the SQL

 **SQL Statistics** function creates statistics based on the SQL that excludes literal values.

Basic screen guide



1. In **1 Time**, select the time and date to view the data. You can select a lookup time after selecting the green button.
2. In **1 Instance**, select a target to view the data.
3. Select .

The top 50 entries appear in the table.

- Select the clock display area to query a random date and time. If you click the date and time text area, the selectable options for the date and time appear.
- If you select each column in the table header, you can sort the list by the selected item.
- You can sort the searched results by the item selected in **Sort by**. Select a desired item and then select .
- In **View count**, you can set the number of entries to display in the table. Finish the settings and then select .
- To download the viewed data as a CSV file, select .
- You can hide the table header columns or add any desired items. You can also change the order of the columns. For more information.

Checking the query-based execution trend

You can classify and search the desired data by usage type or change the sorting order. Select the desired tab from **2**.

On the utmost left for each item of the table, select **4** ►. A chart appears that displays the number of executions every 5 minutes and the average execution time of the query statement. If you hover your mouse over each bar graph, you can see the number of that time.

Sorting criteria	Description
elapse avg (sec)	Sort by the longest average execution time
execute count	Sort by the greatest number of executions
elapse sum (sec)	Sort by the longest total execution time
elapse max (sec)	Sort by the longest maximum execution time
elapse wait	Sort by the longest waiting time for queries
physical read	Sort by the greatest number of data blocks read from disk
session logical reads	Sort by the greatest number of data blocks read from memory

Previewing the query

sql	elapse avg (sec)	execute count ↓	elapse sum (sec)
select /* WhaTap6D#4 */ processlist_id,processlist_user,processlist_host,proces...	<pre>select /* WhaTap6D\$4 */ processlist_id,processlist_user,processlist_host,processlist_db,processlist_command,processlist_state,thread_id,processlist_time,type,name,connection_type,thread_os_id,i.info from performance_schema.threads p left outer join information_schema.processlist i on p.processlist_id=i.id where type<>'S' and processlist_command<>'S'</pre>		

You can preview the query statement by moving the mouse pointer to the [sql](#) column in the search results list.

❗ If the [sql](#) column is selected from the table list, the **SQL details** window appears.

Filtering the searched results

Adding the filter conditions

You can filter the results by SQL text, agent name, or DB user name.

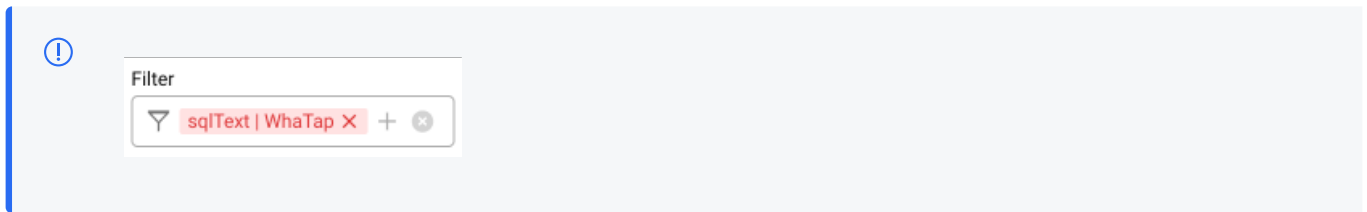
1. In the **Filter** option, click **+**.
2. In **Filter key**, select a desired filtering criteria.
 - If the value of the selected item is text, you can select any of **Includes** (blue) and **Excludes** (red).

example

- If the value of the selected item is number, you can select any condition of **==** (equal to), **>=** (greater than or equal to), and **<=** (less than or equal to).
3. In **Condition**, select a condition.
 4. Enter a string or number to match the condition.
 5. Select **Apply**.

❗ Adding and Deleting Filter Conditions

- To add a filtering condition, click the **Add** button and repeat steps 1~5. Added conditions are applied using the AND (&&) operator.
- To delete specific conditions while adding, click the  button on the right side of the filter condition. To delete all conditions, click the  **Delete All** button.
- To quickly remove all applied conditions from the **Filter** option, click the **×** button.
- The default setting is to exclude if `sqlText` contains 'WhaTap'.



Modifying the filter conditions

1. Click the applied item in the **Filter** option at the top of the screen.
2. In the **Edit filter** window, edit the desired item and click the **Apply** button.

Log

WhaTap's log monitoring has improved user convenience and accessibility based on the establishment of an integrated system. WhaTap supports the efficiency of the parser as well as Live Tail, explorer, searcher, and event notification that users mainly use through reliable data collection based on its own technology.

The **Log** menu provides functions such as lookup, analysis, and options setting. Through the **Event Configuration** menu, you can set the log-related event notifications.

- Home > Select Project > **Log**

Select a project on the initial screen of the WhaTap monitoring service, and then select **Log** under the project menu. By using the following functions, you can access complex logs more easily and then check and analyze them in real time under various conditions.

- **Live Tail**
- **Log Trend**
- **Log Search**

- Home > Select Project > **Alert** > **Event Configuration**

Select a project on the initial screen of the WhaTap monitoring service, and then select **Alert** under the Project menu to enter the **Event Configuration** menu. You can set event conditions and receive alerts through various routes such as email, SMS, messenger, and App Push.

The guide to the main menus of WhaTap log monitoring service is provided as follows.



Application

It guides you how to collect logs from the database and AWS RDS.

 Live Tail

It guides you to Live Tail of the log monitoring.

 Log Explorer

Guides you through the Log Explorer menu in Log Monitoring.

 Log Trend

It guides you to the Log Trend menu of Log Monitoring.

 Log Search

It guides you to the Log Search menu of Log Monitoring.

 Log setting

It explains how to set the log monitoring.

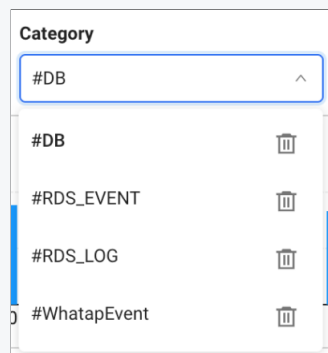
 Parsing the logs

The following guides you to the log parser provided by WhaTap.

Application

WhaTap provides the database log monitoring service. You can collect logs from server-installed databases and AWS RDS that is a cloud service. Through this, you can check the scattered logs in the database at a glance in **Live Tail** and **Log Trend** for log monitoring.

- ✓ After log monitoring has been applied, installed DB is displayed as `#DB`, AWS RDS as `#RDS_LOG`, and `#RDS_EVENT` in **Category** of each menu and the log list.



Applying to the server-installed DB

To monitor logs in server-installed databases, apply the following options in `xos.conf` file.

`xos.conf`

```
file=/home/oracle/diag/rdbms/ora18/ORA18/trace/alert_ORA18.log
```

- ⓘ **Oracle Pro** products come with the built-in XOS agent, eliminating the need for a separate installation process.

Starting the log monitoring

After configuring all settings, go to **Log > Log Configuration** to activate the log monitoring.

Set up the agent and enable log monitoring

You can easily check the log of the application with the Live Tail and Log Trend.

How to set up log monitoring

[Installable DB](#)

AWS RDS

1. Install the XOS agent

In the case of ORACLE PRO, xos is built-in, so xos installation process is not required.

2. Apply options to xos.conf file

Apply the following options to the xos.conf file.

```
# e.g.  
file=/home/oracle/diag/rdbms/ora18/ORA18/trace/alert_ORA18.log
```

3. Activate log monitoring

Click the toggle button to activate log monitoring after setting is complete.



Live Tail

! Without the **Log Read** role, you cannot enter the menu.

Home > Select Project > **Log** > **Live Tail**

In the **Live Tail** menu, you can directly check the log data stream on the monitoring screen without connecting to the server console. Even with large volumes of complex logs, you can filter and quickly identify them using the **Filter** and **Highlight** features. The log data retrieval cycle is 2 seconds.

The major terms are as follows:

- **Category:** Unit for log collection and inquiry
- **Content:** Log messages
- **Search Key:** It is generated through the log parser settings.
- **Tag:** Key to search the collected logs

! You can change the column width by dragging an edge of the log table column.

Agent option

If the agent option is set, the log levels are collected and the colors for each log level appear as follows:

2023-12-18 14:31:02.563	[level]	INFO	[pcode]	2277	[agenttime]	1702877462042	[oid]	778873916	[category]	AppLog	[loggerName]	io.home.test.logback02starter.base.web.LogbackControllerGreeting
INFO log in our greeting method.												
2023-12-18 14:31:02.563	[level]	WARN	[pcode]	2277	[agenttime]	1702877462043	[oid]	778873916	[category]	AppLog	[loggerName]	io.home.test.logback02starter.base.web.LogbackControllerError
io.home.test.logback02starter.base.errors.exception.ApiException: [2204] Process failure. Please try again.												
2023-12-18 14:31:02.586	[level]	error	[pcode]	2277	[agenttime]	1702877462043	[oid]	778873916	[category]	AppServer	[_event_status_]	error
Servlet.service() for servlet [dispatcherServlet] in context with path [] threw exception [Request processing failed: io.home.test.logback02starter.base.error]												
2023-12-18 14:31:02.586	[level]	error	[pcode]	2277	[agenttime]	1702877462057	[oid]	778873916	[category]	AppServer	[_event_status_]	error
Servlet.service() for servlet [dispatcherServlet] in context with path [] threw exception [Request processing failed: io.home.test.logback02starter.base.error]												
2023-12-18 14:31:02.586	[level]	error	[pcode]	2277	[agenttime]	1702877462081	[oid]	778873916	[category]	AppServer	[_event_status_]	error
Servlet.service() for servlet [dispatcherServlet] in context with path [] threw exception [Request processing failed: io.home.test.logback02starter.base.error]												
2023-12-18 14:31:02.586	[level]	error	[pcode]	2277	[agenttime]	1702877462087	[oid]	778873916	[category]	AppServer	[_event_status_]	error
Servlet.service() for servlet [dispatcherServlet] in context with path [] threw exception [Request processing failed: io.home.test.logback02starter.base.error]												

! Setting the agent options



- The agent options are as follows:

```
# whatap.conf
weaving=log4j-2.17
weaving=logback-1.2.8
```

- In Java agent 2.2.22 or later, it can be used when setting log4j-2.17 or logback-1.2.8 in the weaving setting. It is required to restart the agent.
- The log level is determined based on the `level` and `type` among the parsed keywords. When there are keys parsed as `level` and `type`, and the parsed values have **FATAL**, **CRITICAL**, **ERROR**, **WARN**, **WARNING**, and **INFO**, the log level color appears.

1 Filter area

Apply Filter

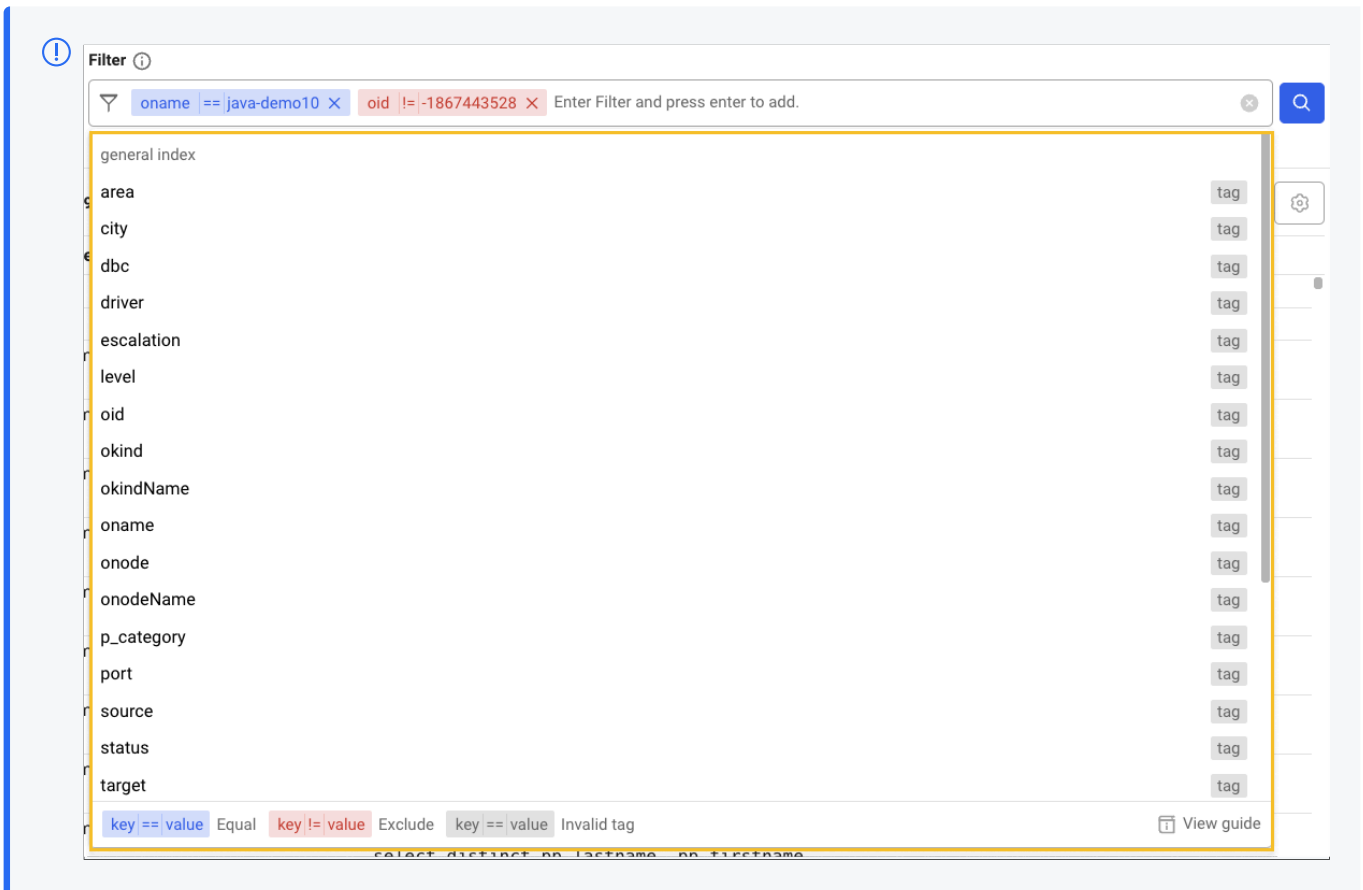
When a filter is applied, the logs that match the specified conditions are filtered. You can enter multiple filters. OR (`||`) if the filters have the same tags, otherwise AND (`&&`).

You can specify a filter by directly entering a value in the input field or clicking the **Filter** field. Filter tags are entered in the following order: `Search key`, `Operator`, and `Search value`. If you select  **Search**, the data with the filter applied can be viewed log list area.



Guide UI

The **Guide UI** is provided under the input field as follows: You can close the **Guide UI** via the ESC key when your mouse cursor is over the input field or filter tag.



Entry of the search key, operator, and search value

- Upon entry of the **search key**, recommended values are provided by distinguishing for general index, reserved word index, and numeric index.
- When entering an **operator**, for a general index search key, **==** and **!=** options are provided at the bottom. In case of the numeric index, the **>**, **<**, **<=**, **>=**, **==**, and **!=** options are provided.
- Upon entry of the **search value** for matched search (**>**, **<**, **<=**, **>=**, **==**), it is highlighted in **blue**. In case of excluded search (**!=**), it is highlighted in **red**. Upon entry of the **search value**, you can search using the case-sensitive option.

- If the filter tags exceed 2 lines, you can fold it by selecting the **^ Fold** icon.
- After entering a filter tag, you can use the Shift and Tab keys simultaneously in the input field to go to the previous filter tag.



- After entering a filter tag, you can use the Tab key in the input field to go to the next filter tag.

Adding filter tags

To add a filter tag, type the desired text in the input field and press **Enter** or **Tab**.

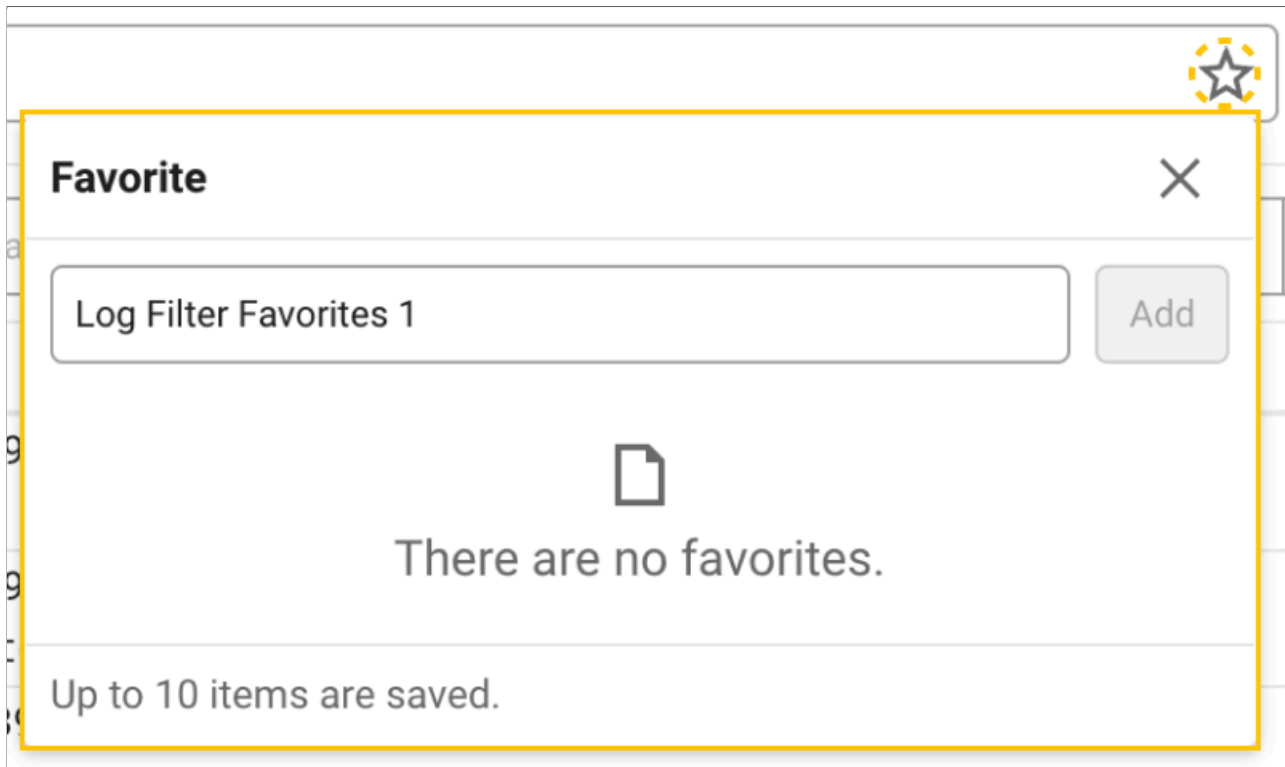
You can also select a recommended value from the **Guide UI** below the input field by clicking it or navigating with the arrow keys.

Deleting filter tags

To remove a single tag, press **Backspace** or click the **X** icon on the tag. To clear all tags at once, click the **X** icon located on the right side of the input field.

Filter favorites

The filter **favorites** feature is provided. You can select the ☆ **Favorites** icon on the right of the input window to **add**, **delete**, and select desired filter search conditions in the favorites. **Favorites** can store up to 50 cases.



Exceptional cases for application of filters

- In the tag with a numeric index (`search key` that ends with `.n`), only numbers can be input for the `search value` .
- Duplicate 'search key' and 'search value' cannot be entered.
- Searching is not possible if there is a tag whose `search key` or `search value` is missing. Invalid tags appear in gray.

ⓘ You cannot enter a `category` with the **Live Tail** `search key` .
The expression under the entered filter value is a preview of the filter expression for querying the log data.

Applying non-parsing keyword filters

You can search the logs containing keywords that have not been parsed, or for which no index has been created.

In this case, all logs in the specified range are fully scanned. Therefore, the search speed may be slower compared to the indexed key. For standardized log data, it is recommended to search using the index key through **Log parser setting**.

Filter ⓘ

🔍
oid != -1009464250 ✕
okind == -398596773 ✕
content == *select* ✕
Enter Filter and press enter to add.
✕

oid != -1009464250 && okind == -398596773 && content == *select*

1. Select **Category**.
2. In the **Filter** field, enter a desired keyword for a search after leaving spaces on the **content** basis.
e.g. `content *select*`
3. Click 🔍 **Search** to view the logs.

- ⓘ • In case of **Live Tail**, all logs can be viewed. Thus, the category is not required to be specified.
- For more information about the parser configuration, see [the following](#).

Modifying the filter

Enter a value in the filter and then click the specified value to modify it.



- You can modify the filter by entering text again in the input field.
- You can modify the recommended values by selecting them in the **Guide UI** below the input field.

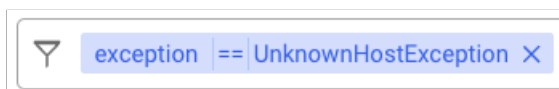
Search Key

In the following image, the blue boxed part is the search key for parsing. In **Log Configuration**, you can register and set the parsing logic in **Log parser setting** tab with the search key.

Search key

Filter input syntax

A tag consists of a search key and a search value. In the following example, the search key is `exception` and the search value is `UnknownHostException`. The example searches for the log data that includes an exception (`UnknownHostException`) that is generated when the server cannot connect the host because the IP address and domain address do not match among the collected log data.



Search key types

Search key types	Search key format	Meaning	Example for Search key and search value	Search example
String keyword	keyword	File Name	- Key: fileName - Value: /data/whatap/logs/yard.log	fileName:/data/whatap/logs/yard.log
Numeric keyword	keyword.n	Response Time	- Key: response_time.n - Value: 2945	response_time.n>=2945
Reserved keyword (predefined keyword)	@keyword	Transaction ID	- Key: @txid - Value: 85459614215434144	-
Log body's keyword	content	Log body	- Key: content - Value: user input	content: *ERROR*

ⓘ Content search key

- The content search key searches the bodies of unindexed logs. For example, if you enter `content *ERROR*`, the logs containing `ERROR` are searched among the log bodies.
- If you do not know which keywords need to be used for indexing, use the content search keys to identify the logs containing problematic keywords. Afterwards, you can improve the search performance by setting a parser with the keyword through the log parser settings in **Log Configuration** and creating indexes.

Common syntaxes

Syntax type	Description	Example
<code>==searchValue</code>	Searches for the logs that match the search value.	<code>exception==RuntimeExceptionexception</code>

Syntax type	Description	Example
<code>!=searchValue</code>	Searches for the logs that exclude the search value.	<code>exception!=RuntimeException</code>
<code>*searchValue</code>	Searches for the logs ending with the search value.	<code>word==*hello</code>
<code>searchValue*</code>	Searches for the logs starting with the search value.	<code>word==hello*</code>
<code>*searchValue*</code>	Searches for the logs that contain the search value in the middle.	<code>word==*hello*</code>
<code>*search*Value*</code>	Searches for the logs that contain the search value.	<code>word==*he*llo*</code>
<code>re:{regex}</code>	Searches for the logs that match a regular expression.	<code>caller==re:^(i\.w\.a\.w\.s\.v\.r\.</code>
<code>**</code>	Searches all logs that match the search key.	

Syntax in case the search key has a numeric value (keyword.n)

The following syntax is supported only when the search key is in `keyword.n` format.

- Search values must only be numbers.
- No prefix is attached to the value of the `.n` keyword. All keywords other than `.n` must be attached with a prefix.

For example, `+>searchValue` is invalid.

Syntax type	Description	Example
<code>>searchValue</code>	Searches for the logs containing values greater than the search value.	<code>response_time.n>3000</code>

Syntax type	Description	Example
<code>>=searchValue</code>	Searches for the logs containing values greater than or equal to the search value.	<code>response_time.n>=3000</code>
<code>==searchValue</code>	Searches for the logs containing the same search value.	<code>response_time.n==3000</code>
<code>!=searchValue</code>	Searches for the logs that contain values different from the search value.	<code>response_time.n!=3000</code>
<code><searchValue</code>	Searches for the logs containing values less than the search value.	<code>response_time.n<3000</code>
<code><=searchValue</code>	Searches for the logs containing values less than or equal to the search value.	<code>response_time.n<=3000</code>

Viewing personal information

If you have the **log personal information query** role, you can display de-identified personal information by enabling the **Display personal information**  toggle button.

- For more information about the **log personal information query** role, see [the following](#).
- For more information about **De-identification of personal information**, see [the following](#).

2 Content highlight area

The highlight function allows you to easily identify desired keywords in log content. You can filter with single or multiple keywords.

- Enter the keyword you want to highlight in the keyword input field.
 - Example: `select`
- The searched keyword will be highlighted in the Content of the log list.
 - If you click the  **Log full screen** icon, you can view **Log** and **Timestamp** in fullscreen.
 - You can navigate between highlighted keywords using shortcuts.
 - `Enter` : Move to the next row
 - `Shift+Enter` : Move to the previous row
 - `^` / `↓` icons: Move between rows
 - `Cmd + F` (Mac) / `Ctrl + F` (Windows): Search keywords

Multi-keyword condition

When highlighting with multiple keywords, write as follows:

String to input	Description	Result
<code>a b c</code>	Separate each keyword with a space.	a, b, c
<code>"Whatap is good."</code>	To include spaces in keywords, wrap them in <code>'</code> or <code>"</code> .	Whatap is good.
<code>"Whatap\\ is good."</code>	If <code>\</code> is included in the keyword surrounded by <code>"</code> , enter it as <code>\\</code> .	Whatap\ is good.

Setting the highlight color

By selecting  **Highlight**, you can set the keyword and its color for highlighting.

- In addition, enter the keyword to set the color for.
- If you click **Color** on the left of the input field, available colors appear.
- Basically, different highlighting by the level (**WARN**, **ERROR**, **FATAL**) is applied.
- The settings are saved in **Project Rules**.

Table settings

Column settings

Right of the  area, select  **Column settings**. You can add columns or set order.

- **Adding columns**

You can add columns to the table by selecting tags. If you unselect the **log** column, you cannot see **Log view detail setting**. Be sure to select at least one column.

- **Setting the column order**

When you add a column, the column is added to **Display columns**. Drag a desired column to change the order of the columns.

Log view detail setting

On the right of , select  **Log view detail setting**. **content** and **Tag** are all checked and two items are also displayed. Be sure to select any of **content** and **Tag**.

Unchecked items are not displayed in the table. If you deselect the **Tag** as follows, the log's **Tag** does not appear in the table.

Log table tag invisible

When you add tags in the **Tag management** list, the tags in the log are listed in the order you added. You can change the order of the tags by dragging them. If you disable an added tag, the disabled tag does not appear.

- **Column settings** and **Log view detail setting** can be used in **Live Tail**, **Log Search**, and **Log Trend**.
- In the same project, the **Live Tail**, **Log Search**, and **Log Trend** menus share **Column settings** and **Log view detail setting**.

Log Explorer

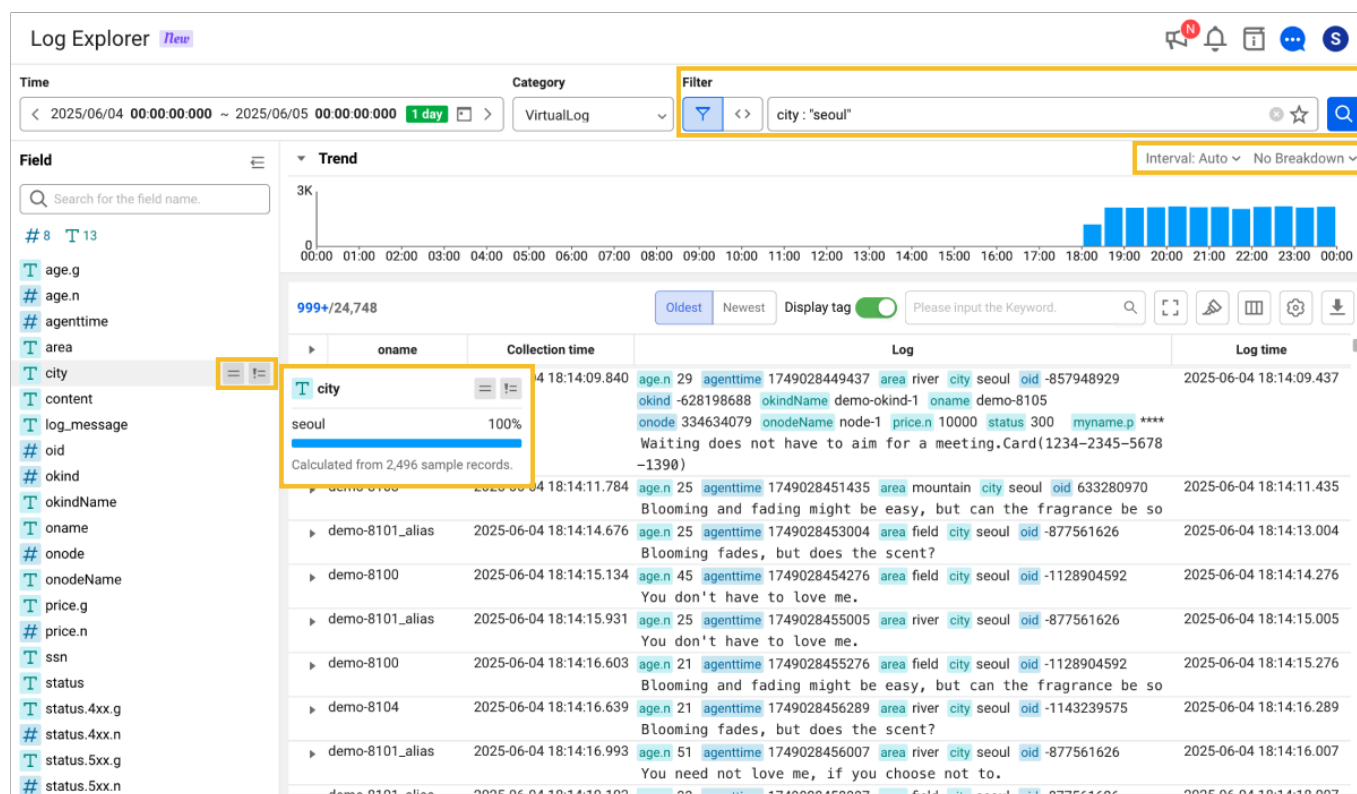
ⓘ Without the **Log Read** role, you cannot enter the menu.

Home screen > Select Project > **Log** > **Log Explorer** *New*

The **Log Explorer** feature allows you to quickly search, filter, and analyze log data. You can understand the field structure, perform query-based filtering to efficiently search for logs, and visualize the results. This feature is based on the Lucene index and offers the following key functions:

1. **Log trend analysis and search:** Enables both log trend visualization and search, helping you understand log patterns over time.
2. **Fast search and filtering:** Search for problematic logs quickly using simple keywords. Supports both WhaTap log search query and Lucene query syntax for defining various search conditions.
3. **Autocomplete functionality:** Helps users write and learn queries easily through auto-suggestions.
4. **Data visualization:** Uses stacked graphs to visualize the volume of logs over time. Drill-down capability allows deep analysis of specific logs.
5. **Favorites and sharing:** Save frequently used filters as favorites and share them easily via URL.

Basic Screen Overview



Time Range Selection

Set the time range for the log search using the **Time Selector** in the top option bar.

Category

Select the type of log you want to explore in **Select Category** on the top option bar (e.g., AppLog).

Filter

In the **Filter** on the top option bar, select the query language and enter the search conditions to apply the filter. The search query syntax supports both WhaTap Log Search Query and Lucene Query.

-  : WhaTap Log Search Query
-  : Lucene Query

Search Query Syntax

WhaTap log search query

WhaTap log search queries support autocomplete and allow quick log searches with simple keyword input. You can define various search conditions using operators like `and` , `or` , etc.

- Example: Entering `error` in the search filter will search within the log content.
 1. In the top option bar, click the filter icon and select a field from the query search box (fields are shown based on selected time and category).
 2. Choose an operator:
 - a. Default: `:` , `.*` , `and` , `or`
 - b. If the field is of type `Number` : Default + `<=` , `>=` , `<` , `>`
 - c. If the field is of type `String` : Default only
 3. Select a value (values shown based on selected time, category, and field).

Search Index Field Types

Only three field types are supported: `Number` , `String` , and `Boolean` .

- `Number` : Numeric values like long, double, float
- `String` : Full text such as email body
- `Boolean` : A value of either True or False

 For more details about WhaTap log search query syntax, refer to the [documentation](#).

Lucene

Use the standard syntax of Apache Lucene version 8.11.1. Autocomplete is not supported.

 For more information on the Lucene QueryParser Classic package, see the [official documentation](#).

Field List

Can be used for drill-down analysis.

Apply Field Info to Filter

Check field names in the **Field List** and click the  or  buttons to apply them as filter conditions.

- Hovering over a field shows the  and  buttons, which can be clicked to apply filters.
- Use the field search function to quickly find the desired field.

Trend Chart

Use **Trend Chart** to view logs in graph form with **interval** and **breakdown** applied.

- **Graph Interaction:** Zoom in using mouse drag or the  button, and click bars to view logs in that area
- **Interval:** Choose from Auto, 1s, 5s, 30s, 1m, 5m, 10m, 30m, 1h
- **Breakdown:** Visualize distribution of Top 3 values + Others for a specific field, and apply drill-down filters

Breakdown

Breakdown is a visualization feature that analyzes log count trends by a specific field.

- Displays a stacked graph separating the Top 3 values and Others for a given field
- Useful for drill-down analysis. You can click parts of the legend or stacked graph to highlight specific values or add filter conditions.

View Logs

You can view logs that match your search conditions. In the **Field List**, field data and log tags are displayed in matching highlight colors.

Content Highlight

The highlight function allows you to easily identify desired keywords in log content. You can filter with single or multiple keywords.

1. Enter the keyword you want to highlight in the keyword input field.
 - Example: `select`
2. The searched keyword will be highlighted in the Content of the log list.
 - If you click the  **Log full screen** icon, you can view **Log** and **Timestamp** in fullscreen.
 - You can navigate between highlighted keywords using shortcuts.
 - `Enter` : Move to the next row
 - `Shift+Enter` : Move to the previous row
 - `^` / `↓` icons: Move between rows
 - `Cmd + F` (Mac) / `Ctrl + F` (Windows): Search keywords

Multi-keyword condition

When highlighting with multiple keywords, write as follows:

String to input	Description	Result
<code>a b c</code>	Separate each keyword with a space.	a, b, c
<code>"Whatap is good."</code>	To include spaces in keywords, wrap them in <code>'</code> or <code>""</code> .	Whatap is good.
<code>"Whatap\\ is good."</code>	If <code>\</code> is included in the keyword surrounded by <code>""</code> , enter it as <code>\\</code> .	Whatap\ is good.

Setting the highlight color

By selecting  **Highlight**, you can set the keyword and its color for highlighting.

- In addition, enter the keyword to set the color for.
- If you click **Color** on the left of the input field, available colors appear.
- Basically, different highlighting by the level (**WARN**, **ERROR**, **FATAL**) is applied.
- The settings are saved in **Project Rules**.

Column settings

Click the  button to add columns or change their order.

• Add Columns

In **Column Settings**, select tags to add them as columns in the table.

 If no log columns are selected, **Log view detail setting** will not be visible. At least one column must be selected.

• Reorder Columns

Added columns will appear in **Display columns**. Drag them to change the display order.

Log view detail setting

Click the  button to configure log display options. By default, both **content** and **Tag** are selected and shown.

- Either **content** or **Tag** must be selected.
- Tags added in **Tag management** appear in order in the log display. You can reorder tags by dragging. Disabled tags are not shown.
- If an item is unchecked, it is not shown in the table. For example, if **Tag** is unchecked, log tags are not displayed.

log table contents remove tag sc

! Scope of Column and Log Display Settings

- **Column settings** and **Log view detail setting** can be used in **Live Tail**, **Log Search**, and **Log Trend**.
- In the same project, **Live Tail**, **Log Search**, and **Log Trend** share **Column settings** and **Log view detail setting**.

Download Log File

You can download the collected logs that match your search results in [CSV](#) or [TXT](#) file formats.

1. Click the  **Download** button at the right center of the screen.
2. Click **Download CSV** or **Download Log** to download the file in your preferred format.

Log Trend

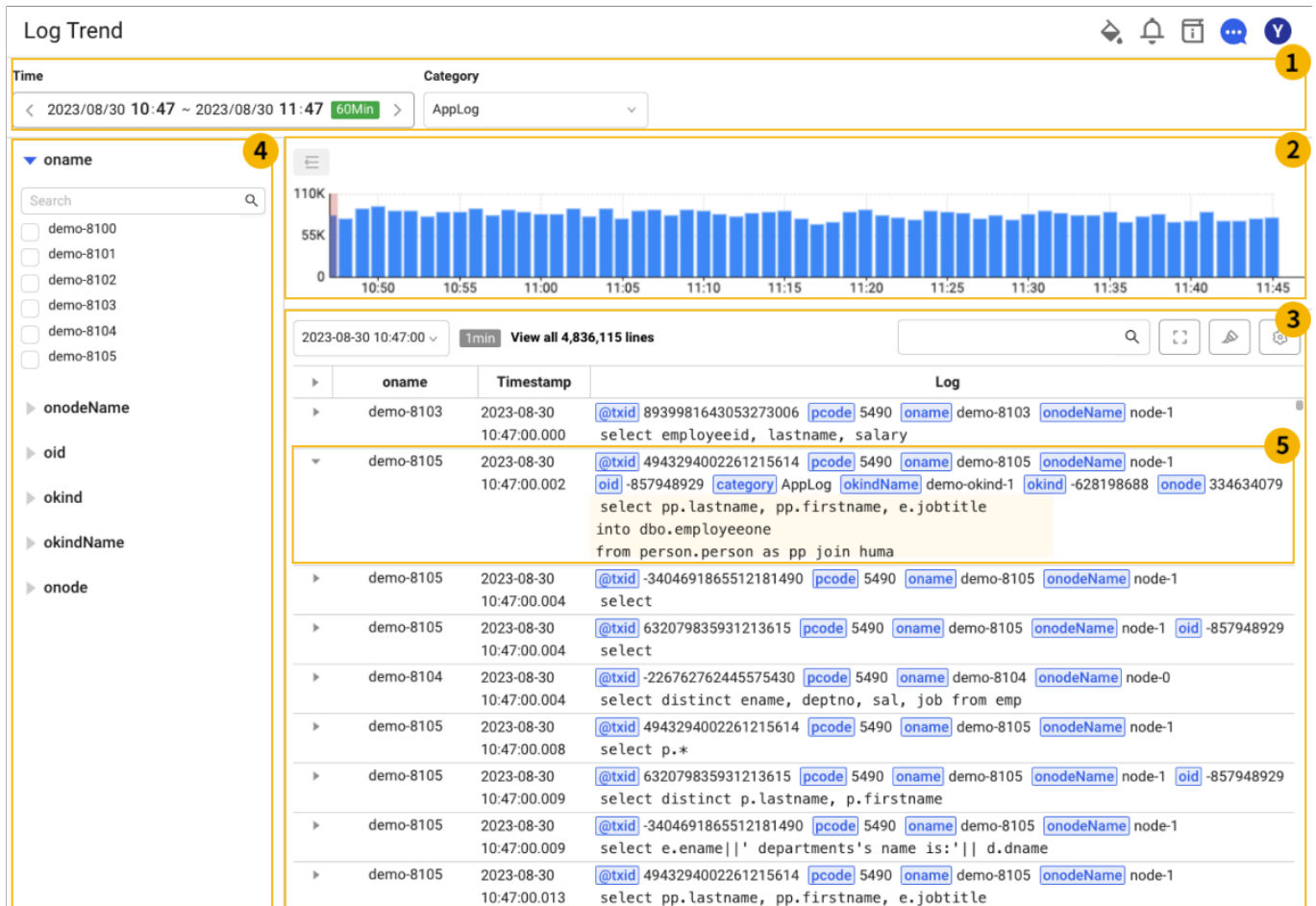
ⓘ Without the **Log Read** role, you cannot enter the menu.

Home > Select Project > **Log** > **Log Trend**

In **Log Trend**, you can check the occurrence pattern of a specific error type through the occurrence count trend of the logs classified by type and see the log data details by time. You can quickly see the desired log through the highlighting feature. You can also check the trends of the collected logs by category.

The major terms are as follows:

- **Category:** Unit for log collection and inquiry
- **Content:** Log messages
- **Search Key:** It is generated through the log parser settings.
- **Tag:** Key to search the collected logs



Request Data

- If the scroll reaches the bottom, the next data is retrieved.
- In 1, specify the **time** and collection unit, **category**, of the log data to be searched.
- If the **category** is changed, the logs that correspond to the selected category are searched. You can see the logs in 2 bar chart and 3 log table.
- If you click a bar in 2 bar chart, the logs are viewed that correspond to the time range of the bar.
- 3 You can see the total number of logs viewed from the upper left of the log table.
- On the upper right of the 3 log table, select the  **full screen** icon, you can see the **log** and **collection time** in the full screen.

- **4** In Side menu, you can see the logs filtered by tags. Up to two search keys can be selected, and multiple search values can be selected.
- If the agent option is set, the log levels are collected and the colors for each log level appear as follows:

2023-12-18 14:31:02.563	level	INFO	pcode	2277	agenttime	1702877462042	oid	778873916	category	AppLog	loggerName	io.home.test.logback02starter.base.web.LogbackControllerGreeting
INFO log in our greeting method.												
2023-12-18 14:31:02.563	level	WARN	pcode	2277	agenttime	1702877462043	oid	778873916	category	AppLog	loggerName	io.home.test.logback02starter.base.web.LogbackControllerError
io.home.test.logback02starter.base.errors.exception.ApiException: [2204] Process failure. Please try again.												
2023-12-18 14:31:02.586	level	error	pcode	2277	agenttime	1702877462043	oid	778873916	category	AppServer	event_status	error
Servlet.service() for servlet [DispatcherServlet] in context with path [/] threw exception [Request processing failed: io.home.test.logback02starter.base.error]												
2023-12-18 14:31:02.586	level	error	pcode	2277	agenttime	1702877462057	oid	778873916	category	AppServer	event_status	error
Servlet.service() for servlet [DispatcherServlet] in context with path [/] threw exception [Request processing failed: io.home.test.logback02starter.base.error]												
2023-12-18 14:31:02.586	level	error	pcode	2277	agenttime	1702877462081	oid	778873916	category	AppServer	event_status	error
Servlet.service() for servlet [DispatcherServlet] in context with path [/] threw exception [Request processing failed: io.home.test.logback02starter.base.error]												
2023-12-18 14:31:02.586	level	error	pcode	2277	agenttime	1702877462087	oid	778873916	category	AppServer	event_status	error
Servlet.service() for servlet [DispatcherServlet] in context with path [/] threw exception [Request processing failed: io.home.test.logback02starter.base.error]												

① Setting the agent options

- The agent options are as follows:

```
# whatap.conf
weaving=log4j-2.17
weaving=logback-1.2.8
```

- In Java agent 2.2.22 or later, it can be used when setting log4j-2.17 or logback-1.2.8 in the weaving setting. It is required to restart the agent.
- The log level is determined based on the `level` and `type` among the parsed keywords. When there are keys parsed as `level` and `type`, and the parsed values have **FATAL**, **CRITICAL**, **ERROR**, **WARN**, **WARNING**, and **INFO**, the log level color appears.

- ① You can change the column width by dragging an edge of the log table column.

Viewing personal information

If you have the **log personal information query** role, you can display de-identified personal information by enabling the **Display personal information**  toggle button.

- For more information about the **log personal information query** role, see [the following](#).
- For more information about **De-identification of personal information**, see [the following](#).

Checking the log content

! What is content?

Content means the log messages.

- The first line of the log column is the log's parsed key and value, and the second line is the log's content.
- **3** For each row (log) of the log table, the **More** button exists. If you select the **More** button, you can check the entire content of the log like **5**.

Configure Highlight

- The highlight function is provided to easily identify the desired keywords.
- **3** In the **input field** on the right, enter the keyword to highlight. You can check the highlighted keywords like this.

Content Highlight

The highlight function allows you to easily identify desired keywords in log content. You can filter with single or multiple keywords.

1. Enter the keyword you want to highlight in the keyword input field.
 - Example: `select`
2. The searched keyword will be highlighted in the Content of the log list.
 - If you click the **Log full screen** icon, you can view **Log** and **Timestamp** in fullscreen.
 - You can navigate between highlighted keywords using shortcuts.
 - `Enter` : Move to the next row
 - `Shift+Enter` : Move to the previous row
 - `^ / v` icons: Move between rows
 - `Cmd + F` (Mac) / `Ctrl + F` (Windows): Search keywords

Multi-keyword condition

When highlighting with multiple keywords, write as follows:

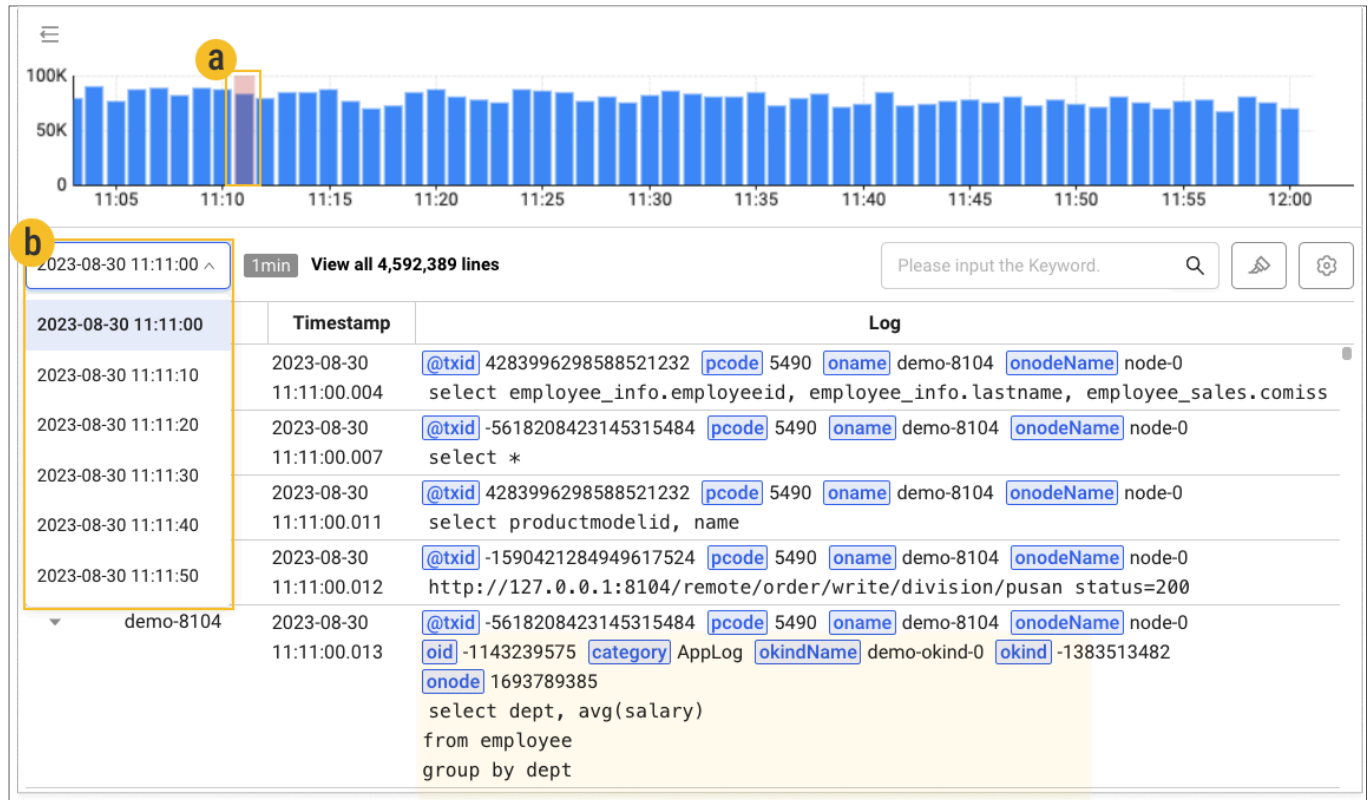
String to input	Description	Result
a b c	Separate each keyword with a space.	a, b, c
"Whatap is good."	To include spaces in keywords, wrap them in "'" or "\"".	Whatap is good.
"Whatap\\ is good."	If \ is included in the keyword surrounded by "", enter it as \\.	Whatap\ is good.

Setting the highlight color

By selecting  **Highlight**, you can set the keyword and its color for highlighting.

- In addition, enter the keyword to set the color for.
- If you click **Color** on the left of the input field, available colors appear.
- Basically, different highlighting by the level (**WARN**, **ERROR**, **FATAL**) is applied.
- The settings are saved in **Project Rules**.

Viewing the logs with a chart



- You can click or drag the desired time range bar on the bar chart (a) to view the logs for the time range.
- You can use the (b) time selection option on the upper left of the log table under the bar chart to further refine your log search in the selected time zone:
 - 1min: interval (spacing between bars on the chart)
 - Select a time option: Time zone whose time range is divided into 6 segments.

Setting the table

Column settings

Right of the (3) area, select **Column settings**. You can add columns or set order.

- **Adding columns**

You can add columns to the table by selecting tags. If you unselect the **log** column, you cannot see **Log view detail setting**. Be sure to select at least one column.

- **Setting the column order**

When you add a column, the column is added to **Display columns**. Drag a desired column to change the order of the columns.

Log view detail setting

On the right of 3, select  **Log view detail setting**. **content** and **Tag** are all checked and two items are also displayed. Be sure to select any of **content** and **Tag**.

Unchecked items are not displayed in the table. If you deselect the **Tag** as follows, the log's **Tag** does not appear in the table.

Log table tag invisible

When you add tags in the **Tag management** list, the tags in the log are listed in the order you added. You can change the order of the tags by dragging them. If you disable an added tag, the disabled tag does not appear.

- **Column settings** and **Log view detail setting** can be used in **Live Tail**, **Log Search**, and **Log Trend**.
- In the same project, the **Live Tail**, **Log Search**, and **Log Trend** menus share **Column settings** and **Log view detail setting**.

Log Search

ⓘ Without the **Log Read** role, you cannot enter the menu.

Home > Select Project > **Log** > **Log Search**

In **Log Search**, you can search a large number of integrated logs under various conditions and specify the desired logs. Multiple search conditions can be specified with parsed keys and values, so only the log data that matches the desired conditions are extracted.

The searched log data is imported to the dynamic page line by line. If the scroll reaches the bottom, the next data is automatically retrieved and displayed.

The major terms are as follows:

- **Category:** Unit for log collection and inquiry
- **Content:** Log messages
- **Search Key:** It is generated through the log parser settings.
- **Tag:** Key to search the collected logs

The screenshot displays the Log Search interface. On the left, there are filters for Time (2023/08/24 15:14 ~ 2023/08/24 16:14, 60Min), Filter, and a list of log categories: #DB (4,651), #RDS_LOG (91), and #WhatapEvent (5). The main area shows a table of logs with columns: oname, Timestamp, app, pcode, timezone, KST, log_level, LOG, pid, oid, duration, file, and oname. The table is sorted by Timestamp. A detailed log entry is expanded, showing the following information:

```

DBX-1-203-5432 2023-08-24 15:14:11.232 [app] [pcode] 878 [timezone] KST [log_level] LOG [pid] 9468 [oid] -611296874 [duration] 16132.022 [file] [oname] DBX-1-203-5432
2023-08-24 15:14:07 KST [9468]: [1-1] user=kwlee,db=db1,app=,client=[local] LOG: duration: 16132.022 ms statement: s
DBX-1-203-5432 2023-08-24 15:14:11.232 [app] [pcode] 878 [timezone] KST [log_level] LOG [pid] 9411 [oid] -611296874 [duration] 15015.888 [file] [oname] DBX-1-203-5432
2023-08-24 15:14:09 KST [9411]: [9-1] user=kwlee,db=db1,app=,client=[local] LOG: duration: 15015.888 ms statement:
DBX-1-203-5432 2023-08-24 15:14:11.232 [app] [pcode] 878 [timezone] KST [log_level] LOG [pid] 9327 [oid] -611296874
[message] process 9327 acquired ShareLock on transaction 2355689 after 15018.411 ms [file] [oname] DBX-1-203-5432 [sessin] 17-1
[client] local [category] #DB [user] kwlee [db] db1 [timestamp] 2023-08-24 15:14:09
2023-08-24 15:14:09 KST [9327]: [17-1] user=kwlee,db=db1,app=,client=[local] LOG: process 9327 acquired ShareLock on
transaction 2355689 after 15018.411 ms
DBX-1-203-5432 2023-08-24 15:14:11.232 [app] [pcode] 878 [timezone] KST [log_level] LOG [pid] 9327 [oid] -611296874 [message] while locking tuple (1,145) in relation 'kwlee'
2023-08-24 15:14:09 KST [9327]: [18-1] user=kwlee,db=db1,app=,client=[local] CONTEXT: while locking tuple (1,145) in
DBX-1-203-5432 2023-08-24 15:14:11.232 [app] [pcode] 878 [timezone] KST [log_level] LOG [pid] 9327 [oid] -611296874 [message] update kwlee set b='67890' where a=17;
2023-08-24 15:14:09 KST [9327]: [20-1] user=kwlee,db=db1,app=,client=[local] LOG: duration: 44061.823 ms statement:
DBX-1-203-5432 2023-08-24 15:14:11.232 [app] [pcode] 878 [timezone] KST [log_level] LOG [pid] 9476 [oid] -611296874
2023-08-24 15:14:09 KST [9476]: [4-1] user=kwlee,db=db1,app=,client=[local] LOG: process 9476 acquired ExclusiveLock
DBX-1-203-5432 2023-08-24 15:14:11.232 [app] [pcode] 878 [timezone] KST [log_level] LOG [pid] 9476 [oid] -611296874 [message] update kwlee set b='67890' where a=17;
2023-08-24 15:14:09 KST [9476]: [5-1] user=kwlee,db=db1,app=,client=[local] LOG: process 9474 acquired AccessExclusiv
DBX-1-203-5432 2023-08-24 15:14:11.232 [app] [pcode] 878 [timezone] KST [log_level] LOG [pid] 9474 [oid] -611296874 [message] select /* this is test */ 'a', 'b' from kwlee for update;
2023-08-24 15:14:09 KST [9474]: [5-1] user=kwlee,db=db1,app=,client=[local] LOG: process 9474 acquired AccessExclusiv
DBX-1-203-5432 2023-08-24 15:14:11.232 [app] [pcode] 878 [timezone] KST [log_level] LOG [pid] 9476 [oid] -611296874
2023-08-24 15:14:10 KST [9476]: [6-1] user=kwlee,db=db1,app=,client=[local] LOG: process 9476 still waiting for Share

```

Request Data

- If the scroll reaches the bottom, the next data is retrieved. You can view 10,000 logs at a time.
- **3** You can see the total number of logs viewed from the upper left of the log table.
- Log data can be queried in chronological order. On the upper right of the **3** log table, select a desired search method among **Oldest** and **Newest**.
- After specifying the time range, select **Apply** to finish the search time setting, and then select **Search** to search for data.
- On the upper right of the **3** log table, select the **Log full screen** icon, you can see the **Log** and **Timestamp** in the full screen.
- If the agent option is set, the log levels are collected and the colors for each log level appear as follows:

2023-12-18 14:31:02.563	level	INFO	pcode	2277	agenttime	1702877462042	oid	778873916	category	AppLog	loggerName	io.home.test.logback02starter.base.web.LogbackControllerGreeting
INFO log in our greeting method.												
2023-12-18 14:31:02.563	level	WARN	pcode	2277	agenttime	1702877462043	oid	778873916	category	AppLog	loggerName	io.home.test.logback02starter.base.web.LogbackControllerError
io.home.test.logback02starter.base.errors.exception.ApiException: [2204] Process failure. Please try again.												
2023-12-18 14:31:02.586	level	error	pcode	2277	agenttime	1702877462043	oid	778873916	category	AppServer	event_status	error
Servlet.service() for servlet [DispatcherServlet] in context with path [/] threw exception [Request processing failed: io.home.test.logback02starter.base.error]												
2023-12-18 14:31:02.586	level	error	pcode	2277	agenttime	1702877462057	oid	778873916	category	AppServer	event_status	error
Servlet.service() for servlet [DispatcherServlet] in context with path [/] threw exception [Request processing failed: io.home.test.logback02starter.base.error]												
2023-12-18 14:31:02.586	level	error	pcode	2277	agenttime	1702877462081	oid	778873916	category	AppServer	event_status	error
Servlet.service() for servlet [DispatcherServlet] in context with path [/] threw exception [Request processing failed: io.home.test.logback02starter.base.error]												
2023-12-18 14:31:02.586	level	error	pcode	2277	agenttime	1702877462087	oid	778873916	category	AppServer	event_status	error
Servlet.service() for servlet [DispatcherServlet] in context with path [/] threw exception [Request processing failed: io.home.test.logback02starter.base.error]												

ⓘ Setting the agent options

- The agent options are as follows:

```
# whatap.conf
weaving=log4j-2.17
weaving=logback-1.2.8
```

- In Java agent 2.2.22 or later, it can be used when setting log4j-2.17 or logback-1.2.8 in the weaving setting. It is required to restart the agent.
- The log level is determined based on the `level` and `type` among the parsed keywords. When there are keys parsed as `level` and `type`, and the parsed values have **FATAL**, **CRITICAL**, **ERROR**, **WARN**, **WARNING**, and **INFO**, the log level color appears.

- ⓘ You can change the column width by dragging an edge of the log table column.

Checking the log content

ⓘ What is content?

Content means the log messages.

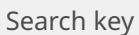
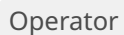
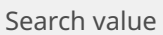
- The first line of the log column is the log's parsed key and value, and the second line is the log's content.
- 3 For each row (log) of the log table, the **More** button exists. If you select the **More** button, you can check the entire content of the log like 4.

- If you select a tag in the log, a drop-down menu appears that allows you to perform **Copy**, **Search**, **Exclusion search**, and **Adjacent Log**.

Filter

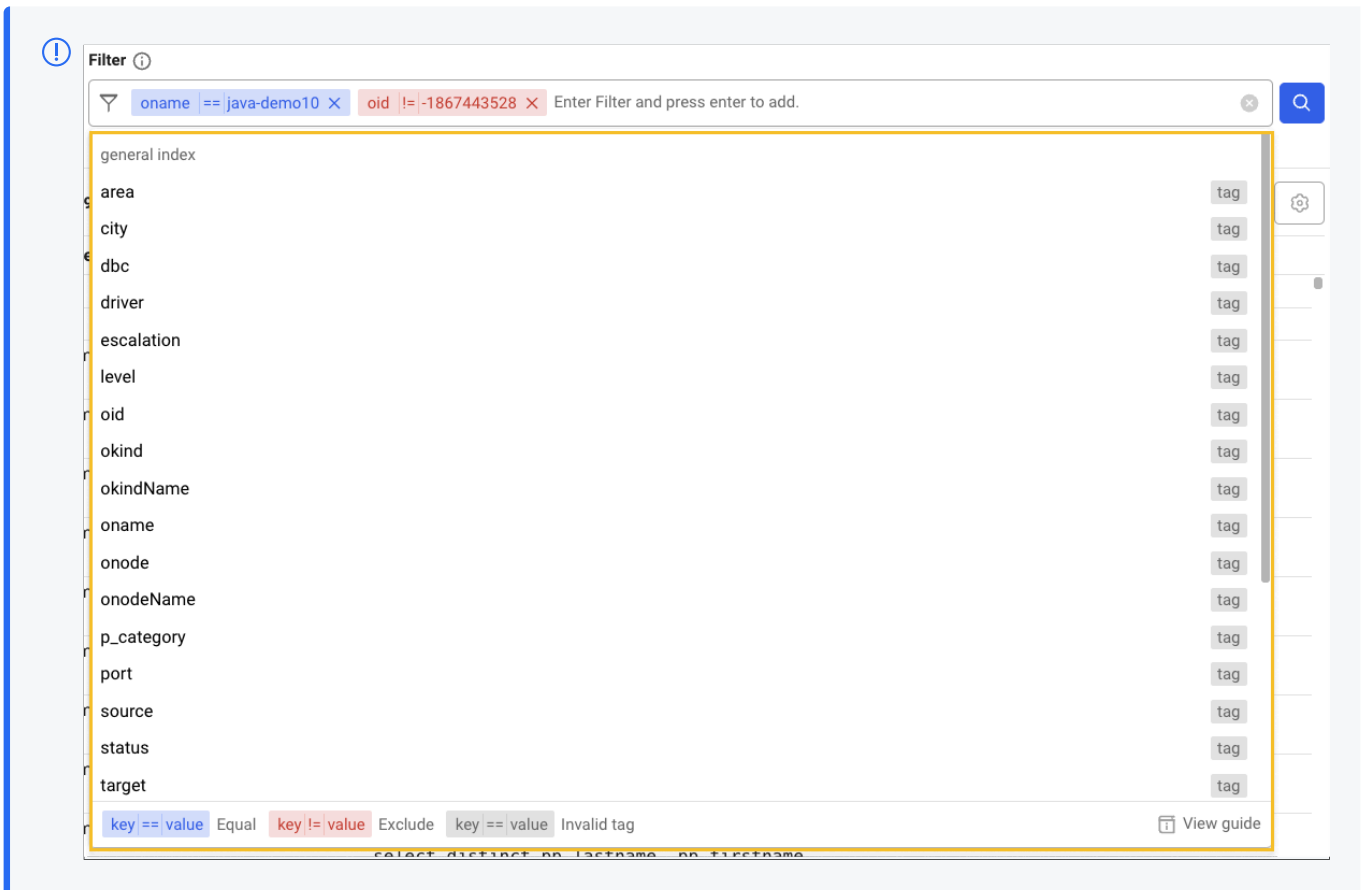
Apply Filter

In **Time selector** on the left of , you can specify the time range. On the right, When a filter is applied, the logs that match the specified conditions are filtered. You can enter multiple filters. OR () if the filters have the same tags, otherwise AND ().

You can specify a filter by directly entering a value in the input field or clicking the **Filter** field. Filter tags are entered in the following order: , , and  . If you select  **Search**, the data with the filter applied can be viewed log list area.

Guide UI

The **Guide UI** is provided under the input field as follows: You can close the **Guide UI** via the ESC key when your mouse cursor is over the input field or filter tag.



Entry of the search key, operator, and search value

- Upon entry of the **search key**, recommended values are provided by distinguishing for general index, reserved word index, and numeric index.
- When entering an **operator**, for a general index search key, **==** and **!=** options are provided at the bottom. In case of the numeric index, the **>**, **<**, **<=**, **>=**, **==**, and **!=** options are provided.
- Upon entry of the **search value** for matched search (**>**, **<**, **<=**, **>=**, **==**), it is highlighted in **blue**. In case of excluded search (**!=**), it is highlighted in **red**. Upon entry of the **search value**, you can search using the case-sensitive option.

- If the filter tags exceed 2 lines, you can fold it by selecting the **^ Fold** icon.
- After entering a filter tag, you can use the Shift and Tab keys simultaneously in the input field to go to the previous filter tag.



- After entering a filter tag, you can use the Tab key in the input field to go to the next filter tag.

Adding filter tags

To add a filter tag, type the desired text in the input field and press **Enter** or **Tab**.

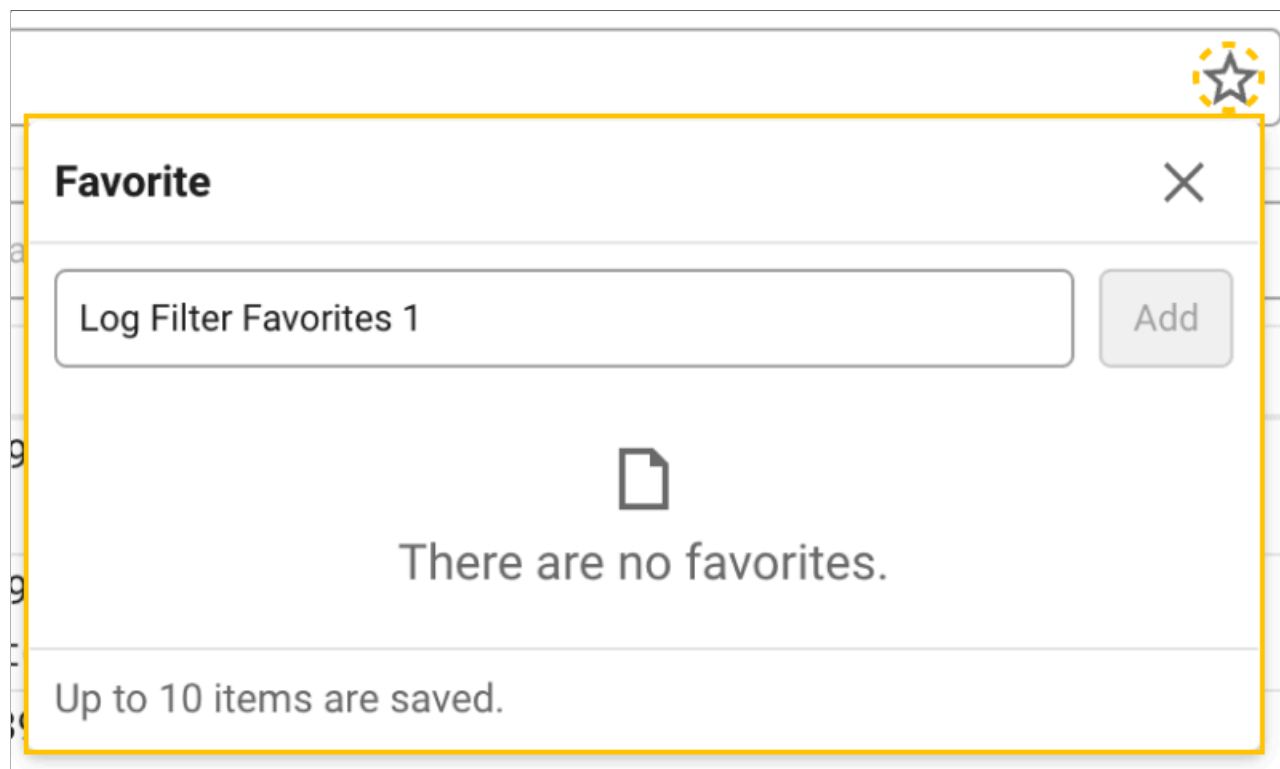
You can also select a recommended value from the **Guide UI** below the input field by clicking it or navigating with the arrow keys.

Deleting filter tags

To remove a single tag, press **Backspace** or click the **X** icon on the tag. To clear all tags at once, click the **X** icon located on the right side of the input field.

Filter favorites

The filter **favorites** feature is provided. You can select the ☆ **Favorites** icon on the right of the input window to **add**, **delete**, and select desired filter search conditions in the favorites. **Favorites** can store up to 50 cases.



Exceptional cases for application of filters

- In the tag with a numeric index (`search key` that ends with `.n`), only numbers can be input for the `search value` .
- Duplicate 'search key' and 'search value' cannot be entered.
- Searching is not possible if there is a tag whose `search key` or `search value` is missing. Invalid tags appear in gray.

Applying non-parsing keyword filters

You can search the logs containing keywords that have not been parsed, or for which no index has been created. In this case, all logs in the specified range are fully scanned. Therefore, the search speed may be slower compared to the indexed key. For standardized log data, it is recommended to search using the index key through **Log parser setting**.

Filter ⓘ

🔍 oid != -1009464250 okind == -398596773 content == *select* Enter Filter and press enter to add.

oid != -1009464250 && okind == -398596773 && content == *select*

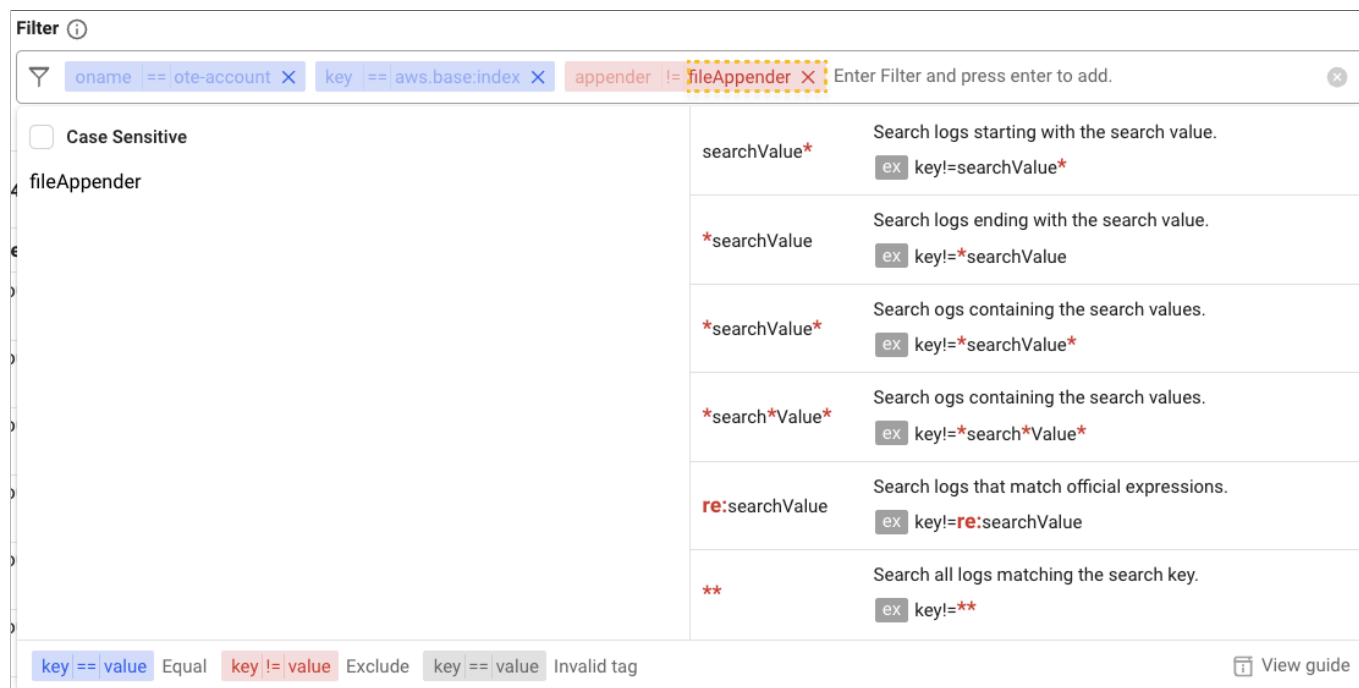
1. Select **Category**. It is required to set the category.
2. In the **Filter** field, enter a desired keyword for a search after leaving spaces on the **content** basis.
e.g. `content *select*`
3. Click **Search** to view the logs. Several logs are viewed first. You can view a maximum of 10,000 logs per each search.
4. You can make additional inquiries by scrolling down and selecting **View more** at the bottom.

The screenshot shows the Log Search interface. At the top, there's a filter bar with the same filters as the previous image. Below it, a message states: "You have viewed 10,500 of the total 36,841 results, and the results retrieved are 9999+." The main area displays a list of logs with columns for time, category, and content. A modal dialog is open, asking "Want to view more logs?" and providing instructions: "Adding a log parser enables search without additional lookups." It includes buttons for "Setting up a log parser" and "View more".

- Among all logs, 10,000 logs are viewed for a server search range. The default range for a server search is 200,000, but the ratio may differ depending on the total log volume.
- For more information about the parser configuration, see [the following](#).

Modifying the filter

Enter a value in the filter and then click the specified value to modify it.



- You can modify the filter by entering text again in the input field.
- You can modify the recommended values by selecting them in the **Guide UI** below the input field.

Search Key

The search key Indicates an identifier for accessing a desired value within the log data. The actual data for the search key is the search value. The tags on the left **2** area are the search keys parsed for each category. You can enter filters by selecting tags. The **orange** tag is the category, and the **blue** tag is the search key.

For example, **AppLog** and **AppStdOut** in **2** are categories. The tags below such as **oid** are search keys parsed. In **Log > Log Configuration**, you can register and set the parsing logic in **Log parser setting** tab with the search key.

Filter input syntax

A tag consists of a search key and a search value. In the following example, the search key is **exception** and the search value is **UnknownHostException**. The example searches for the log data that includes an exception (**UnknownHostException**) that is generated when the server cannot connect the host because the IP address and domain address do not match among the collected log data.



exception

==

UnknownHostException

×

Search key types

Search key types	Search key format	Meaning	Example for Search key and search value	Search example
String keyword	keyword	File Name	- Key: fileName - Value: /data/whatap/logs/yard.log	fileName:/data/whatap/logs/yard.log
Numeric keyword	keyword.n	Response Time	- Key: response_time.n - Value: 2945	response_time.n>=2945
Reserved keyword (predefined keyword)	@keyword	Transaction ID	- Key: @txid - Value: 85459614215434144	-

Common syntaxes

Syntax type	Description	Example
<code>==searchValue</code>	Searches for the logs that match the search value.	<code>exception==RuntimeExceptionexception</code>
<code>!=searchValue</code>	Searches for the logs that exclude the search value.	<code>exception!=RuntimeException</code>
<code>*searchValue</code>	Searches for the logs ending with the search value.	<code>word==*hello</code>
<code>searchValue*</code>	Searches for the logs starting with the search value.	<code>word==hello*</code>
<code>*searchValue*</code>	Searches for the logs that contain the search value	<code>word==*hello*</code>

Syntax type	Description	Example
	in the middle.	
*search*Value*	Searches for the logs that contain the search value.	word==*he*llo*
re:{regex}	Searches for the logs that match a regular expression.	caller==re:^(i\.w\.a\.w\.s\.v\.r\.
**	Searches all logs that match the search key.	

Syntax in case the search key has a numeric value (keyword.n)

The following syntax is supported only when the search key is in `keyword.n` format.

- Search values must only be numbers.
- No prefix is attached to the value of the `.n` keyword. All keywords other than `.n` must be attached with a prefix.

For example, `+>searchValue` is invalid.

Syntax type	Description	Example
>searchValue	Searches for the logs containing values greater than the search value.	response_time.n>3000
>=searchValue	Searches for the logs containing values greater than or equal to the search value.	response_time.n>=3000
==searchValue	Searches for the logs containing the same search value.	response_time.n==3000
!=searchValue	Searches for the logs that contain values different from the search value.	response_time.n!=3000
<searchValue	Searches for the logs containing values less than the search value.	response_time.n<3000

Syntax type	Description	Example
<code><=searchValue</code>	Searches for the logs containing values less than or equal to the search value.	<code>response_time.n<=3000</code>

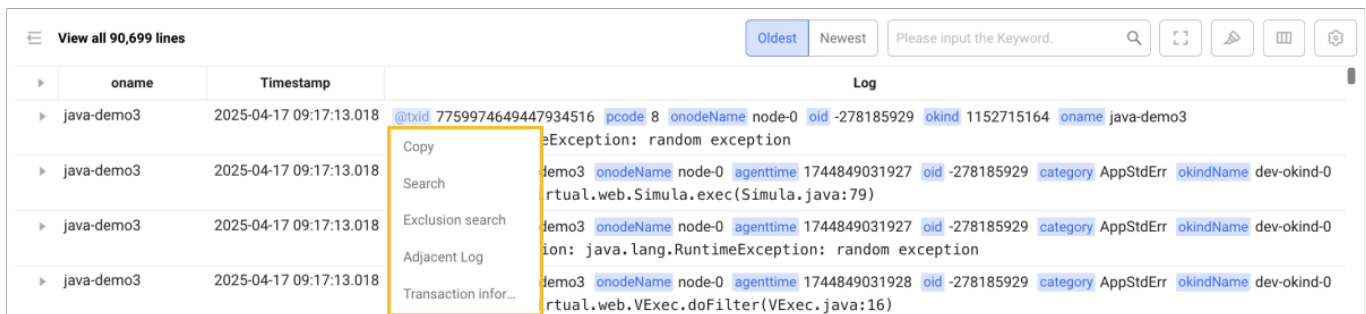
Viewing personal information

If you have the **log personal information query** role, you can display de-identified personal information by enabling the **Display personal information** ☒ toggle button.

- For more information about the **log personal information query** role, see [the following](#).
- For more information about **De-identification of personal information**, see [the following](#).

Log tagging options

When you select a log tag, the following drop-down menu appears. You can find the **Search**, **Exclusion search**, **Adjacent Log** and **Transaction information** options.



- Copy
- **Search**

If you select the **Search** option, the tags that correspond to **Filter** are entered with the inclusion (`==`) condition.

- **Exclusion search**

If you select the **Exclusion search** option, the tags that correspond to **Filter** are entered with the exclusion (`!=`) condition.

- **Adjacent Log**

If you select the **Adjacent Log** option, the **Adjacent Log** detail window appears. Search for the logs of the target server and the logs in time zones adjacent to the selected log. You can click **Time selector** to see the logs in adjacent time zones. Reference logs appear with a blue background.

- **Transaction**

If the log contains the `@txid` and `@mtid` tags, you can see the information in the transaction Information details window by clicking the transaction information option.

Content highlighting

The highlight function allows you to easily identify desired keywords in log content. You can filter with single or multiple keywords.

1. Enter the keyword you want to highlight in the keyword input field.
 - Example: `select`
2. The searched keyword will be highlighted in the Content of the log list.
 - If you click the  **Log full screen** icon, you can view **Log** and **Timestamp** in fullscreen.
 - You can navigate between highlighted keywords using shortcuts.
 - `Enter` : Move to the next row
 - `Shift+Enter` : Move to the previous row
 - `^` / `↓` icons: Move between rows
 - `Cmd + F` (Mac) / `Ctrl + F` (Windows): Search keywords

Multi-keyword condition

When highlighting with multiple keywords, write as follows:

String to input	Description	Result
<code>a b c</code>	Separate each keyword with a space.	a, b, c

String to input	Description	Result
"Whatap is good."	To include spaces in keywords, wrap them in ' ' or " ".	Whatap is good.
"Whatap\\ is good."	If \ is included in the keyword surrounded by "", enter it as \\.	Whatap\ is good.

Setting the highlight color

By selecting  **Highlight**, you can set the keyword and its color for highlighting.

- In addition, enter the keyword to set the color for.
- If you click **Color** on the left of the input field, available colors appear.
- Basically, different highlighting by the level (**WARN**, **ERROR**, **FATAL**) is applied.
- The settings are saved in **Project Rules**.

Setting the table

Column settings

Right of the  area, select  **Column settings**. You can add columns or set order.

- **Adding columns**

You can add columns to the table by selecting tags. If you unselect the **log** column, you cannot see **Log view detail setting**. Be sure to select at least one column.

- **Setting the column order**

When you add a column, the column is added to **Display columns**. Drag a desired column to change the order of the columns.

Log view detail setting

On the right of , select  **Log view detail setting**. **content** and **Tag** are all checked and two items are also displayed. Be sure to select any of **content** and **Tag**.

Unchecked items are not displayed in the table. If you deselect the **Tag** as follows, the log's **Tag** does not appear in the table.

Log table tag invisible

When you add tags in the **Tag management** list, the tags in the log are listed in the order you added. You can change the order of the tags by dragging them. If you disable an added tag, the disabled tag does not appear.

- **Column settings** and **Log view detail setting** can be used in **Live Tail**, **Log Search**, and **Log Trend**.
- In the same project, the **Live Tail**, **Log Search**, and **Log Trend** menus share **Column settings** and **Log view detail setting**.

Log Setting

Home > Select Project > **Log** > **Log Configuration**

You can configure the log monitoring related settings in **Log Configuration**. Using the tab at the top, you can use menus for checking the agent settings, determining whether to enable log monitoring, setting the log data retention period and lookup password, registering the log parser, and setting the quick indexes.

- To use the **Activate log monitoring** function, the **Edit project** role is required.
- The **Log edit** role allows you to modify the **Log Configuration** menu other than **Activate log monitoring**.

Starting the log monitoring

The screenshot shows the 'Log Configuration' interface. At the top, there are tabs: 'Getting started with log monitoring' (selected), 'Log primary parser setting', 'Log secondary parser setting', 'Fast index setting', and 'Log long-term archive statistics'. Below the tabs, there is a section titled '1 Set up the agent and enable log monitoring'. This section includes instructions on how to set up log monitoring, with sub-sections: '1. Install the XOS agent', '2. Apply Options to xos.conf File', and '3. Activate log monitoring'. The 'Activate log monitoring' section has a toggle switch that is currently turned on. Below this, there is a section titled '2 Log monitoring data setting' with a 'View guide' icon. This section includes settings for 'Log usage' (0 Lines), 'Log lookup password' (turned off), and 'Data retention' (1 Days). At the bottom, there is a table showing 'Data retention per category'.

Category	Data period	Daily log count ⓘ	Yesterday log count ⓘ	Weekly log count ⓘ	Monthly log count ⓘ	Expected log count ⓘ
WhatapEvent.whatap	1 Days	4	0	4	4	4
DB.whatap	1 Days	0	0	0	0	0

At the top, select the **Getting started with log monitoring** tab. If you select the **View guide** icon, the corresponding guide screen appears.

Set up the agent and enable log monitoring

In the **1** area, check the agent settings, and then click the **Activate log monitoring** toggle button to set whether or not to enable the log monitoring.

Check agent configuration

Check the agent version and settings to start the log monitoring. For more information about the agent settings, see [the following](#).

Activate log monitoring

Select **Activate log monitoring** to set whether or not to enable the log monitoring.

Set up the agent and enable log monitoring

You can easily check the log of the application with the Live Tail and Log Trend.

How to set up log monitoring

[Installation DB](#)[AWS RDS](#)

1. Install the XOS agent

The logs of the DB installed on the Linux file system are collected by the XOS agent on the and tabs. Please proceed with the installation through the guide link below.

[Agent Installation Guide >](#)

2. Apply Options to xos.conf File

Apply the following options to the xos.conf file.

```
# Enable log monitoring
xos.conf log_monitoring = true
# Enable log monitoring
xos.conf log_monitoring = true
# Enable log monitoring
xos.conf log_monitoring = true
```

[Copy](#)

3. Activate log monitoring

Click the toggle button to activate log monitoring after setting is complete.



- If you turn on the  toggle button, the log monitoring is enabled. You can try it for free for 15 days from the activation date.
- If you turn off the  toggle button, the log monitoring is disabled. Logs are no longer saved.

Log monitoring data setting

In the **2** area, you can see the **Log usage**. You can also change the **Log retention period** and **Log lookup password** settings.

Log retention period

This is the default data retention period to be applied commonly. If not specified, the default value is 1 day. In addition to the option to select a log retention period, you can enter a desired period of time. If not set separately in the **Log usage** list, this log data retention period is applied by default. If you set the data retention periods for each category in the **Log usage** list and select **Initialized**, the default data retention period is reset.

Log usage amount

You can specify the log data retention periods for each category in the **Log usage** list. **Log Count** means the number of logs stacked for the period. For example, **Daily log count** is the number of logs accumulated during the day, and **Expected log count** is the number of logs of the today's log count multiplied by the days of data retention.

You can specify the log data retention period as follows. You can free the space by deleting old data according to the specified period.

- **Trial Project**

You can select 1, 2, or 3 days for the data retention period.

- **Paid Project**

For the data retention period, you can select one of 1 day, 2 days, 3 days, 4 days, 5 days, 6 days, 7 days, 10 days, 30 days, and 40 days.

- **Data size-based billing**

The price differs depending on the data retention period.

For example, if an average of 2 million logs per day accumulates and the data retention period is set to 3 days, an average of 6 million logs is kept on the collection server and subject to billing.

Log lookup password

To enhance security, set the **Log lookup password**. The log lookup password is optional. If you are using a log lookup password, enter the password to go to the log screen.

ⓘ In case you forgot the password

If you have the **Edit Log** role, you can modify it with a new password in **Log Configuration**.

Log primary parser setting

Select the **Log primary parser setting** tab at the top of **Log Configuration** to register parsers for collected logs.

The log primary parser provides **GROK** and **JSON** parsers, essential for aggregating log counts by type and searching for specific logs. It extracts search keys and values that match pattern conditions, and the parsed keys are used for log type classification and search indexing.

- **GROK**: This parsing is based on the regular expression by default. It provides parsing based on the reserved keywords.
- **JSON**: Batch parsing is provided for the JSON output parts of the logs.

✅ In case of no parsing logic registered, keys for search

category , oid , oname , okind , okindName , @txid , @login , httphost

Reserved words that cannot be registered for the parser

timestamp , message , pcode , category , content , logContent

- For the following reserved words, indexes are not created even if a parser has been registered.

ⓘ For more information about the log parser, see [the following](#).

Setting item

Set value	Description	ETC
Category	Category to apply the pattern to	required
Log detection condition	Enter the search key and search value to be applied as the filter. The pattern is applied only to the log data that meets the log detection conditions. If you do not enter any log detection condition, the pattern is applied to all logs.	optional
Pattern	It is the pattern to parse the log for. Parsing is performed according to the created pattern, and indexes are created. It supports GROK, regular expression syntax.	required

Parser list

You can search for registered parsers, add, or edit them. Parsers are evaluated in registered order, and only the first matching parser is applied.

- If you select **+ Add** on the upper right, the **Parser Add** window appears.
- You can change the order of parser settings by dragging the  icon in the **Priority** column of the parser list.
- Through the parser list's  **Enable** toggle button, you can set whether or not to enable the parser.
- Through the parser list's  **Edit** and  **Delete** icons, you can modify or delete the registered parser.

Parser registration order

The following shows the common parser registration procedure.

1. In **Log primary parser setting**, select **+ Add**.
2. In the **Parser Add** window, select a **Parser**.

 Parser and pattern registration method

- ❗ • [GROK Parser and Pattern Registration](#)
- [JSON Parser and Pattern Registration](#)

3. When you select a parser, the **Register pattern** button is enabled. After clicking **Register pattern**, enter the pattern and log in the **Parser Simulation** window.

a. Enter the **Pattern**.

b. Enter the **Log Category** and **simulation parsing log**. Click the **Log Search** button, then click the **Select** button to the right of the desired log in the search results to auto-fill the log.

- ❗ Through **Log Search**, you can select a sample log from collected logs. The selected log is automatically reflected in the simulation input field, allowing you to test immediately without copying and pasting.

c. To check whether the pattern is normal, click **Simulation** and then measure the performance of simulation and pattern.

- ❗ For more information on simulation and performance measurement, see [the following](#).

4. After confirming the simulation result parses successfully, click **Apply Pattern**.

5. Click **Add** to add the parser.

❗ Personal information de-identification

If the search key name is specified as `search-key.p` during parser registration, de-identification is applied by default (e.g. `myname.p **`). WhaTap recommends managing de-identification in the **De-identification of personal information** tab. For more information, see [Personal information de-identification](#).

GROK parser pattern registration

The default syntax is `%{SYNTAX:SEMANTIC}`. For more information about the GROK parser, see [the following](#). Pattern registration and simulation are required.

- **SYNTAX**

GROK definition pattern.

- **SEMANTIC**

It is the key allocated to the parsed data.

! It is recommended to use combination words in SEMANTIC so that reserved words are not used.

Registering the JSON format parser pattern

If all or part of a log is output in JSON format, you can parse the JSON output through the JSON format parser. To detect the JSON output of the log, the **prefix** and **postfix** options are combined to specify which part of the log to be recognized by JSON for parsing. For more information about the JSON parser, see [the following](#). Simulation is required.

Option	Description
Prefix	Specify the string before the beginning of JSON string. If not specified, it is identified as a JSON string from the beginning of the log output.
Postfix	Specify the string after the end of JSON string. If not specified, it is identified as a JSON string up to the end of the log output.
Ignore	Specify fields in the JSON output to exclude from key extraction.

- **Registration example**

Log

```
[2022-10-25 10:15:34:145]...(line feed)
```



```
Request : {"key1":"value1","key2":"value2",...}(line feed)
Response : {"key3":"value3","key4":"value4",...}
```

As in the example, to parse both Request JSON and Response JSON for incoming logs, register the following two patterns.

- Pattern for request parsing

Strings between "Request : " and "Response" `{"key1":"value1","key2":"value2",...}`

- Pattern for response parsing

Strings from "Response : " to the end of a log `{"key3":"value3","key4":"value4",...}`

- **JSON custom pattern registration**

If part of a log is output in JSON format, the JSON output can be parsed by a dedicated custom parser. Enter the pattern as follows:

```
io.whatap.logsink.parser.JsonFormatParser{}
```

To detect the JSON output of the log, the **prefix** and **postfix** options are combined to specify which part of the log to be recognized by JSON for parsing.

Specify an option in `{}` of `JsonFormatParser{}` .

- **Registration example**

Log

```
[2022-10-25 10:15:34:145]...(line feed)
Request : {"key1":"value1","key2":"value2",...}(line feed)
Response : {"key3":"value3","key4":"value4",...}
```

As in the example, to parse both Request JSON and Response JSON for incoming logs, register the following two patterns.

- Pattern for request parsing

Strings between "Request : " and "Response" `{"key1":"value1","key2":"value2",...}`

```
io.whatap.logsink.parser.JsonFormatParser {prefix:"Request : ",postfix:"Response"}
```

■ Pattern for response parsing

Strings from "Response : " to the end of a log `{"key3":"value3","key4":"value4",...}`

```
io.whatap.logsink.parser.JsonFormatParser {prefix: "Response : "}
```

Parser simulation and performance measurement

After the **parser simulation**, you can register a pattern. **Performance measurement** measures the time taken by the parser to perform repeated parsing on the target string for simulation.

1. Enter values for **Pattern** and **Log**. Click the **Log Search** button, then click the **Select** button to the right of the desired log in the search results to auto-fill the log.
2. Click **Simulation** to check whether the parsing has been successful with the pattern to register.
3. If the simulation is successful, you can view **Simulation result** and **Performance measurement results**.
4. When you click **Apply pattern** after simulation, the pattern that has been entered for the selected parser is applied.

Parsing success

If a key is generated by registering a parsing logic, the value parsed with the key is added upon log inquiry. As in the following **Live Tail** menu's example, the parsed key and value are added.

	Timestamp	Log
▶	2022-08-17 14:28:00.612	oname dev949400-8093 onodeName node-1 oid 413390913 category AppStdOut okindName dev-okind-1 load 52

The parsed key can be checked in **Live Tail**, **Log Search**, and **Log Trend**.

Log secondary parser setting

Select the **Log secondary parser setting** tab at the top of **Log Configuration** to register log parsers.

The log secondary parser processes the values extracted by the primary parser (GROK or JSON) to generate statistics data or new fields. In addition to the HTTP status code-based statistics parsers, it provides various processors such as keyword filters, field processing, response time classification, and GeoIP conversion.

! The log secondary parser provides the special purpose secondary parsing function for the primary parsed results. To use the secondary parser, the **primary parser** must have been **registered**.

Parser list

Priority	Parser	Category	Log detection condition	Pattern	Enable
0	4xx, 5xx status code parser	AppStdErr	All logs		☒
1	Status code success rate parser	AppLog	All logs		☒

You can search for registered parsers, add, or edit them.

Adding parsers, changing order, enabling/disabling toggles, editing, and deleting work the same as in the [primary parser list](#).

Parser registration order

The following shows the common parser registration procedure.

× Parser Add

Parser*

WhaTap reserves the right to disable parsers that may affect the stability of the WhaTap service.

Please select the Parser. ▼

Status codes to exclude

Please select the Parser.

Category*

If multiple parsers are registered in the same category, only the first matching parser is applied.

Please select the Category.

Log detection condition

If you don't enter any filter, or if you enter only one of the search keys/values, the pattern is applied to all logs.

Search key

Search value

Add

1. If you select **+ Add**, the **Parser Add** window appears.
2. In the **Parser** selection window, select a parser. For more information about each parser's setting items, see the following.
 - **4xx, 5xx status code parser** [Setting item](#)
 - **Status code success rate parser** [Setting item](#)
3. Enter the **Status codes to exclude**.
4. In the **Category** selection window, select a category or enter it.
5. Select **Log detection condition** or enter it.
6. Select **Add** to register a parser.

4xx, 5xx status code parser setting item

The **4xx, 5xx status code parser** generates 4xx and 5xx count data based on the status values parsed by the primary parser. To exclude specific status codes, enter the status codes to exclude.

Setting item

Set value	Description	ETC
Category	It is the category to generate the 4xx, 5xx count data.	required
Log detection condition	Enter the search key and search value to be applied as the filter. The 4xx, 5xx count data is generated only for the log data that meets the log detection condition. If you do not enter any log detection condition, the data is generated for all logs.	optional
Status codes to exclude	The status codes to exclude upon generation of statistics data. If no entry, the 4xx, 5xx count data is generated for all error status codes that correspond between 4xx and 5xx.	optional

Registration example for the status parser

Log primary parser setting You can register parsers for the collected logs. Parsers are applied in the order they are applied, with only the first match being applied. + Add Save						
Priority	Parser	Category	Log detection condition	Pattern	Enable	
0	JSON	AppLog	status:404	status:404	☐	✎ ✖
1	JSON	AppLog	status:500	status:500	☒	✎ ✖
2	GROK	AppLog	All logs	%{NUMBER:status}	☒	✎ ✖

If the incoming log is `{"msg":"message","status":404}` and its status is parsed by the GROK parser as shown in the example, it is parsed as `status: 404`. If you confirm that the status has been normally parsed, register the status codes to exclude by the 4xx, 5xx status code parser.

Data Search

If all parsers have been registered, go to **Integrated Flex Board** and then create the **Log 4XX, 5XX count** widget.

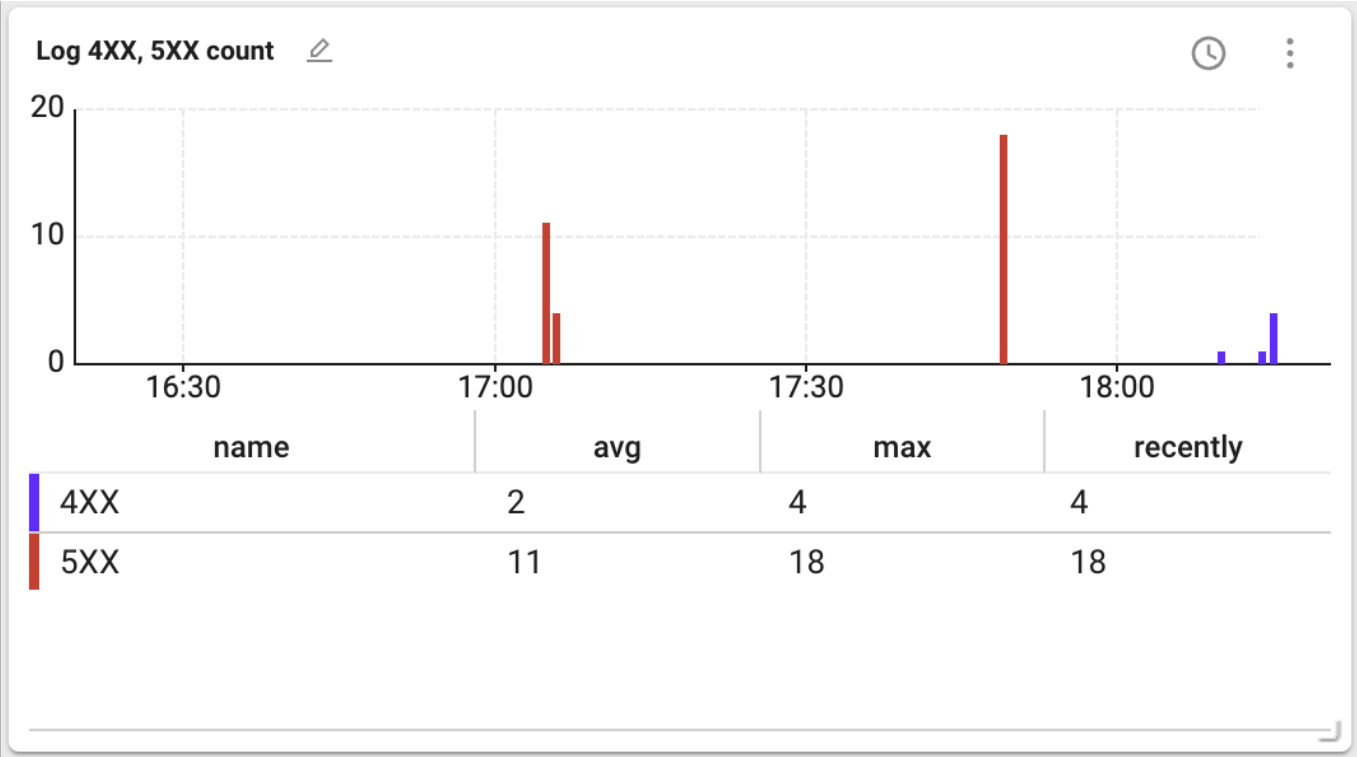
Widget templates

4xx

Q

Log 4XX, 5XX count

If the widget is created, you can see the following data.



- **avg**: Average value of the data during the query period
- **max**: Maximum value of the data during the query period
- **recently**: Final value of the data during the query period

Status code success rate parser setting item

The **Status code success rate parser** generates HTTP request success rate data based on the status values parsed by the primary parser. To exclude specific status codes, enter the status codes to exclude. For status parsing, see [4xx, 5xx status code parser setting item](#).

Setting item

Set value	Description	ETC
Category	It is the category to generate the request success rate data.	required
Log detection condition	Enter the search key and search value to be applied as the filter. The request success rate data is generated only for the log data that meets the log detection condition. If you do not enter any log detection condition, the data is generated for all logs.	optional
Status codes to exclude	The status codes to exclude upon generation of request success rate data. If no entry, the request success rate data is generated for all success rate status codes that correspond between 2xx and 3xx.	optional

Data Search

If all parsers have been registered, go to **Integrated Flex Board** and then create the log request success rate widget.

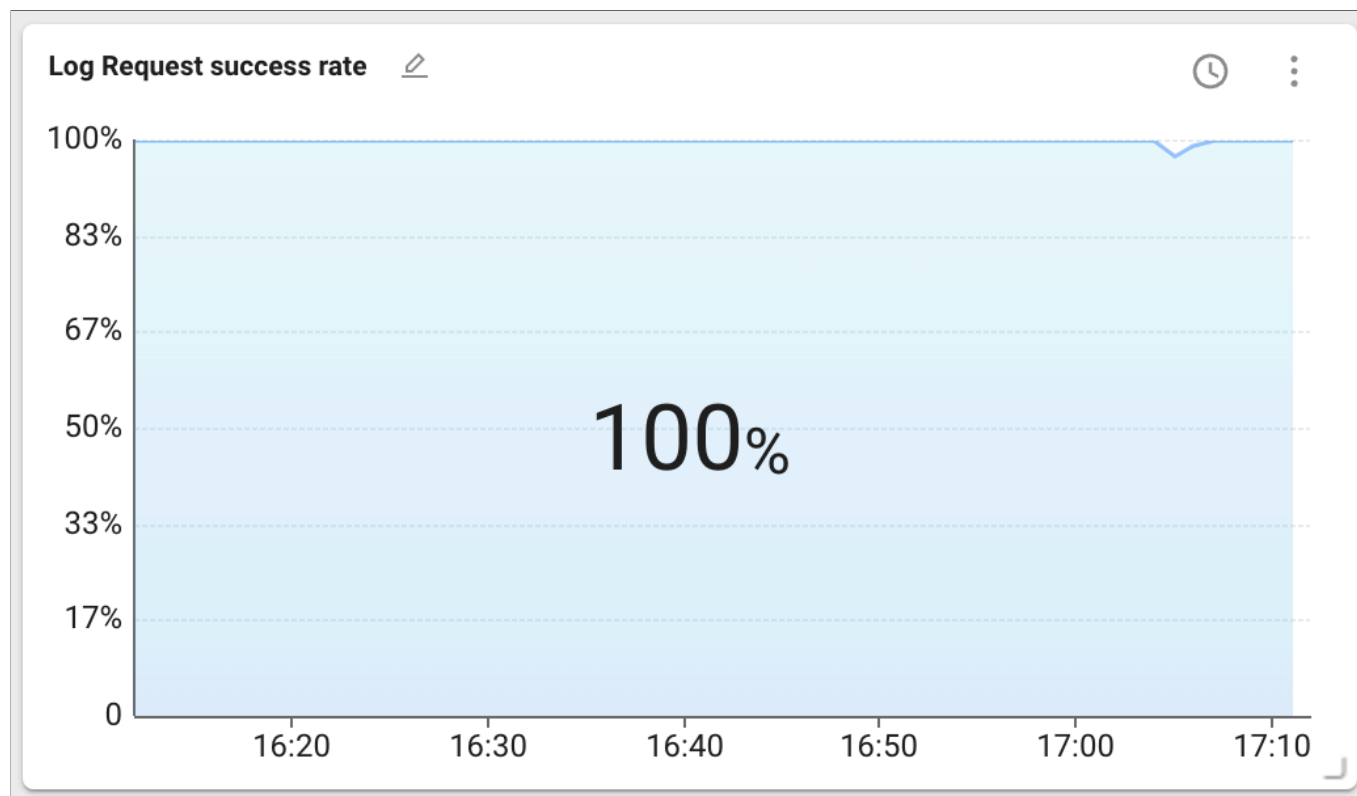
Widget templates

⇌ All metrics

request success

Log Request success rate

If the widget is created, you can see the following data.



The data above the chart represents statistics for the lookup period. You can select the statistical method with the latest value, maximum value, or average value. The latest value is selected by default.

Secondary parser processors

In addition to **4xx, 5xx status code parser** and **Status code success rate parser**, the log secondary parser provides the following field processing processors.

Log secondary parser processor list

Processor	Description
Contains all keywords filter	Adds a flag field to logs that contain all of the specified keywords.
Contains any keyword filter	Adds a flag field to logs that contain at least one of the specified keywords.
Rename field	Renames an existing field name to a different name.

Processor	Description
Status code group classification	Automatically classifies HTTP status codes into 2xx/3xx/4xx/5xx groups.
URL static/dynamic classification	Classifies a URL as static content if it has an extension (.), and as a dynamic servlet if it does not.
URL parameter removal	Stores the value with the query parameters after ? removed from the URL.
Response time classification	Automatically classifies the response time (ms) into fast/normal/slow/critical ranges.
JSON field extraction	Extracts specific keys from a JSON string field and stores them as new fields.
GeoIP conversion	Converts an IP address into country/city information.
Custom input	Registers a processor directly with an FQN class name.

Contains all keywords filter

Adds a flag field to logs that contain **all** of the specified keywords.

Set value	Description	Default
Input field	The field name to search for keywords	content
Storage field	The field name to store the result	has_str_all
keyword	The list of keywords to search (enter separated by commas)	required

Usage example: If you set the input field to `content` and the keywords to `error, timeout, database`, the `has_str_all` field is added only to logs that contain all three keywords.

Contains any keyword filter

Adds a flag field to logs that contain **at least one** of the specified keywords.

Set value	Description	Default
Input field	The field name to search for keywords	content
Storage field	The field name to store the result	has_str_any
keyword	The list of keywords to search (enter separated by commas)	required

Usage example: If you set the input field to `content` and the keywords to `reboot, shutdown, restart`, the `has_str_any` field is added to logs that contain at least one of the keywords.

Rename field

Renames an existing field name to a different name. If the target field name already exists, it is not overwritten.

Set value	Description	Default
Source field	The existing field name to rename	required
New field name	The field name after renaming	required

Status code group classification

Automatically classifies HTTP status codes into the `status_2xx`, `status_3xx`, `status_4xx`, and `status_5xx` groups.

Set value	Description	Default
Input field	The field name that contains the status code	status
Storage field	The field name to store the group classification result	status_nxx

Usage example: If the `status` value is `404`, `status_4xx` is stored in the `status_nxx` field.

URL static/dynamic classification

Classifies a URL as **static content** if it contains an extension (.), and as a **dynamic servlet** if it does not.

Set value	Description	Default
Input field	The field name that contains the URL	url
Storage field	The field name to store the classification result	url_type

URL parameter removal

Stores only the path with the query parameters after ? removed from the URL.

Set value	Description	Default
Input field	The field name that contains the URL	url
Storage field	The field name to store the parameter removal result	url_without_params

Usage example: /api/v1/users?page=1&size=20 → /api/v1/users is stored in the url_without_params field.

Response time classification

Automatically classifies the response time (ms) by range. You can set the range thresholds directly.

Set value	Description	Default
Input field	The field name that contains the response time (ms)	elapsed
Storage field	The field name to store the classification result	response_class
Fast upper limit (ms)	The upper limit of the fast range	100
Normal upper limit (ms)	The upper limit of the normal range	500

Set value	Description	Default
Slow upper limit (ms)	The upper limit of the slow range	2000

The classification results based on the default values are as follows.

Response time	Classification
0 ~ 100ms	fast
101 ~ 500ms	normal
501 ~ 2000ms	slow
2001ms or more	critical

Usage example: If you filter by `response_class: critical` in the log explorer, you can query only the logs whose response time exceeds 2 seconds.

JSON field extraction

If a field extracted by the primary parser contains a JSON string, this extracts specific keys from that JSON and stores them as new fields.

Set value	Description	Default
Input field	The field name that contains the JSON string	payload
Keys to extract	The list of JSON keys to extract (enter separated by commas)	required
Storage prefix	The prefix to prepend to the extracted key names (e.g., <code>json_</code>)	optional

- If the entire log is JSON, the primary JSON parser parses it automatically. This processor is useful for additionally parsing the JSON part contained within a text log when using the **GROK primary parser**.
- To protect performance, parsing is skipped if the JSON field size exceeds 10 KB.

Usage example

Primary GROK parsing result

```
payload = {"user":"alice","action":"purchase","product":"laptop","amount":45000}
```

If you set the input field to `payload`, the keys to extract to `user, action`, and the storage prefix to `pay_`, they are extracted as follows.

JSON field extraction result

```
pay_user = alice
pay_action = purchase
```

GeoIP conversion

Converts an IP address into country and city information. It uses the IP2Location database to enable region-based analysis of access logs.

Set value	Description	Default
Input field	The field name that contains the IP address	<code>client_ip</code>
Storage prefix	The prefix of the result field names	<code>geo</code>

As a result, two fields, `{prefix}_country` and `{prefix}_city`, are created.

Usage example: If the input field `client_ip` value is `223.130.195.200` and the storage prefix is `geo`:

GeoIP conversion result

```
geo_country = KR
geo_city = Seongnam
```

Custom input

You can register a processor that is not in the list directly with an FQN (Fully Qualified Name) class name.

Custom input format

```
io.whatap.logsink.processor.ClassName{"param1":"value1","param2":"value2"}
```

ⓘ The class name and parameter format must be accurate. If the input is incorrect, the processor may not work.

Secondary parser chaining

You can register multiple secondary parsers in the same category and chain them. Each processor runs in the order of registration.

Original log

```
2026-04-23T10:30:00.123+0900|INFO|api-01|192.168.1.10|/api/v1/
orders?detail=true|200|1234|{"user":"alice","action":"purchase"}|Request processed
```

Order	Parser	Input → Output
Primary	GROK	timestamp, level, host, client_ip, url, status, elapsed, payload, content
Secondary-1	JSON field extraction	payload → user, action
Secondary-2	Response time classification	elapsed → response_class (slow)
Secondary-3	Status code group classification	status → status_nxx (status_2xx)
Secondary-4	URL parameter removal	url → url_without_params (/api/v1/orders)
Secondary-5	GeoIP conversion	client_ip → geo_country (KR), geo_city (Seoul)

ⓘ The **Category** of the secondary parser must be set identically to that of the primary parser. If you set a different category, the secondary parser is not applied to that log.

Fast Index Setting

Select the **Fast index setting** tab at the top of **Log Configuration**. Collecting a large number of logs can significantly decrease the log search performance. The frequently used search conditions are created as **index**, you can improve the log search performance for quick search. The setting items are as follows:

Set value	Required	Description
Category	Mandatory	Category to be set as fast index
Search Key	Mandatory	Search key for fast index setting
Case insensitive	Option	Whether to be case sensitive
Rule	Mandatory	* must be included at least one.
Enabled	Mandatory	Active or inactive (default value is <code>true</code>)

Importing/exporting log settings

You can save parser settings, Fast index setting, and filter settings in JSON file format, and apply them by importing a JSON file from other projects. You can reduce the hassle of creating the settings repeatedly for each project.

1. Add parser settings, Fast index setting, and filter settings to one project.
2. Select **JSON**  on the upper right of each configuration tab.
3. On the upper right of the **Export JSON** window, select **Export**.
4. The JSON configuration file is saved on your PC.
5. Move to another project and then go to the **Log > Log setting** menu.
6. Select the configuration tab where you have exported the JSON configuration file before, and then select .
7. If the file selection window appears, select the JSON configuration file saved on your PC.

8. If the **Import JSON** window appears, check the configuration file and then select **Add to list** or **Overwrite**.
9. Select **Save** on the upper right of the screen.

⚠ After importing the JSON configuration file, you cannot save the imported settings unless you select **Save**.

Log long-term archive statistics

Select the **Log long-term archive statistics** tab at the top of **Log Configuration**. Log data is so large and difficult to retain for a long time. Using the Set log statistics data function, you can **save information on how many logs that meet specific conditions are collected every 5 minutes**. Even if actual log data has been deleted for a long time, you can check the trend of how many logs that meet the conditions are collected.

Adding the log long-term archive statistics

× Log long-term archive statistics Add

Statistic key*

Please input the Statistic key.

Data retention period(disk usage)

15 Days (About 3 MB) ▾

Category*

Please select the Category.

Log detection condition*

Please select a category first. ▾

Please select a category first.

Exclude ☐

Case Sensitive ☒

+ Add

Add

If you select + **Add** under the **Log long-term archive statistics** tab, the **Log long-term archive statistics Add** window appears. You can add rules by using + **Add** or delete the created rules by using the - icon.

Setting item

Field	Description
Category	Category to apply the rule to.
Statistic Key	The same key cannot be set twice to save when a log that meets the rule is generated.
Log detection condition	Condition for generating log statistical data. Statistical data is generated based on how many logs that meet this condition are collected.
Exclude	If checked, statistical data is generated with values that do not correspond to the entered conditions.
Case Sensitive	Specify case sensitivity for the values of the entered log detection conditions.
Enabled	Active or inactive (default value is <code>true</code>)

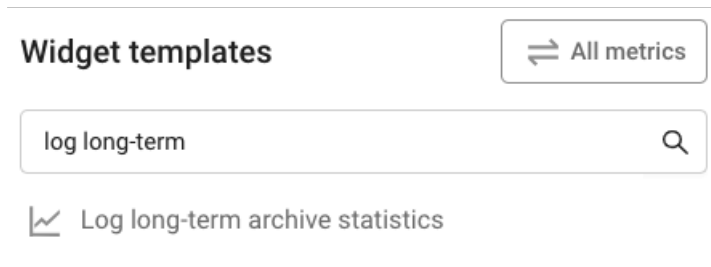
Example

If a setting is added as follows, statistical data is generated with a key value of **TotalCount** for the logs whose status is `200` or `300`.

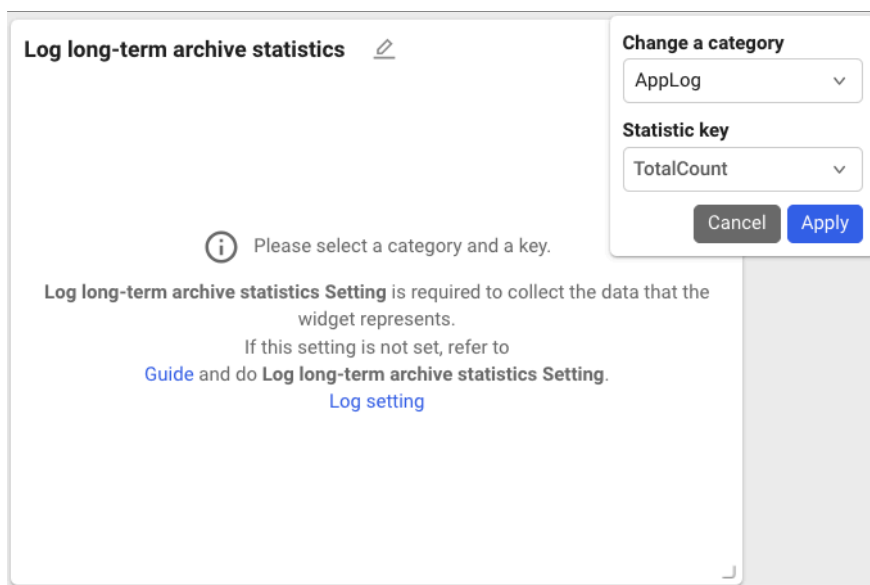
Log long-term archive statistics You can generate statistical data about how many logs have been collected that meet certain conditions. 						+ Add	Save
Category	Statistic key	Log detection condition	Data retention period(disk usage)	Enable			
AppLog	TotalCount	status 200,300	15 Days (About 3 MB)	☐			
AppLog	TotalCount	status 200	15 Days (About 3 MB)	☐			
AppLog	TotalCount	status 300	15 Days (About 3 MB)	☒			
AppLog	TotalCount	status 200,300	15 Days (About 3 MB)	☒			
AppLog	TotalCount	status 200,300	15 Days (About 3 MB)	☐			

Data Search

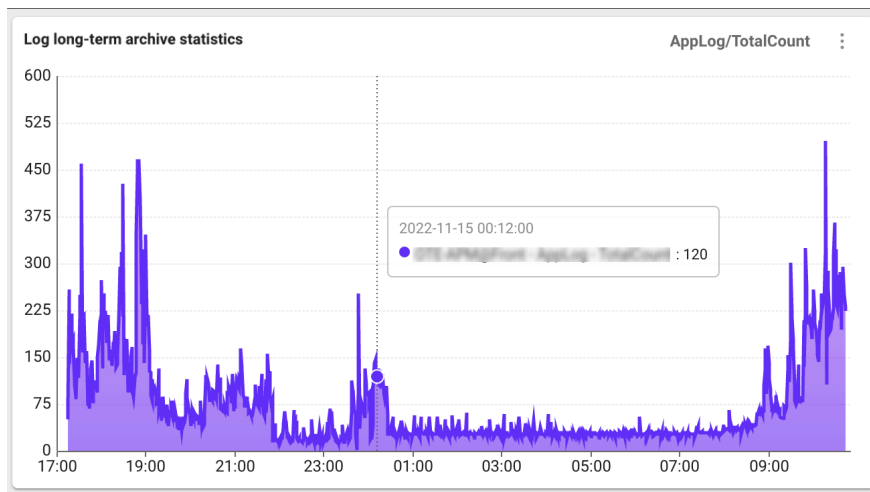
1. Create a widget by searching **Log long-term archive statistics** in **Widget templates** of **Integrated Flex Board**.



2. Enter the **Category** and **Statistic key** to view and then select **Apply**.



3. With the added settings, you can check the **Log long-term archive statistics** data as follows.



Personal information de-identification

In **De-identification of personal information**, you can mask personal information in log data or replace it with safe values.

- Encryption: Specified sensitive data is encrypted and stored in the database.
- Masking: When displayed on the log screen, sensitive data is masked (*******) by default.
- Substitution: Sensitive data is replaced with a user-specified value that is safe.

Personal information classification

The items classified as personal information are as follows: If there are any items that need to be treated as sensitive data in addition to personal information, mask or replace the desired items in the log data by using the **De-identification of personal information** feature.

- National ID number
- Passport number
- Driver's license number
- Foreigner registration number
- Credit card number

- Bank account number
- Biometric information

Required roles

De-identified personal information is displayed as masked. Only users with the **log personal information query** role can view original data through the **Display personal information** toggle. Adding or modifying de-identification target search keys requires the **log editing** role.

❗ For more information about the **log personal information query** and **log editing** roles, see [the following](#).

Setting the personal information de-identification

You can protect sensitive information by applying masking to new log data.

Log Configuration

Getting started with log monitoring | Log primary parser setting | Log secondary parser setting

De-identification of personal information

When a search key containing per

Enable	Category
☒	AppLog

@txid agenttime @mt

Log's personal information de-identification checking

It de-identifies the

Time
< 2025/01/21 16:47 ~ 2025/01/21 17:47 1 hr >

Category
Please select the Category.

Personal information entity
▼

Personal information pattern

Whether to apply substitution
☒

Value to substitute ⓘ

Start inspection Stop inspection

De-identification of personal information Add

Category *
Please select the Category.

Search key *
Select search keys to be encrypted and masked.
Enter the search key to encrypt.

Add

1. In the **De-identification of personal information** tab, select **Add**.
2. In the **De-identification of personal information Add** pane on the right, select **Category** and enter the **Search key** to mask.
3. Click **Add** to add the setting.
4. After checking the de-identified items in the **De-identification of personal information** list, click **Save** on the upper right to save the current settings.

De-identification of personal information If you add a search key containing personal information, the data is encrypted, stored, and masked.

+ Add
Save

Enable	Category	Search key	Edit
☐	VirtualLog	age.n	
☒	VirtualLog	status	
☒	AppLog	agenttime	

- The de-identified search keys are output in the format of `search key.pii` in the log content.
- You can perform ☒ **Enable** for de-identified items in the **Enable** column of the **De-identification of personal information** list.
- You can **modify** and **remove** in the **edit** column of the **De-identification of personal information** list.

- ✓ If the search key name is specified as `search-key.p` in the **Log primary parser setting** tab, it is automatically registered as a de-identification target. You can check or disable auto-registered items in **Specified search key** after selecting the **Category** in the **De-identification of personal information** tab.

We recommend configuring de-identification directly in the **De-identification of personal information** tab rather than using the `.p` suffix in parser settings.

Personal information display

De-identified data is displayed as masked in the **Live Tail**, **Log Trend**, and **Log Search** menus. Users with the **log personal information query** role can click the **Display personal information** ☒ button in the log list to view the original data.

▼	java-demo2	2024-12-16 16:06:49.988	area	river	!privacy	true	!status.pii	***	pcode	8	city	seoul	onodeName	node-0	myname.p	***
			oid	-1737750367	okind	1152715164	price.n	12000	oname	java-demo2	age.n	33				
			agenttime	1734332809555	category	VirtualLog	okindName	dev-okind-0	onode	1693789385						
			You need not love me, if you choose not to.													
▶	java-demo1	2024-12-16 16:06:49.993	area	mountain	!privacy	true	!status.pii	***	pcode	8	city	junju	onodeName	node-1	myname.p	****
			Waiting does not have to aim for a meeting.													
▼	java-demo2	2024-12-16 16:06:49.988	area	river	!privacy	true	!status.pii	500	pcode	8	city	seoul	onodeName	node-0	myname.p	lee
			oid	-1737750367	okind	1152715164	price.n	12000	oname	java-demo2	age.n	33				
			agenttime	1734332809555	category	VirtualLog	okindName	dev-okind-0	onode	1693789385						
			You need not love me, if you choose not to.													
▶	java-demo1	2024-12-16 16:06:49.993	area	mountain	!privacy	true	!status.pii	200	pcode	8	city	junju	onodeName	node-1	myname.p	hong
			Waiting does not have to aim for a meeting.													
▶	java-demo6	2024-12-16 16:06:50.064	area	sea	!privacy	true	!status.pii	200	pcode	8	city	daegu	onodeName	node-1	myname.p	song
			Waiting does not have to aim for a meeting.													

- ❗ The search key specified in the **De-identification of personal information** tab appears in the log content of the Log menu in the format of `Search key.pii`.

Example of status.pii ***

Checking the log's personal information de-identification

- ❗ If you replace the existing data using the **Log's personal information de-identification checking** feature, the data cannot be recovered.

Log's personal information de-identification checking identifies personal information in existing log data and replaces it with the **Value to substitute** specified by the user.

Replacement by target key

If you specify a target key, only the value stored in that key is replaced, and the rest of the log content is kept as is. The target key applies to both the `field` value and the `tag` value of a log. If you do not specify a target key, personal information is identified and replaced across the entire log content, the same as before.

- ❗ Replacement applies only to the stored log data. The original value may remain in the search index even after replacement.

You specify the target key in the **Target key** input field of the log personal information de-identification check

form. On the screen, **field** and **tag** are not distinguished and are entered as a single target key.

Log's personal information de-identification checking

It de-identifies the accumulated logs.

Check inspection history

Time

< 2025/03/21 12:26 ~ 2025/03/21 13:26 1 hr >

Category

VirtualLog

Personal information entity

Resident registration number

Personal information pattern

(?<!\w)\d{2}(0[1-9]|1[0-2])((0-2)[0-9]|3[0-1])-[1-4]\d{6}(?!\\w)

Whether to apply substitution

☐

Value to substitute

*****,*****

Start inspection

Stop inspection

Category

VirtualLog

Search start time

2025/03/21 12:26

Search end time

2025/03/21 13:26

Personal information entity

Resident registration number

Personal information pattern

(?<!\w)\d{2}(0[1-9]|1[0-2])((0-2)[0-9]|3[0-1])-[1-4]\d{6}(?!\\w)

Value to substitute

*****,*****

Pattern matching result

0 Check results

Inspection start time

2025/03/21 13:28

Inspection end time

2025/03/21 13:28

Whether to apply substitution

No

Inspection result

SUCCESS

1. Specify the start time and end time.
2. Set values for **Category**.
3. Select **Personal information entity** from the examples or **Direct Input**.

! When selecting **Direct Input**, directly enter a regular expression pattern to identify personal information.

4. Select whether to replace personal information, if it is identified via the **Whether to apply substitution** toggle.
5. Enter **Value to substitute** to change the string identified by the regular expression pattern in the existing log.

! • **Value to substitute** must not be longer than the original value.

• When enabling the **Whether to apply substitution** toggle, be sure to enter **Value to substitute**.

6. Enter the conditions and then select **Start inspection**.

! You can stop this step at any time by clicking **Stop inspection** during the checkup.

- When applying substitution, enter **Replace** in the **Start of inspection for personal information de-identification** confirmation window, and then click the **OK** button.

Start of inspection for personal information de-identification ×

The value that matches the selected personal information pattern is replaced.

! Data cannot be restored after replacement.

If replacement is required, see below. "Replace" Please enter directly.

Cancel
 OK

- After the inspection is finished, you can see **Inspection result** on the right as in the example image.
 - Category:** It displays the categories checked.
 - Search start time** and **Search end time:** It displays the time range for the checkup.
 - Personal information entity:** It displays the personal items that have been checked.
 - Personal information pattern:** It displays the regular expression pattern used.
 - Value to substitute:** It displays the applied replacement value.
 - Pattern matching result:** The number of personal data identified is displayed. You can check the sample logs by clicking **Check results**.
 - Inspection result:** It displays the result where personal information has been identified.

SUCCESS (check success), FAILED (check failure), STOP (check stop), TIMED_OUT (check ended due to timeout)

Check inspection history

To view the results of past checks after performing de-identification check for the log's personal information, click the **Check inspection history** button at the upper right. You can trace the substitution history, inspection

time, and applied personal information items. You can also download the list information in [CSV](#) file format by clicking the  [CSV](#) button at the upper right of the **Check inspection history** window.

Check inspection history

< 2025/03/25 08:17 ~ 2025/03/25 09:17 1 hr  >

Personal infor... 



 CSV

stitute	Pattern matching result	Inspection start time	Inspection end time	Whether to apply substi...	Inspection result
***	0	2025/03/25 09:16	2025/03/25 09:16	No	SUCCESS
***	0	2025/03/25 09:16	2025/03/25 09:16	No	SUCCESS
***	0	2025/03/25 09:16	2025/03/25 09:16	No	SUCCESS

Personal information pattern

Examples of regular expression patterns used to identify personal information in the existing log are as follows:

Personal information pattern	Personal information example	Regular expression pattern	Replacement value
National ID number	123456-1234567	(?<!w)d{2}(0[1-9] 1[0-2])([0-2][0-9] 3[0-1])-[1-4]d{6}(?!w)	*****_ *****
Passport number	M1234567	([MSRODTC][0-9]{7}\b)	*****
Driver's license number	01-23-456789-00	(?<!w)d{2}-d{2}-d{6}-d{2}(?!w)	**_**_ *****_**
Foreigner	987654-1234567	(?<!w)d{6}-?[5-8]d{6}(?!w)	*****_

Personal information pattern	Personal information example	Regular expression pattern	Replacement value
registration number			*****
Credit card number	1234-5678-9012-3456	(?<!w)d{4}-d{4}-d{4}-d{4}(?!w)	****_****_ ****_****
Mobile phone number	010-1234-5678	(?<!w)(+82-?1[016789]-?d{3,4}-?d{4} 01[016789]-?d{3,4}-?d{4})(?!w)	***_****_ ****
Email address	example@domain.com	([a-zA-Z0-9][a-zA-Z0-9._%+-]*@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,})	***@***.***
Bank account number	123456-01-654321	(?<!w)d{6}-d{2}-d{6}(?!w)	*****_**_ *****
Business registration number	111-11-01111	(?<!w)d{3}-d{2}-d{5}(?!w)	***_**_*****
Corporate registration number	111111-1111111	(?<!w)d{6}-?d{7}(?!w)	*****_ *****

Parsing The Logs

The log parser helps you transform the irregular logs into a structured form. WhaTap Log Monitoring provides two types of parsers.

- **GROK parser:** The logs collected in arbitrary forms are parsed using regular expressions and GROK syntax.
- **JSON parser:** The logs collected in the JSON form are parsed.

! Common precautions

- If multiple parsers are registered in the same category, only the first matching parser is applied.
- WhaTap has the role to disable the parsers that may affect the stability of the WhaTap service.

GROK parser

If the logs are collected in an irregular form, you can use the GROK parser to parse them. GROK syntax provides named regular expressions, allowing you to use regular expressions more easily.

Starting GROK

GROK provides two types of syntaxes.

1. `%{SYNTAX:SEMANTIC}` : Syntax provided in the GROK library. Using the **named regular expressions**, you can extract tags. For usage examples, see [the following](#).
 - **SYNTAX:** Specify the named regular expressions provided by GROK.
 - **SEMANTIC:** Specify the name for the matched value.

! named regular expressions

The syntax is provided by GROK. This function provided by GROK allows you to assign names to complex regular expressions.



name	regular expression
WORD	<code>\b\w+\b</code>
SPACE	<code>\s*</code>
NOTSPACE	<code>\S+</code>
UUID	<code>[A-Fa-f0-9]{8}-(?:[A-Fa-f0-9]{4}-){3}[A-Fa-f0-9]{12}</code>

To see all named regular expressions provided by WhaTap, see the following [link](#).

2. `(?<SEMANTIC>REGX)` : The **named capturing group** syntax for regular expressions. You can use regular expressions to extract tags according to the user's intent. For usage examples, see [the following](#).

- **SEMANTIC**: Specify the name for matched values.
- **REGX**: Enter regular expressions for matching.



named capturing group

It is the syntax provided by regular expressions.

- capturing group: This indicates the function to bundle multiple tokens into one as a single matching unit.
- named capturing group: Name assigned to the capturing group.
- Let's look at the string matching example. [dev@whatap.io](#)
 - Example 1 `(\w+)@(\w+\.\w+)`
 - Example 2: When the entire email is matched and the username and domain are matched additionally, `(?<username>\w+)@(?<domain>\w+\.\w+)`

%{SYNTAX:SEMANTIC} Usage example

The following is an example that uses the `%{SYNTAX:SEMANTIC}` syntax.

Sample log

```
[2023-08-08 02:02:30,101 GMT][INFO ][i.w.y.l.c.LogSinkDexScheduleThread.realProcess(159)] 8 VirtualLog 20230808
02:01:00.000 {area=4, city=5} 56ms
```

You can look at the sample log to infer what each word means. When replacing each part with semantic words, it can be expressed as follows:

semantic replace

```
[date][logLevel][caller] projectCode logCategory dexBuildStartTime {area=areaEnum, city=cityEnum} dexBuildElapsed
```

Any semantic word can be replaced with a regular expression. The GROK parser allows you to use the predefined named regular expressions. `TIMESTAMP_ISO8601`, `LOGLEVEL`, and `DATA` used here are named regular expressions provided by GROK. These values are matched after replacing each of them with the following regular expression.

- name: `TIMESTAMP_ISO8601`
 - regular expression: `%{YEAR}-%{MONTHNUM}-%{MONTHDAY}[T]
%{HOUR}:%{MINUTE}(?:%{SECOND})?%{ISO8601_TIMEZONE}?`
- name: `LOGLEVEL`
 - regular expression: `LOGLEVEL
([Aa]lert | ALERT | [Tt]race | TRACE | [Dd]ebug | DEBUG | [Nn]otice | NOTICE | [Ii]nfo | INFO | [Ww]arn(?:ing)? | WARN(?:ING)?`
- name: `DATA`
 - regular expression: `.*`

GROK parsing pattern

```
\[%{TIMESTAMP_ISO8601:date}\sGMT\]\[%{LOGLEVEL:level}\s\]\[%{DATA:caller}\]
```

If you parse with the above syntax, you can extract tags as follows. In this way, GROK's `%{SYNTAX:SEMANTIC}` syntax helps you apply complex and long regular expressions easily and concisely.

Tag extraction

```
- date : 2023-08-08 02:02:30,101
- caller : i.w.y.l.c.LogSinkDexScheduleThread.realProcess(159)
- level : LEVEL
```

(?<SEMANTIC>REGX) Usage example

The parts that do not match with the named regular expressions can be parsed using the `(?<SEMANTIC>REGX)` pattern. In the [sample log](#) above, the parts that cannot be parsed with the `%{SYNTAX:SEMANTIC}` syntax are as follows.

Unparsed area

```
8 VirtualLog 20230808 02:01:00.000 {area=4, city=5} 56ms
```

When replacing each part of the log with semantic words, it can be expressed as follows.

semantic replace

```
projectCode logCategory dexBuildStartTime {area=areaEnum, city=cityEnum} dexBuildElapsed
```

These irregularly strings can be parsed using the `(?<SEMANTIC>REGX)` syntax.

Regular expressions that match for each sample log parsing keyword

Parsing keyword	(?<SEMANTIC>REGX)
8	(?<projectCode>\d)
VirtualLog	(?<logCategory>\w*)
20230808 02:01:00.000	(?<dexBuildStartTime>\d{8}\s\d{2}:\d{2}:\d{2}\.\d{3})
area=4	area=(?<areaEnum>\d)
city=5	city=(?<cityEnum>\d)

Parsing keyword	(?<SEMANTIC>REGX)
56ms	(?<dexBuildElapsed>\d{2})ms

Basic regular expression syntax

Syntax rule	Meaning	Alias
<code>?</code>	0 or 1	-
<code>+</code>	1 or more	-
<code>*</code>	0 or more	-
<code>a{5}</code>	exactly 5	-
<code>\w</code>	word character	<code>[a-zA-Z_0-9]</code>
<code>\s</code>	white space	-
<code>.</code>	any character except newline	
<code>[abc]</code>	any of	-
<code>[^abc]</code>	not a,b, or c	-
<code>[a-z]</code>	character between a and z	-
<code>[1-3[7-9]]</code>	union (combining two or more character classes)	-
<code>[1-6&&[3-9]]</code>	intersection	-
<code>[0-9&&[^2468]]</code>	subtraction	-
<code>a{2,}</code>	2 or more	-
<code>a{1,3}</code>	between 1 and 3	-



Syntax rule	Meaning	Alias
<code>a+?</code>	match as few as possible	-
<code>{2,3}?</code>	match as few as possible	-
<code>(abc)</code>	capturing group (processes multiple strings as a single unit)	-
<code>\d</code>	digit	<code>[0-9]</code>
<code>\D</code>	non-digit	<code>[^0-9]</code>
<code>\W</code>	non-word character	-
<code>\S</code>	non-white space	-

By connecting the parsed keywords with space (`\s`) and special character escapes (`\{` , `\,` , `\}`), you can apply the pattern as follows.

GROK parsing pattern

```
(?<projectCode>\d)\s(?<logCategory>\w*)\s(?<dexBuildStartTime>\d{8})\s\d{2}:\d{2}:\d{2}\.\d{3})\s\{area=(?<areaEnum>\d),\scity=(?<cityEnum>\d)\}\s(?<dexBuildElapsed>\d{2})ms
```

If you parse with the above syntax, you can extract tags as follows.

Tag extraction

```
- projectCode : 8
- logCategory : VirtualLog
- dexBuildStartTime : 20230808 02:01:00.000
- areaEnum : 4
- cityEnum : 5
- dexBuildElapsed : 56
```


Applying GROK

Log Configuration > Log primary parser setting

1. To apply the GROK pattern parser, go to the **Log primary parser setting** tab in **Log Configuration**.
2. Select **+ Add** and then select the **GROK** parser in the **Parser** field.
3. When you select **Register pattern**, the pattern registration and simulation window appears on the right.
4. Enter **Pattern** and **Log**, and then click **Simulation** to check whether parsing is successful with the pattern to apply.

Pattern e.g.

```
[%{TIMESTAMP_ISO8601:date}\sGMT][%{LOGLEVEL:level}\s][%{DATA:caller}]\s(?:<projectCode>\d)\s(?:<logCategory>\d{2}:\d{2}:\d{2}\.\d{3})\s\{area=(?:<areaEnum>\d),\scity=(?:<cityEnum>\d)\}\s(?:<dexBuildElapsed>\d{2})ms
```

Log example:

```
[2023-08-08 02:02:30,101 GMT][INFO ][i.w.y.l.c.LogSinkDexScheduleThread.realProcess(159)] 8
VirtualLog 20230808 02:01:00.000 {area=4, city=5} 56ms
```

5. If the simulation is successful, you can view **Simulation result** and **Performance measurement results**.
6. When you click **Apply pattern** after simulation, the pattern that has been entered for the selected parser is applied.
7. After applying the pattern, enter values for **Category**, **Log detection condition**, and **Pattern**.

Parser Add

Parser*

WhaTap reserves the right to disable parsers that may affect the stability of the WhaTap service.

GROK

parser pattern*

The pattern is registered after the simulation is performed to ensure that the log is successfully parsed with the entered pattern.

Success

```

\[%{TIMESTAMP_ISO8601:date}\sGMT\}\[%{LOGLEVEL:level}\s\}\[%{DATA:caller}\}\s(?:<projectCode>\d)\s(?:
<logCategory>\w*)\s(?:<dexBuildStartTime>\d{8}\s\d{2}:\d{2}:\d{2}\.\d{3})\s\{area=(?:<areaEnum>\d),\scity=(?:
<cityEnum>\d)\}\s(?:<dexBuildElapsed>\d{2})ms

```

Modify parser pattern

Category*

If multiple parsers are registered in the same category, only the first matching parser is applied.

AppLog

Log detection condition

If you don't enter any filter, or if you enter only one of the search keys/values, the pattern is applied to all logs.

oname

demo-8100

Add

- **Category**

Select a log category. **Category** is required.

- **Log detection condition**

- Only the logs that meet the conditions are applied to the parser.
- Select or enter values for **Search key** and **Search value**.
- **Log detection condition** are applied before all parsers are run. Accordingly, you cannot use the **Tag** that have been added for the parser.

8. Select **Add** to register a parser.

- In the log parser list, you can change the **Priority** or **Enable**, **Edit**, or **Delete** the parser.
- After **parser simulation**, you can register a pattern.

! GROK parser precautions

- GROK parser supports two patterns: `%{SYNTAX:SEMANTIC}` and `%{SYNTAX:SEMANTIC} .`
- When using the `%{SYNTAX:SEMANTIC}` pattern, `SEMANTIC` must be entered.
- When using the `%{SYNTAX:SEMANTIC}` pattern, `SEMANTIC` must be unique in a parser.
- When using the `(?<SEMANTIC>REGX)` pattern, `SEMANTIC` can only contain characters (a-z, A-Z), numbers (0-9), and specified special characters (`.` , `_` , `-`).
- `SEMANTIC` must start with a character (a-z, A-Z).
- `SEMANTIC` must end with a character (a-z, A-Z) or number (0-9).

Selecting additional GROK patterns

During GROK parser simulation, in addition to the basic patterns (`WORD` , `NUMBER` , `IP` , etc.), you can select **additional patterns** that match a specific log format. 32 pattern types, such as Java, Nginx, and MySQL, are provided grouped into 11 categories.

Additional pattern category dropdown

1. In the parser simulation window, click the dropdown in the **Additional patterns (optional)** area.
2. From the list of pattern types grouped by category, select the pattern type you need. You can also search by entering a pattern type name or description in the search box.
3. The selected pattern type is loaded together with the basic patterns during simulation.

! The basic patterns (Core) are always included automatically. Selecting additional patterns is optional.

Pattern type list by category

Category	Pattern type
Web Servers	haproxy, httpd, iis, nginx, squid
Application Servers & Frameworks	java, log4j, rails, ruby, springboot

Category	Pattern type
Databases	mongodb, mysql, oracle, postgresql, redis
Cloud	aws
System & Infrastructure	docker, kubernetes, linux-syslog
Messaging & Search	elasticsearch, kafka
Network & Security	bind, bro, firewalls, junos, zeek
Mail	exim, postfix
Monitoring & Backup	bacula, mcollective, nagios
Build Tools	maven

Usage example

To parse a Tomcat Catalina 8 log, select the **java** pattern type additionally and then use the `%{CATALINA8_LOG}` pattern.

Sample log

```
31-Jul-2020 16:40:38 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Server version name: Apache Tomcat/8.5.57
```

GROK parsing pattern

```
%{CATALINA8_LOG}
```

Tag extraction

```
- log_timestamp : 31-Jul-2020 16:40:38
- log.level : INFO
- java.log.origin.thread.name : main
- java.log.origin.class.name : org.apache.catalina.startup.VersionLoggerListener
```

```
- log.origin.function : log
- log_message : Server version name: Apache Tomcat/8.5.57
```

Viewing pattern details

You can check which patterns are included in the selected additional pattern type in the details window. You can look up the pattern definitions (regular expressions) and sample logs and apply them directly to the simulation.

Viewing pattern details

After selecting an additional pattern type, click the selected pattern type tag to open the details window on the right. The details window is displayed side by side without hiding the simulation form, so you can proceed with the simulation while checking the pattern definitions.

You can use the following functions in each pattern item.

Button	Function	Description
Sample	Enter a sample log	Automatically enters an example log that matches the pattern into the Simulation parsing log input box.
Select	Insert a pattern	Inserts it into the Pattern input box in the <code>%{pattern_name}</code> format. The corresponding pattern type is also selected automatically.
Copy	Copy to clipboard	Copies the <code>%{pattern_name}</code> format to the clipboard.

- When you click a pattern item, its pattern definition (regular expression) expands. You can expand multiple patterns at the same time to compare their definitions.
- In the `%{PATTERN:field}` references included in a pattern definition, the pattern name, field name, and literal characters are displayed distinguished by different colors.
- You can run the **simulation** directly with the log entered by the **Sample** button to check the parsing result.

Saving simulation sample logs

The log used in the simulation is saved automatically together with the parser settings. When you edit a parser, the sample log used previously is restored automatically to the simulation window, so you do not need to enter the log again each time.

JSON parser

If logs are collected in JSON format, they can be easily parsed using the JSON parser.

Applying the JSON

Log Configuration > Log primary parser setting

1. To apply the JSON pattern parser, go to the **Log primary parser setting** tab in **Log Configuration**.
2. Select **+** **Add** and then select the **JSON** parser in the **Parser** field.
3. When you select **Register pattern**, the pattern registration and simulation window appears on the right.
4. Enter **Pattern** and **Log**, and then click **Simulation** to check whether parsing is successful with the pattern to apply.

Example

```
2023-08-08 02:43:28,615 -- {"host":"10.21.3.24","method":"POST","status":"200","url":"http://dev.whatap.io/yard/
api/flush"} --
```

In the example log, if **Prefix** and **Postfix** are specified as `--` and `url` is specified in **Ignore**, only three **Tag** (`host` , `method` , `status`) are created.

- **Prefix** Specify the position where the JSON format begins in the log. If the entire log is in JSON format, set an empty value.
- **Postfix** Specify the position where the JSON format ends in the log. If the entire log is in JSON format, set an empty value.
- **Ignore** Specify the key for which tags are not created during JSON formatting.

5. If the simulation is successful, you can view **Simulation result** and **Performance measurement results**.

Parser Simulation

View Guide

Pattern

agenttime, category, content, logContent, message, pcode, time, timestamp is a reserved word.

Reserved words do not work even if registered as log detection conditions or patterns.

Prefix

Postfix

Ignore

Log *

Log simulation is essential and additional pattern registration may be required depending on simulation results.

```
2023-08-08 02:43:28.615 --
{"host":"10.21.3.24","method":"POST","status":"200","url":"http://devote.whatap.io/yard/api/flush"} -

```

[Simulation](#)

Simulation result Success

Key	Value	Result
method	POST	Ok
host	10.21.3.24	Ok
status	200	Ok

Performance measurement results Success

Simulation count	Time min(ns)	Time max(ns)	Time a
1	7,047	7,047	7,047
10	2,663	4,255	2,872
100	2,655	4,574	2,818
1,000	2,651	94,191	2,961
10,000	2,651	146,472	2,987
100,000	2,648	194,424	2,917
1,000,000	2,647	80,465,427	3,136

[Apply pattern](#)

6. When you click **Apply pattern** after simulation, the pattern that has been entered for the selected parser is applied.
7. After applying the pattern, enter values for **Category**, **Log detection condition**, and **Pattern**.

× Parser Add

Parser*
 WhaTap reserves the right to disable parsers that may affect the stability of the WhaTap service.

JSON

parser pattern*
 Simulate that the log is parsed successfully. Simulation must be performed, and additional pattern registration may be required depending on the simulation results.

✓ Success

Prefix --

Postfix --

Ignore url

Simulation verification and pattern modification

Category*
 If multiple parsers are registered in the same category, only the first matching parser is applied.

AppLog

Log detection condition
 If you don't enter any filter, or if you enter only one of the search keys/values, the pattern is applied to all logs.

oname demo-8100

Add

- **Category**

Select a log category. **Category** is required.

- **Log detection condition**

- Only the logs that meet the conditions are applied to the parser.
- Select or enter values for **Search key** and **Search value**.
- **Log detection condition** are applied before all parsers are run. Accordingly, you cannot use the **Tag** that have been added for the parser.

8. Select **Add** to register a parser.

- In the log parser list, you can change the **Priority** or **Enable**, **Edit**, or **Delete** the parser.
- Before registering a parser, you can check whether the pattern to register is normal through



Simulation. It is the same as the GROK parser registration simulation process. See [the following](#).

Example of using JSON format

Sample log

```
{"host":"10.21.3.24","method":"POST","status":"200","url":"http://dev.whatap.io/yard/api/flush"}
```

If the above sample log has been collected, select the **JSON** parser in the **Parser Add** window. Without having to write a complex parsing logic, you can extract the **Tag** for log analysis as follows:

Tag extraction

```
- host : 10.21.3.24
- method : POST
- status : 200
- url : http://dev.whatap.io/yard/api/flush
```

Usage example when configuring part of the JSON format

Some JSON format sample log

```
2023-08-08 02:43:28,615 -- {"host":"10.21.3.24","method":"POST","status":"200","url":"http://dev.whatap.io/yard/api/flush"} --
```

If part of the log is in JSON format as in the example, specify **Prefix** and **Postfix**. WhaTap log monitoring recognizes and parses the area between **Prefix** and **Postfix** in JSON format.

Tag extraction

```
- host : 10.21.3.24
- method : POST
- status : 200
- url : http://dev.whatap.io/yard/api/flush
```

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.

Anomaly Detection

The anomaly detection feature allows you to configure alert notifications by detecting unexpected patterns.

Real-time logs

Sends alert notifications when event conditions configured in real-time collected logs are detected.

 Composite log

Configure composite log events that trigger alert notifications based on log content.

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.

 Anomaly Detection

The anomaly detection feature allows you to configure alert notifications by detecting unexpected patterns.

 Real-time logs

Sends alert notifications when event conditions configured in real-time collected logs are detected.

Composite log

Configure composite log events that trigger alert notifications based on log content.

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.

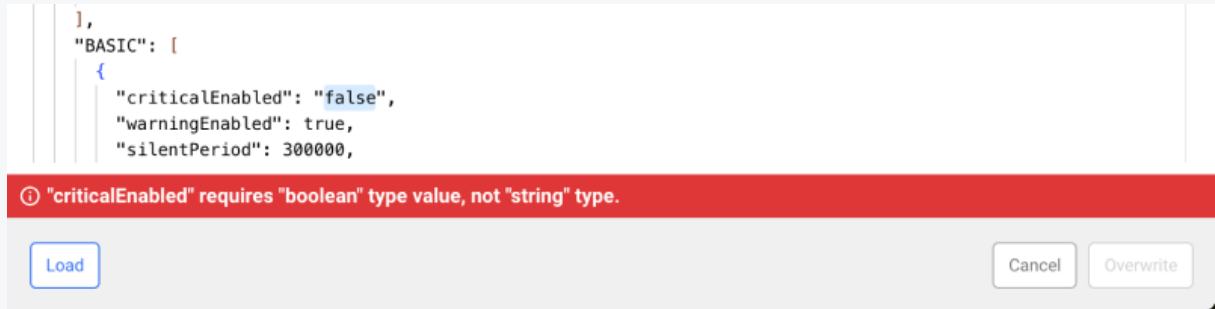
After overwriting, you cannot restore the previous event configuration.

- 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
◊ 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

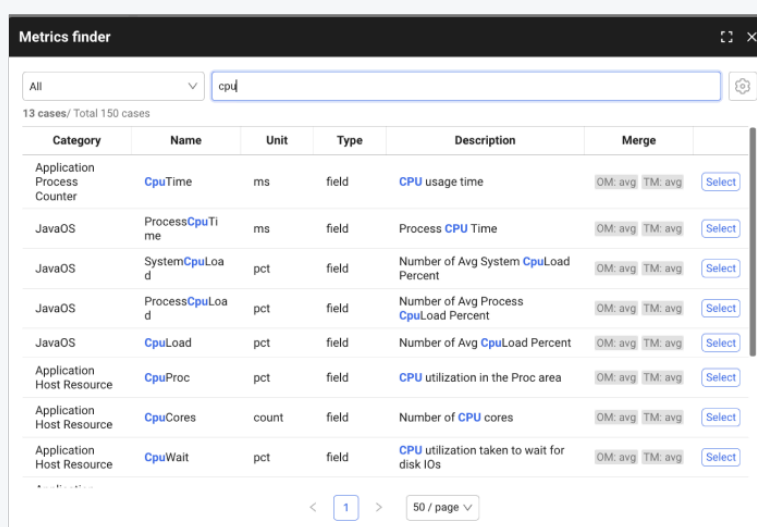
- Enter a category in the search field, or select a category (metric) from the category list. You must select a category to proceed with metric configuration.

- 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.



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](#).

Anomaly Detection

Home > Select Project >  **Alert** > **Event Configuration** > **Anomaly detection** tab

Anomaly detection identifies abnormal rising or falling patterns in metric data and triggers events when movements that differ from the usual behavior are detected.

Default Options

- **Search:** Search for specific events by selecting an event name or an event recipient tag.
- **+ Add Event Rule:** Add a new anomaly detection event.

Table | Anomaly Detection Event List Layout

Item	Description
Event Message	Message sent when an event is triggered
Category	Unit used to classify metric data and serve as the basis for metric event configuration
Field	Metric field used to detect anomalies
Filter	Conditions that limit the data range subject to anomaly detection
Rising/Falling Pattern	Sensitivity of rising or falling patterns
Event Recipient Tag	Users or groups that receive notifications for the event
	Edit event settings
	Delete event
	Event enablement status - Enabled: Event is triggered when conditions are met - Disabled: Event does not run and conditions are not evaluated

Event Configuration

You can configure alert notifications to be sent when metric values rise or fall in a way that differs from their usual behavior.

Add Event Rules

1. Click the [+ Add Event Rule] button on the right side of the screen.
2. Configure **Event Basic Information**, **Metrics**, **Anomaly Detection**, and **Additional Information** in order.

Event Basic Information

1. Select the event level (**Critical**, **Warning**, Info).
 - This determines the event priority and alert level.
2. Enter the alert message to be delivered when the event is triggered.
 - You can use variables in the message to include actual values at the time the event occurs.

! Available Variables

- `$pcode` : Project code
- `$category` : Category
- `$field` : Field
- `$value` : Triggered value
- `$oname` : Event target

Metrics

1. Select a **Category**.
 - This category serves as the baseline for the metric data used to detect anomalies.

2. Select the **Field** to be used for the event trigger conditions. Multiple selections are supported.
3. Select filters that define the event condition targets used to narrow the event scope. Multiple selections are supported.

Anomaly Detection

1. Select the sensitivity levels for **Rising Pattern** and **Falling Pattern**.
2. Use the toggle buttons to enable or disable each pattern.
 - To detect only rising patterns: Rising Pattern **ON**, Falling Pattern **OFF**
 - To detect only falling patterns: Falling Pattern **ON**, Rising Pattern **OFF**

Additional Information

Event Active Time

Configure tags to control when an 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 event 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**.

Edit Event

1. Go to the **Alert > Event Configuration > Anomaly detection** tab.
2. In the **Anomaly Detection** list, click the  icon on the right of the event you want to edit.
3. In the **Anomaly Detection** window, update the options and click the [Save] button.

Delete Event

1. Go to the **Alert > Event Configuration > Anomaly detection** tab.
2. In the **Anomaly Detection** list, click the  icon on the right of the event you want to delete.
3. In the delete confirmation window, click the [Delete] button.

Real-time Log Events

Home > Select Project >  **Alert** > **Event Configuration** > **Real-time Log** tab

You can set alert notifications by filtering collected log data against conditions. Log monitoring must be activated to use log events. For more information on **activating log monitoring**, refer to the log documentation for each product.

- **Real-time log event:** An event is triggered when the configured event conditions are met in real-time collected logs.

Default options

- **Search:** Search for events by selecting an event name or event notification tag
- **JSON download/upload:** Event rules can be directly edited through the JSON editor, or downloaded as a file, modified, and re-uploaded
- **Notification message settings:** Select detailed items for the message to receive and edit the content for each item
- **+ Add event:** Add a new log event

Table | Real-time log event list configuration

Item	Description
Event name	Event name used as the notification title - Up to 255 bytes
Category	Log classification name (log folder name)
Search key	The field to search for in log data - Example: To access an HTTP response status code, use the search key <code>status</code>
Search value	The actual value corresponding to the search key. An alert is sent when logs contain the specified value

Item	Description
	- Example: If the search key is <code>status</code> and the search value is <code>200</code>, an alert is triggered when logs containing HTTP status code 200 are collected
Pause event notifications	Pauses the same event notification for a specified time after an event notification is triggered - Available times: Disabled, 5 min, 10 min, 15 min, 20 min, 30 min, 1 hour, 2 hours, 3 hours, 6 hours, 12 hours, 1 day - If resolved notifications are enabled, no notification is sent for the same event during the selected time after receiving a normal (RECOVERED) status notification
Event target filtering	Conditions that define the targets to which the event applies - If no condition is set, alerts are evaluated for all collected data - Select input: Build conditions by selecting tags, operators, and values - Direct input: Manually enter conditions to specify targets
Event notification tags	User or group (notification tag) to receive the event notification
	Edit and delete events
	Whether the event is active - Enabled: Event occurs when conditions are met - Disabled: Event does not operate and conditions are not checked

Event settings

Log events allow you to configure alert notifications when logs are collected in real time or when logs that meet specific conditions accumulate.

Add real-time log event

1. Click the **Real-time Log** tab in **Alert > Event Configuration**.

2. Click the [+ Add event] button on the right side of the screen.
3. Set the log event conditions in the **Add event rule** window.
 - **Event name:** Enter a name to use as the event title.
 - **Event activation:** When the toggle button is activated, an event occurs when conditions are met.
 - **Level:** Select the event level (**Critical**, **Warning**, **Info**).
 - **Message:** Enter the notification message to receive when the event is triggered. You can use variables in the message to include actual values at the time of event occurrence.

✓ **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.
 - A list of available variables can be checked in **Metrics Query**.
- **Category:** Select the log classification name (log folder name) from the list or enter it directly. You must select a category before entering a search key and search value.
 - **Search key:** Enter the field you want to search for in log data.
 - Example: To access an HTTP response status code, use the search key `status`.
 - **Search value:** Enter the actual value corresponding to the **search key**. An alert is sent when logs contain the specified value.
 - Example: If the **search key** is `status` and the **search value** is `200`, an alert is triggered when logs containing HTTP status code 200 are collected.
 - **Event target filtering:** Enter conditions to define the targets to which the event applies. If no value is entered, alerts are evaluated for all collected data.
 - **Select input:** Select a tag, operator, and value to set the target conditions.

ⓘ For condition expression syntax and supported operators, refer to the [Condition](#)

! settings guide.

- **Direct input:** Enter conditions directly.
- **Pause event notifications:** Select a duration to temporarily suppress alerts. After the event is triggered, the same event will not occur during the selected time.
 - Available times: Disabled , 1 min , 2 min , 3 min , 5 min , 10 min , 20 min , 30 min , 1 hour , 3 hours , 6 hours , 12 hours
- **Event operation time:** Set tags to activate the event during specific time periods (work/non-work/maintenance hours, etc.). If no tag is set, the event operates 24 hours a day when activated.
 1. Click + **Add**.
 2. Click + **Create new tag** in **Event operation time**.
 3. In **Create event operation time tag**, select the **tag name, day of week, time, and color**, then click the [**Apply**] button.
 4. Created tags can be viewed in the tag list and are applied by selecting the checkbox.
 5. Click the  icon for the tag to modify or delete, and edit or delete the tag in **Edit event operation time tag**.

! Deleting a tag removes the tag for all users to whom it has been applied. However, tags in use by event rules cannot be deleted.

- **Event notification tags:** If notification tags are set for an event, notifications are sent to project members with those tags. If no recipient tags are set, notifications are sent to all users in the project.
 1. Click + **Add tag** or +.
 2. Click + **Create new tag** at the bottom of the **Event notification tags** window.
 3. In the **Create tag** window, enter the tag name, select a color, and click the [**Create tag**] button to create the tag.

4. Edit or delete created tags by clicking the  icon in the tag list.
5. Select the desired tag from the **tag list** in the **Event notification tags** window to apply it.

✓ When configuring events, you can send notifications to project members and 3rd-party plugins with the selected event notification tags. You can assign tags to project members and 3rd-party plugins respectively in **Alert notifications > Event notification settings**.

4. When configuration is complete in the **Add event rule** window, click the [Save] button.

Edit/delete events

1. Go to the **Real-time log** tab in **Alert > Event Configuration**.
2. Click the  icon for the event to edit or delete in the log event list.
3. Modify options in the **Event settings** screen and click the [Save] button.
 - a. To delete the selected event, click the [Delete] button in the upper right of the **Event settings** 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

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 Log Event Settings

Home > Select Project >  **Alert** > **Event Configuration** > **Composite Log** tab

You can set alert notifications by filtering collected log data against conditions. Log monitoring must be activated to use log events. For more information on **activating log monitoring**, refer to the log documentation for each product.

- **Composite log event:** An event is triggered when the number of logs that meet the log search conditions within the configured **data query range** reaches or exceeds the threshold count.

Permission

Alert configuration permission is required to configure composite metrics events. Alert configuration permission can be set by clicking the  icon for the member to grant permission in **Home Screen** > **Select Project** > **Management** > **Integrated Member Management**.

Default options

- **Search:** Search for events by selecting an event name or event notification tag
- **JSON download/upload:** Event rules can be directly edited through the JSON editor, or downloaded as a file, modified, and re-uploaded
- **Notification message settings:** Select detailed items for the message to receive and edit the content for each item
- **+ Add event:** Add a new log event

Table | Composite log event list configuration

Item	Description
Event name	Event name used as the notification title
Template	Composite log template

Item	Description
Category	Log classification name (log folder name)
Rule	Conditions that trigger the event
Additional notification when event is resolved	Whether to receive notifications in normal (RECOVERED) status when Critical and Warning level events are resolved - Enabled: Sends a RECOVERED notification when transitioning from in-progress → normal status, and the in-progress/resolved status can be distinguished in event history - Disabled: A one-time event occurs each time the threshold is exceeded
Pause event notifications	Pauses the same event notification for a specified time after an event notification is triggered - Available times: Disabled, 5 min, 10 min, 15 min, 20 min, 30 min, 1 hour, 2 hours, 3 hours, 6 hours, 12 hours, 1 day - If resolved notifications are enabled, no notification is sent for the same event during the selected time after receiving a normal (RECOVERED) status notification
Interval	Time range for querying log data
Event notification tags	User or group (notification tag) to receive the event notification
	Edit and delete events
	Whether the event is active - Enabled: Event occurs when conditions are met - Disabled: Event does not operate and conditions are not checked

Event settings

The composite log event feature triggers alert notifications when the number of logs matching the log search conditions within the configured **data query range** reaches or exceeds the threshold count.

Add composite log event

1. Click the **Composite Log** tab in **Alert > Event Configuration**.
2. Click the [+ Add event] button on the right side of the screen.
3. Set the log event conditions in the **Add event rule** window. For details, refer to [Basic settings](#), [Detail settings](#), and [Additional settings](#).

Add composite log event

4. When configuration is complete in the **Add event rule** window, click the [Save] button.

Basic settings

- **Event name:** Enter a name to use as the event title.
- **Event activation:** When the toggle button is activated, an event occurs when conditions are met.
- **Level:** Select the event level (**Critical**, **Warning**, Info).

Detail settings

Configure [Template \(preset\)](#), [Message](#), [Category](#), [Filter](#), and [Event trigger conditions](#).

Template (preset)

Selecting a predefined preset automatically configures the **filter**, **query type**, and **event trigger conditions**. You can modify the settings after selecting a preset.

Composite log template types		
Preset	Query type	Default rule
None	none	rows > 10
2xx status code count	count	count > 10

Preset	Query type	Default rule
3xx status code count	count	count > 10
4xx status code count	count	count > 10
5xx status code count	count	count > 10
Normal status code (2xx, 3xx) count	count_v2	include_minus_exclude_count > 10
Error status code (4xx, 5xx) count	count_v2	include_minus_exclude_count > 10
Error received count	count	count > 10
Exception received count	count	count > 10
Log count per second	rps	rps > 100
Field aggregation (Aggregation)	aggr	avg > 1000
Rate	rate	rate > 10
Unique value count (Cardinality)	cardinality	cardinality > 100
Percentile	percentile	p99 > 3000

Message

Enter the notification message to receive when an event occurs. You can use variables in the message to compose dynamic notifications.

- **Example:** Message `${host} error ${count} occurred` → Notification message web-01 error 15 occurred

Table | Message variables

Variable	Description	Usage condition
<code>\${count}</code>	Number of logs matching the condition	Always available
<code>\${groupField}</code>	Grouping field name	Grouped notifications only
<code>\${groupValue}</code>	Grouping field value	Grouped notifications only
<code>\${fieldName}</code>	Grouping field value (referenced directly by field name)	Grouped notifications only

ⓘ The `${count}` variable can be used in all notifications regardless of grouping. The `${groupField}` , `${groupValue}` , and `${fieldName}` variables are only substituted when a grouping field is configured.

Category

Select the log classification name (log folder name) from the list or enter it directly. You can enter the search key and search value only after selecting a category.

Filter

Enter log search conditions using Lucene query syntax. Two input modes are supported.

- **Filter mode** : Build a query by selecting fields and values.
- **Code mode** : Enter Lucene queries directly.

Table | Lucene query examples

Query	Description
<code>level:ERROR</code>	Logs where the level field is ERROR
<code>status_nxx:status_4xx OR status_nxx:status_5xx</code>	4xx or 5xx status codes
<code>appender:accessLog</code>	Logs from the accessLog appender
<code>response_time.n:*</code>	Logs where the response_time.n field exists

Query	Description
host:web-01 AND level:ERROR	Error logs from host web-01

Event trigger conditions

Configure by query type. Additional settings are displayed depending on the selected template (preset) or query type.

Table | Event trigger condition key meanings

Key	Meaning
rows / count	Number of logs matching the filter condition
rps	Log count per second (count / interval in seconds)
avg	Average value of the target field
sum	Sum of the target field values
min	Minimum value of the target field
max	Maximum value of the target field
rate	Numerator count / denominator count × 100 (%)
cardinality	Number of unique values in the target field
p99 / p95 / p90 / p75 / p50	Percentile values of the target field
include_minus_exclude_count	Include filter count – exclude filter count

- **Count (count / none):** Counts the number of logs matching the configured filter conditions.

Table | Count settings

Setting	Description
Filter	Log search condition (Lucene query)
Event trigger condition	<code>count</code> or <code>matching logs</code> <code>{operator}</code> <code>{threshold}</code> (e.g., <code>count > 10</code>)

- **Log count per second (rps):** Calculates the number of logs per second (Requests Per Second) matching the configured filter conditions. The total count during the interval is divided by the interval in seconds.

Table | Log count per second settings

Setting	Description
Filter	Log search condition (Lucene query)
Event trigger condition	<code>RPS</code> <code>{operator}</code> <code>{threshold}</code> (e.g., <code>rps > 0.5</code>)

- **Field aggregation (aggr):** Calculates the aggregated value (average, sum, min, max) of the specified numeric field. The aggregation method can be selected from a dropdown in the event trigger condition.

Table | Field aggregation settings

Setting	Description
Filter	Log search condition (Lucene query)
Target field	Numeric field to aggregate (e.g., <code>response_time.n</code>)
Event trigger condition	<code>avg / sum / min / max</code> <code>{operator}</code> <code>{threshold}</code> (e.g., <code>avg > 1000</code>)

- **Rate:** Calculates the percentage (%) ratio of the numerator filter count to the denominator filter count. Useful for monitoring the ratio of specific conditions against all logs.

Table | Rate settings

Setting	Description
Filter (numerator)	Log condition for the numerator (e.g., <code>level:ERROR</code>)
Denominator filter	Log condition for the denominator (e.g., <code>level:*</code>)
Event trigger condition	<code>Rate (%) {operator} {threshold}</code> (e.g., <code>rate > 10</code>)

- **Example:** Filter `level:ERROR`, denominator filter `level:*`, rule `rate > 10` → Alert triggered when the error rate among all logs exceeds 10%
- **Unique value count (cardinality):** Counts the number of unique values in the specified field. Useful for detecting an abnormally large variety of values.

Table | Unique value count settings

Setting	Description
Filter	Log search condition (Lucene query)
Target field	Field to calculate unique values for (e.g., <code>host</code>)
Event trigger condition	<code>Unique value count {operator} {threshold}</code> (e.g., <code>cardinality > 100</code>)

- **Percentile:** Calculates the percentile value of the specified numeric field. Useful for monitoring tail latency in response times. Changing the percentile selection automatically updates the key in the event trigger condition.

Table | Percentile settings

Setting	Description
Filter	Log search condition (Lucene query)
Target field	Numeric field to calculate percentile for (e.g., <code>response_time.n</code>)
Percentile	Select from P99, P95, P90, P75, P50

Setting	Description
Event trigger condition	P99 / P95 / ... {operator} {threshold} (e.g., p99 > 3000)

Table | Percentile meanings

Percentile	Meaning
P50	Median. 50% of all data is at or below this value
P75	75% of all data is at or below this value
P90	90% of all data is at or below this value
P95	95% of all data is at or below this value
P99	99% of all data is at or below this value

- **Include/exclude count (count_v2):** Calculates the net count by subtracting the exclude filter count from the include filter count.

Table | Include/exclude count settings

Setting	Description
Filter (include)	Log conditions to include (e.g., status_nxx:status_4xx OR status_nxx:status_5xx)
Filter exclude	Log conditions to exclude (e.g., status_nxx:status_4xx)
Denominator filter	Denominator for ratio calculation (optional)
Event trigger condition	Include-exclude count {operator} {threshold} (e.g., include_minus_exclude_count > 10)

- **Example:** Filter status_nxx:status_4xx OR status_nxx:status_5xx , filter exclude status_nxx:status_4xx → Alert triggered when the net 5xx count (4xx+5xx minus 4xx) exceeds 10

Group by field (Group By)

When a group by field is configured, alert conditions are evaluated independently for each value of that field. Select the field to group by from the dropdown. Clicking the dropdown shows a list of available fields for the current category with autocomplete.

- **Example:** Filter `level:ERROR` , group by field `host` , rule `rows > 3`

Table | Grouping comparison

Item	Without grouping	With grouping
Evaluation unit	All logs aggregated as one	Independent aggregation per group value
Alert occurrence	1 alert	One per group meeting the condition
Alert title	Event title	[Group value] Event title
Stateful	State transition based on all logs	Independent state transition per group

- `[web-01] Error alert` — 8 errors from web-01 → alert triggered
- `[web-02] Error alert` — 5 errors from web-02 → alert triggered
- `[api-01] Error alert` — 2 errors from api-01 → condition not met, no alert

❗ When **Additional notification when event is resolved** is enabled, the ABNORMAL → RECOVERED state transition occurs independently per group. For example, when errors from `web-01` are resolved, a `[web-01] RECOVERED` notification is triggered, with no effect on other hosts.

Additional settings

- **Interval:** Check alert conditions at the selected time interval (`unset` , `1 min` , `5 min` , `10 min` , `30 min` , `1 hour` , `12 hours`).
- **Mute:** After an event occurs, the same event is not triggered during the selected time (`unset` , `1 min` , `5 min` , `10 min` , `30 min` , `1 hour` , `12 hours`).

- **Event operation time:** Set tags to activate the event during specific time periods (work/non-work/maintenance hours, etc.). If no tag is set, the event operates 24 hours a day when activated.
 1. Click **+ Add**.
 2. Click **+ Create new tag** in **Event operation time**.
 3. In **Create event operation time tag**, select the **tag name**, **day of week**, **time**, and **color**, then click the **[Apply]** button.
 4. Created tags can be viewed in the tag list and are applied by selecting the checkbox.
 5. Click the  icon for the tag to modify or delete, and edit or delete the tag in **Edit event operation time tag**.

 Deleting a tag removes the tag for all users to whom it has been applied. However, tags in use by event rules cannot be deleted.

- **Event notification tags:** If notification tags are set for an event, notifications are sent to project members with those tags. If no recipient tags are set, notifications are sent to all users in the project.
 1. Click **+ Add tag** or **+**.
 2. Click **+ Create new tag** at the bottom of the **Event notification tags** window.
 3. In the **Create tag** window, enter the tag name, select a color, and click the **[Create tag]** button to create the tag.
 4. Edit or delete created tags by clicking the  icon in the tag list.
 5. Select the desired tag from the **tag list** in the **Event notification tags** window to apply it.

 When configuring events, you can send notifications to project members and 3rd-party plugins with the selected event notification tags. You can assign tags to project members and 3rd-party plugins respectively in **Alert notifications > Event notification settings**.

Simulation

You can simulate using historical data before saving the alert to check the expected number of alert occurrences.

1. After completing the alert configuration, select a time range at the bottom.
2. Click the [Simulation] button.
3. The chart displays data by time period and the number of alert occurrences.

! You can adjust the simulation time range to check alert occurrence patterns across different time periods. Data from the last 1–2 minutes may not be available due to index build delays.

4. When configuration is complete in the **Add event rule** window, click the [Save] button.

Create alert from Log Explorer

You can create alerts directly from Log Explorer.

1. Navigate to **Log > Log Search** menu.
2. Configure the category and search query.
3. Click the [Add event] button in the upper right of the screen.
4. The alert configuration form opens in a drawer, with the current category and search query automatically filled in.
5. Complete additional settings and click the [Save] button.
6. After saving, click the [Go] button to navigate to the alert list.

Edit/delete events

1. Go to the **Composite log** tab in **Alert > Event Configuration**.
2. Click the  icon for the event to edit or delete in the log event list.
3. Modify options in the **Event settings** screen and click the [Save] button.
 - a. To delete the selected event, click the [Delete] button in the upper right of the **Event settings** 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

Database

Home > Select project >  **Alert** > **Event Configuration** classic

Alert notifications are sent by configuring event conditions for each database metric. By entering threshold values per metric, notifications can be sent at **Critical** and **Warning** levels.

Default options

You can configure event conditions based on key metrics of the database instance.

- **Instance:** The database agent for which event conditions are configured
- **Database:** The target database to monitor
- **Inactive agent alert:** Sends an alert when no response is received from the agent after the configured time

Table | Database event configuration screen

Item	Description
Metrics	Database metrics used to define event conditions
Critical	Threshold for the Critical level
Warning	Threshold for the Warning level
Detection count	The number of consecutive times the metric exceeds the threshold while being collected every 5 seconds – Example: If set to 3, an alert is triggered when the condition is met for 5 seconds × 3 consecutive times
Send notification when event is resolved	Sends an additional notification when the event state is recovered after occurring – Supported in DBX agent version 1.6.30 or later
	Delete event

Item	Description
	Enable or disable the event - Enabled: An event occurs when conditions are met - Disabled: The event does not operate and conditions are not evaluated

Add event

Based on the selected **Metrics** item, you can configure alerts to be sent when the entered **Critical** and **Warning** conditions are met for the specified **Detection count**.

1. Navigate to **Alert > Event Configuration** *classic*.
2. Select the agent for which you want to configure alerts under **Instance**.
3. Click the [+ Add event] button.
4. Select the desired metric from **Metrics**.
5. Choose a comparison operator (**<**, **≤**, **>**, **≥**) for **Critical** and **Warning**, then enter values.
 - An alert is triggered when the configured value is exceeded.
6. Enter the desired value for **Detection count**.
 - An alert is triggered when the condition occurs consecutively at 5 second intervals for the specified count.
 - Example: If the detection count is 3
 5 seconds (1st) - 5 seconds (2nd) - 5 seconds (3rd) → Total 15 seconds → Alert sent
7. If **Send notification when event is resolved** is enabled, an additional notification is sent when the event state is recovered (RECOVERED).

 The **Send notification when event is resolved** feature is supported in **DBX agent version 1.6.30 or later**.
8. Enable the toggle button on the far right () to send alert notifications.
 - a. To delete an event, click the  icon for that event.

Copy alerts

You can copy alert events created for the currently selected instance to other instances.

1. Navigate to **Alert > Event Configuration** *classic*.
2. Click the [Copy alerts] button.
3. In the **Copy alerts** window, select the target instances. To copy to all instances, select **Select all**.
4. Click the [Copy] button to duplicate the alerts.

 If you use the **Copy alerts** feature when the target instance already has alert events configured, the existing alert events will be deleted.

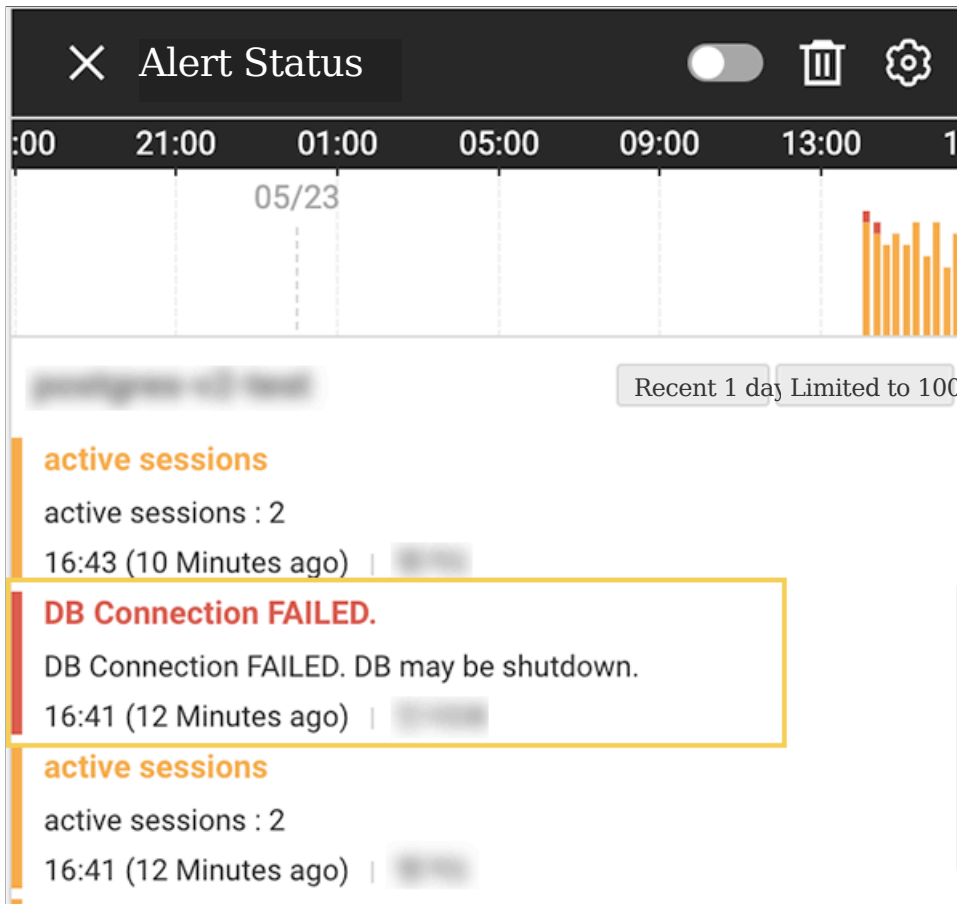
Inactive agent alert

An alert is sent when no response is received from the agent after the configured time.

1. Navigate to **Alert > Event Configuration** *classic*.
2. Under **Inactive agent alert**, select the desired time (30 seconds , 1 minute , 2 minutes , 3 minutes , 5 minutes , 10 minutes , 30 minutes).
3. Click the toggle button on the right to enable it.
 -  **Enabled:** If no data is received from the agent for the configured time, an **INACTIVE MONITORING AGENT** event occurs.

Default alert: Connection FAILED

Even without configuring event rules, an alert is sent when the agent fails to connect to the database 15 times or more. This may occur due to database shutdowns, network issues, or account or permission problems.



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

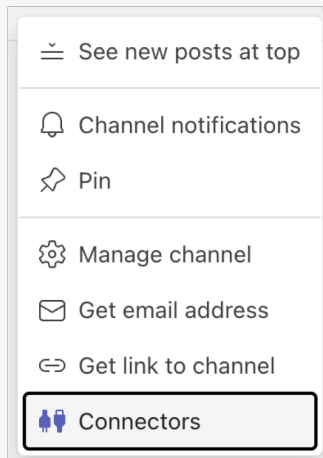


- `/remove` : Enter the project code for which you want to stop receiving alerts
- `/list` : View the list of project codes currently receiving alerts

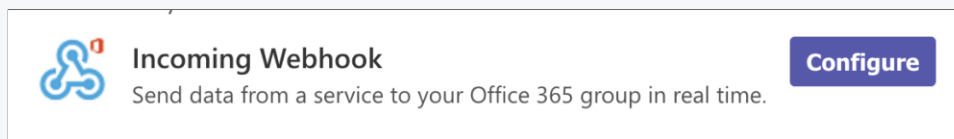
Teams



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

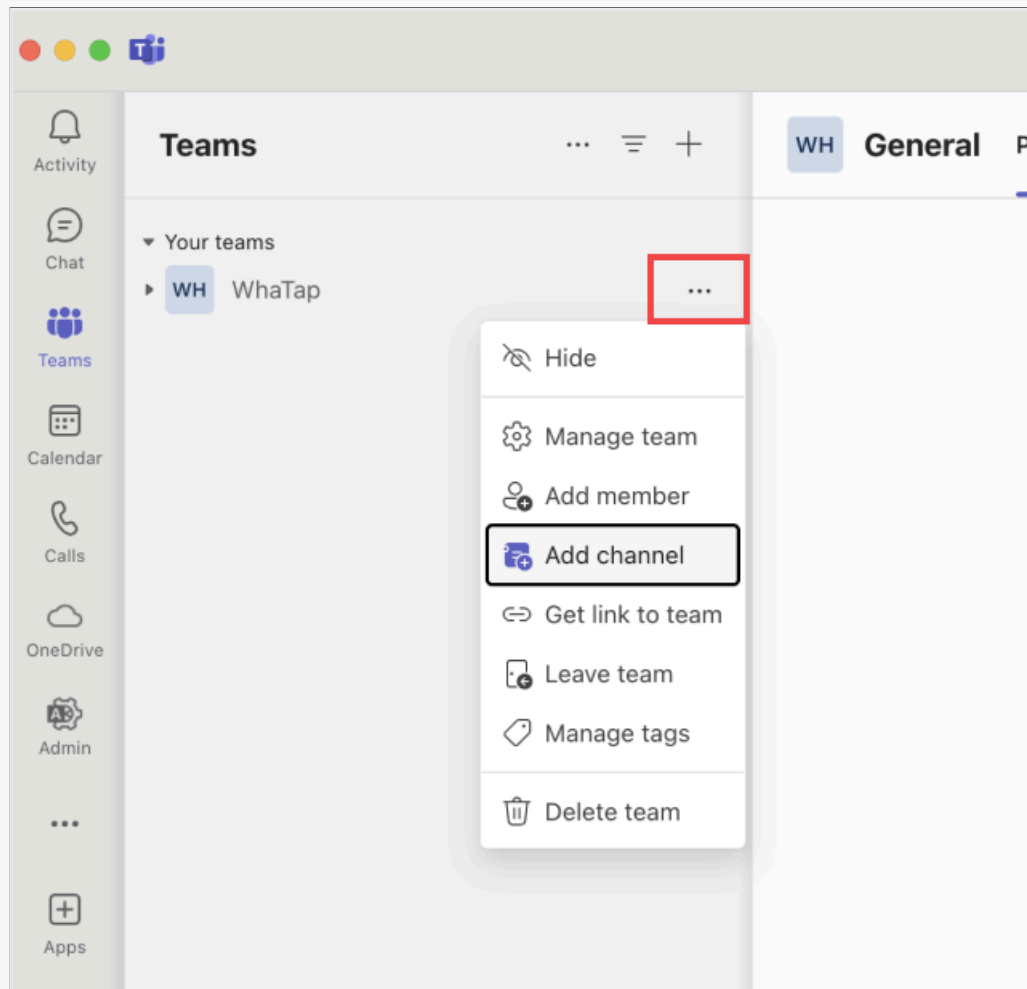
5. Return to the **Add a third-party plug-in** window in the WhaTap Monitoring service, and enter values for **Channel Name** and **Webhook URL** at the bottom of the **Teams** tab.
6. Click the [**Registration**] button.

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 * ⓘ

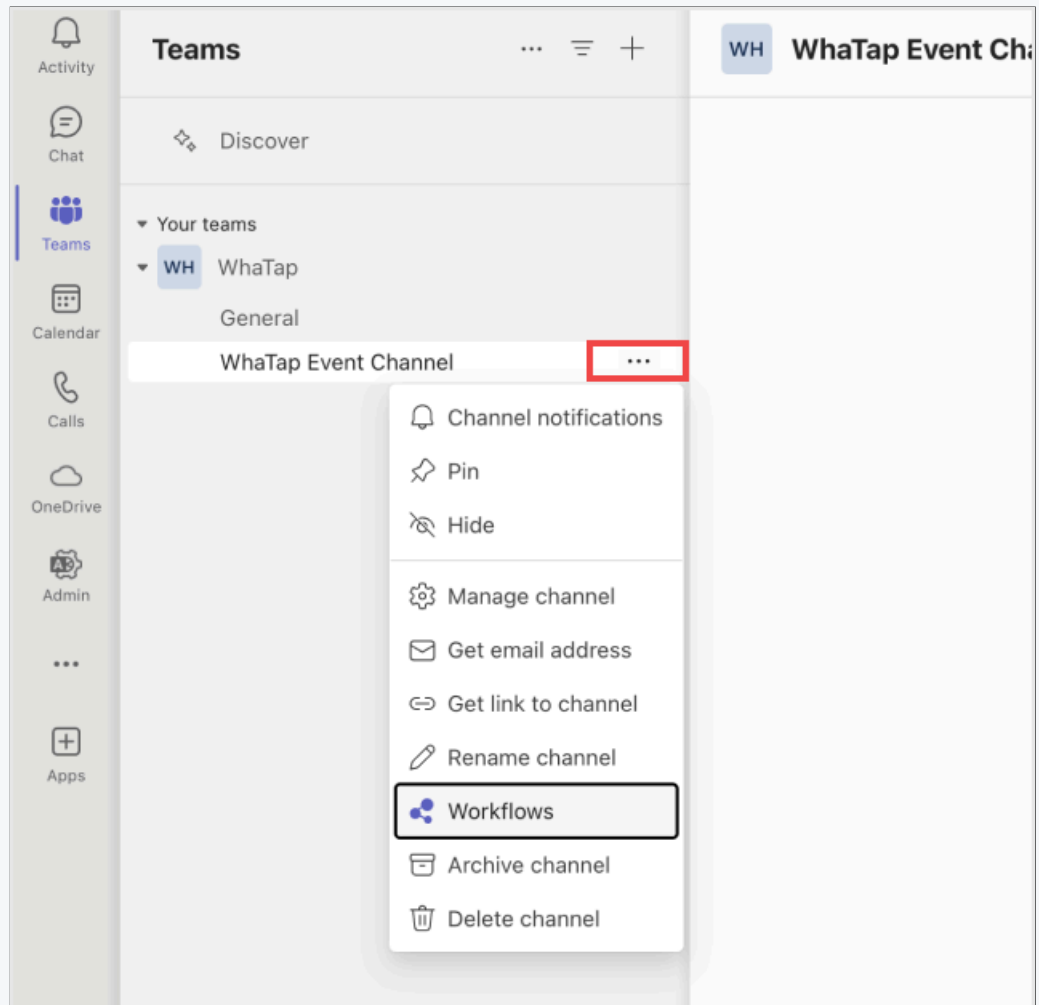
Select

- 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.

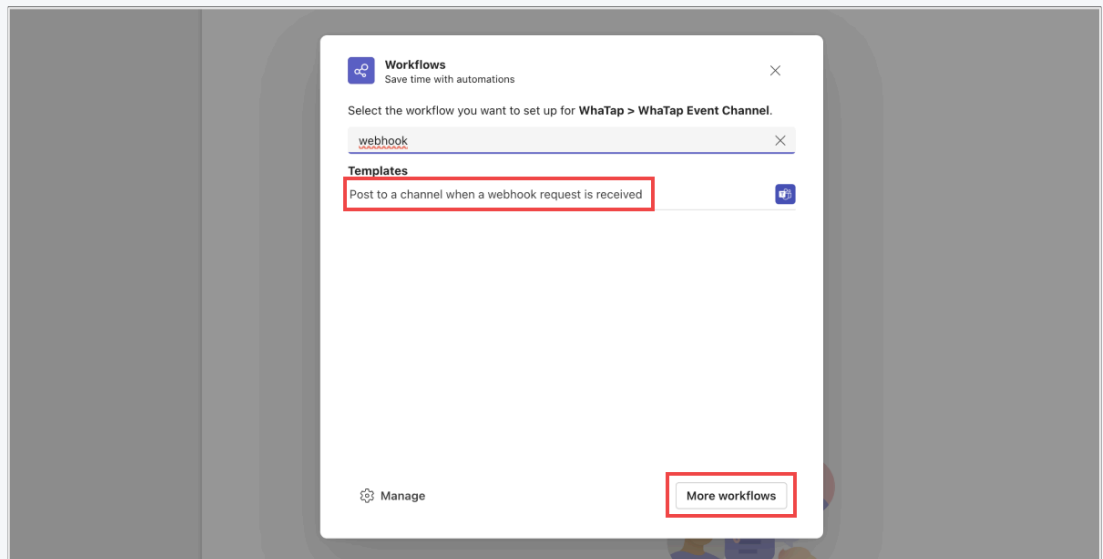
Cancel Create

ⓘ 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.

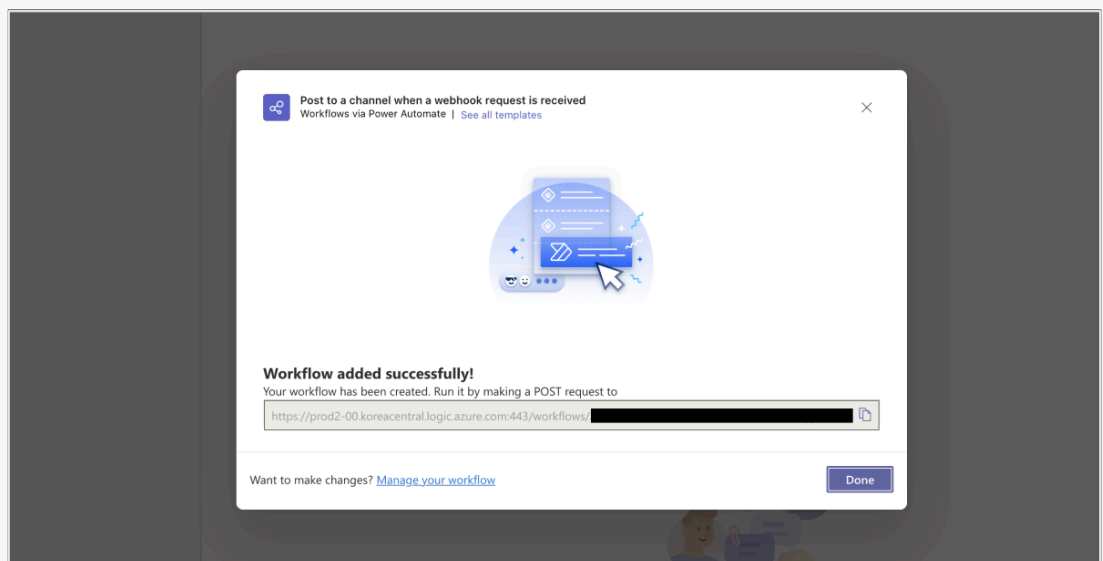
3. Click **More actions (...)** of the created channel, then select **Workflows**.



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.



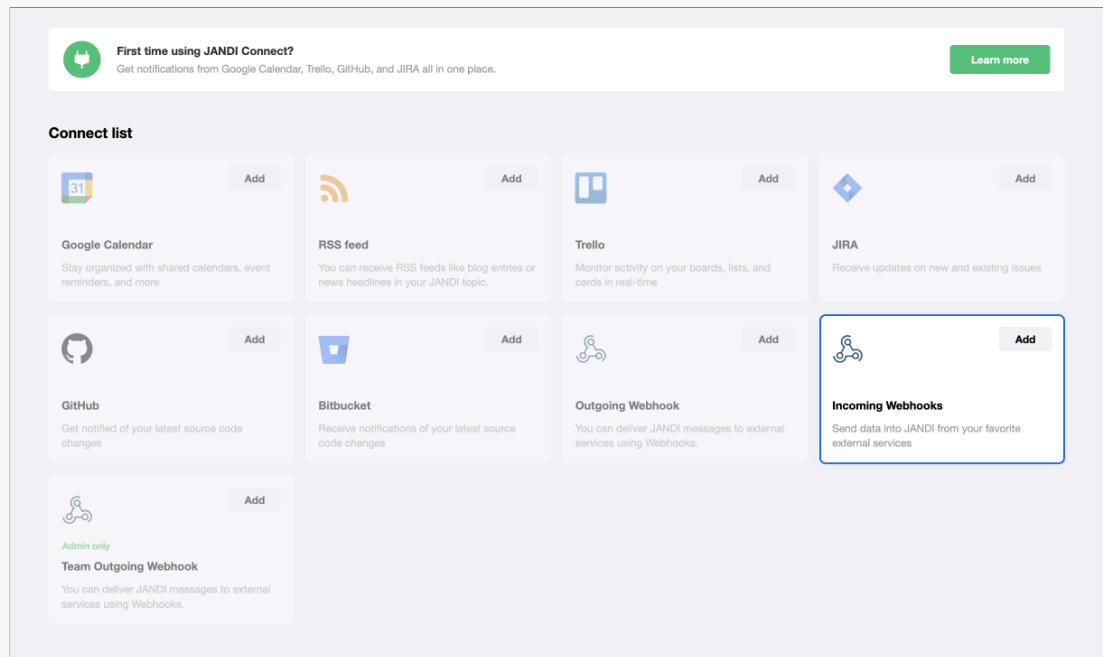
9. Click the [**Registration**] button.



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.

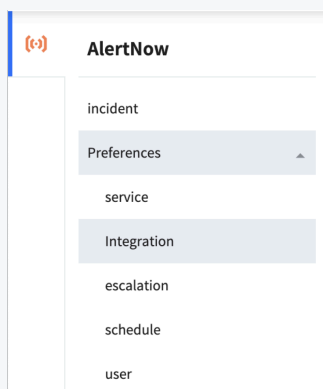
<https://wh.jandi.com/connect-api/webhook/31257583/d3...>
Copy

- 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}",
}
```



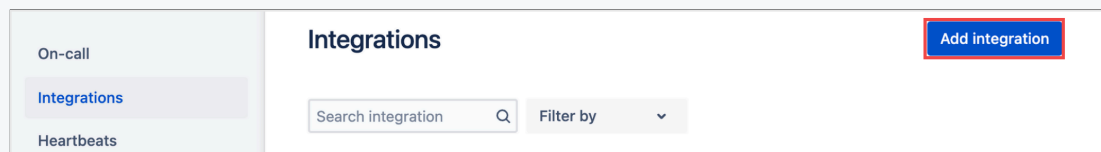
```
"metricName": "${metricName}",
"metricValue": "${metricValue}",
"metricThreshold": "${metricThreshold}",
"uuid": "${uuid}",
"okind": "${okind}",
"onode": "${onode}"
}
```

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.

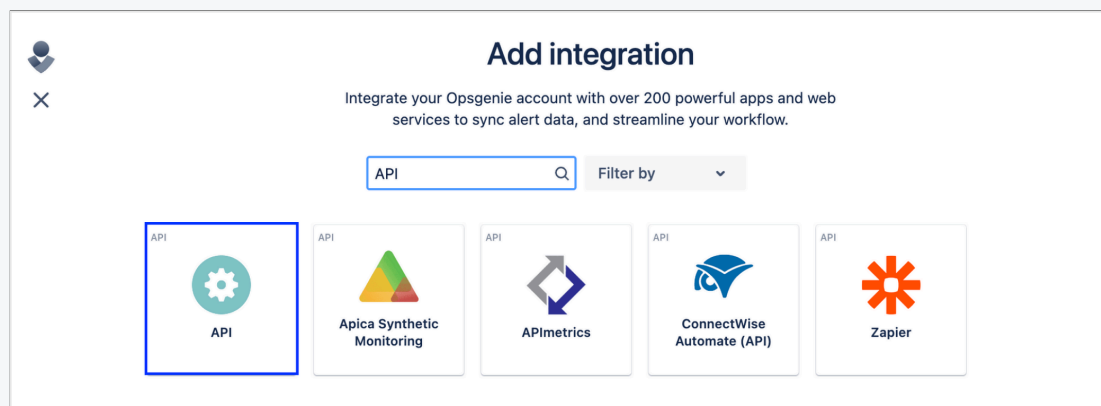
Opsgenie



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

- 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.

- Go to **Alert > Event History** and click the [CSV] button.
- 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.

- Go to **Alert > Event History** and click the [View Only Events in Progress] button.
- 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.

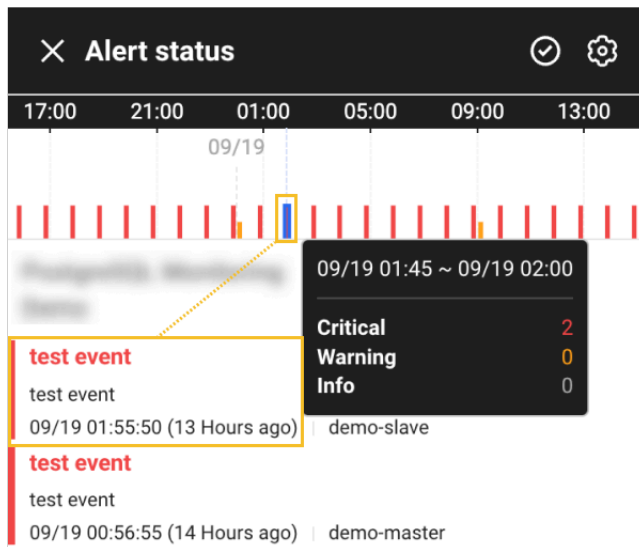
Viewing Real-time Notifications

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.

Correlated Project Management

Home > Select Project > **Management** > **Correlated Project Management**

If you have created a database monitoring (DPM) project and an application monitoring (APM) project, you can relate the two projects for monitoring the data collected from DPM in the APM project.

You can check the details of the transaction in real time through active transactions of the application linked to the active session in the database. This integrated monitoring allows you to analyze the interactions between applications and databases in detail and diagnose and resolve performance bottlenecks in real time.

Correlated Project Management

Add a project to correlate
Select the project you want to correlate, then click the add button. (Supported types: Server, MySQL, PostgreSQL, Oracle, Oracle Pro, Redis, MSSQL, Altibase, MongoDB, Cubrid, Java, Kubernetes, Namespace)

Correlated project list

pcode	Project name	Group	Type	
2994	petclinic-Postgresql-apm		Java	

The **Correlated Project Management** feature allows you to go beyond product-/equipment-centric monitoring and view monitoring data across multiple projects. This is useful for analyzing how different system components such as applications and databases, interact together. In particular, when performance degradation occurs, you can quickly determine whether it is a hardware problem or an individual application problem, which can significantly reduce the troubleshooting time. Through **Correlated Project Management**, you can comprehensively analyze performance data at the user system level and obtain more visible insights.

Through the following video clip, see the content details.

ⓘ Notes before you begin

- Currently, only the **Java** platform is supported. The supported platforms are to be expanded through upcoming updates.
- The **Correlated DB session** feature requires Java agent 2.2.33 or later.
- For more information on how to link a database project to the application project, see [the following](#).

Adding any linked project

1. Go to **Management > Correlated Project Management**.
2. In the **Add a project to correlate** section, select the **Project selection** input field.

Add a project to correlate

Select the project you want to correlate, then click the add button. (Supported types: Server, MySQL, PostgreSQL, Oracle, Oracle Pro, Redis, MSSQL, Altibase, Mong)

Project selection Add

☐ OTE-APM@Front	Java	13	OTE-SELF
☐ DB-REDIS-JINS-TEST	Redis	19	Jinslab
☐ DB-MARIADB-JINS-TEST	MySQL	20	Jinslab
☐ hsnam-ecs-1	Kubernetes	29	
☐ hsnam & vm	Server	38	
☐ DB-POSTGRESQL-JINS-TEST	PostgreSQL	60	Jinslab
☐ SQLServer-V1	MSSQL	82	
☐ rum_previous	Java	88	swlim
☐ java-apm-docker	Java	181	bspark-test

3. If you see a list of projects, select a project to link to. You can also enter a string to search for matching projects.
4. Select one or more projects and then select **Add** that is active.

The selected project is added in **Correlated project list**.

! You need the **Edit** role on the project to add a linked project. If you have the **Edit** role, the **Add** button is enabled in the **Correlated Project Management** menu.

Checking any linked project

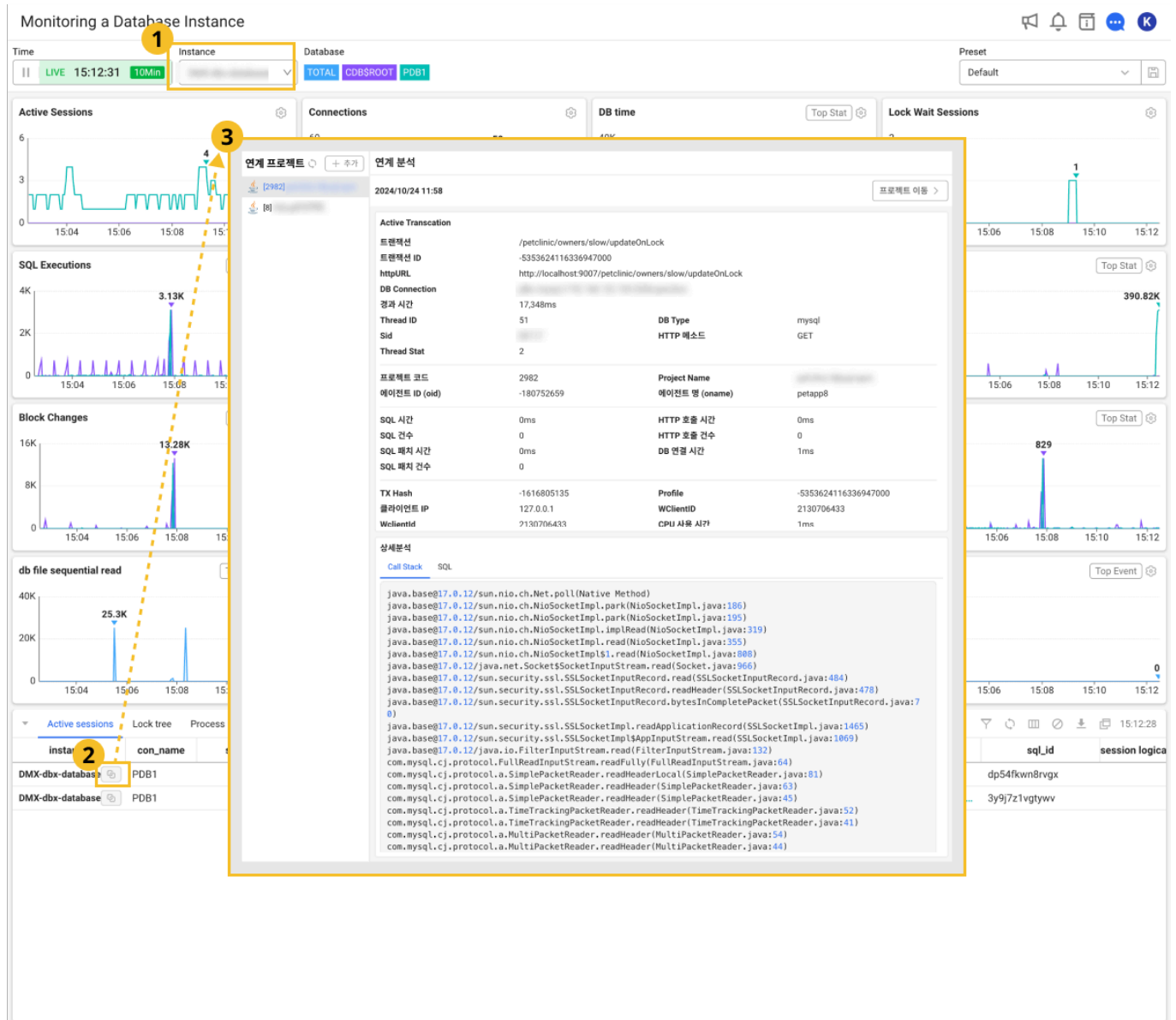
It provides real-time details about active transactions in applications associated to active sessions in the database. Users can check specific metrics such as transaction ID, thread ID, client IP, and CPU usage time, along with basic information such as transaction URL and database connection information, DB type, HTTP method, and execution time. Additionally, you can monitor the SQL query execution time, number of SQL calls, DB connection time, and such, which allows you to comprehensively analyze the system performance.

In particular, it provides the call stack information and SQL queries through stack traces, helping you trace and

resolve detailed causes when problems occur. This allows you to perform in-depth analysis of interactions between applications and databases, and to quickly diagnose performance bottlenecks.

Tracing active transactions

Go to **Dashboard > Monitoring a Database Instance**. The link analysis feature can be checked only in real-time query status.



1. Select an instance (agent) for query in the **Instance** option.
2. In the **Active sessions** table list, select  for the active session to correlate.
3. The **Correlation analysis** window appears, which provides transaction information associated with the selected active session.

Select the desired project from the **Correlated project** list to check the linked analysis information.

- You can go to the **Application Dashboard** menu for that project by selecting **Move project** on the upper right of the screen.
- If there is no transaction associated with the selected active session, the screen displays **No data**.

Correlation analysis Screen guidance

The **Correlation analysis** window displays information details for active transactions, as well as the call stack and SQL statements.

Active Transaction

You can check information details of the linked active transactions.

Active Transaction			
Transaction	/petclinic/owners/slow/updateOnLock		
Transaction ID	4362591315698310000		
httpURL	http://localhost:9007/petclinic/owners/slow/updateOnLock		
DB Connection	jdbc:mysql://127.0.0.1:3306/petclinic		
Elapsed time	12,950ms		
Thread ID	49530	DB Type	MySQL
Sid	49530	HTTP method	GET
Thread Stat	RUNNABLE		
Project code	2982	Project Name	petclinic-slow
Agent ID (oid)	-180752659	Agent name (oname)	petapp8
SQL time	0ms	HTTP call time	0ms
SQL count	0	HTTP call count	0
SQL patch time	0ms	DB CONN. time	2ms
SQL fetch count	0		
TX Hash	-1616805135	Profile	4362591315698310000
Client IP	127.0.0.1	WClientID	2130706433
Wclientid	2130706433	CPU usage time	2ms

- **DB Connection:** Database connection URL.
- **Thread ID:** Unique identifier of the thread used to process the transaction.
- **Sid:** ID of the selected active session.
- **Thread Stat:** Current status of the thread.
- **DB Type*:** Type of the linked database platform.

- ⓘ • In the versions before Java Agent 2.2.40, the **ThreadStat** value appears in numeric format. To display in text format, install [Java Agent 2.2.40](#) or later in the related projects.
- For more information about others, see [the following](#).

Call Stack

It provides information to trace the order of methods or function calls.

More Information

[Call Stack](#)
[SQL](#)

```
java.base@17.0.12/sun.nio.ch.Net.poll(Native Method)
java.base@17.0.12/sun.nio.ch.NioSocketImpl.park(NioSocketImpl.java:186)
java.base@17.0.12/sun.nio.ch.NioSocketImpl.park(NioSocketImpl.java:195)
java.base@17.0.12/sun.nio.ch.NioSocketImpl.implRead(NioSocketImpl.java:319)
java.base@17.0.12/sun.nio.ch.NioSocketImpl.read(NioSocketImpl.java:355)
java.base@17.0.12/sun.nio.ch.NioSocketImpl$1.read(NioSocketImpl.java:808)
java.base@17.0.12/java.net.Socket$SocketInputStream.read(Socket.java:966)
java.base@17.0.12/sun.security.ssl.SSLSocketInputRecord.read(SSLSocketInputRecord.java:484)
java.base@17.0.12/sun.security.ssl.SSLSocketInputRecord.readHeader(SSLSocketInputRecord.java:478)
java.base@17.0.12/sun.security.ssl.SSLSocketInputRecord.bytesInCompletePacket(SSLSocketInputRecord.java:70)
java.base@17.0.12/sun.security.ssl.SSLSocketImpl.readApplicationRecord(SSLSocketImpl.java:1465)
java.base@17.0.12/sun.security.ssl.SSLSocketImpl$AppInputStream.read(SSLSocketImpl.java:1069)
java.base@17.0.12/java.io.FilterInputStream.read(FilterInputStream.java:132)
com.mysql.cj.protocol.FullReadInputStream.readFully(FullReadInputStream.java:64)
com.mysql.cj.protocol.a.SimplePacketReader.readHeaderLocal(SimplePacketReader.java:81)
com.mysql.cj.protocol.a.SimplePacketReader.readHeader(SimplePacketReader.java:63)
com.mysql.cj.protocol.a.SimplePacketReader.readHeader(SimplePacketReader.java:45)
com.mysql.cj.protocol.a.TimeTrackingPacketReader.readHeader(TimeTrackingPacketReader.java:52)
com.mysql.cj.protocol.a.TimeTrackingPacketReader.readHeader(TimeTrackingPacketReader.java:41)
com.mysql.cj.protocol.a.MultiPacketReader.readHeader(MultiPacketReader.java:54)
com.mysql.cj.protocol.a.MultiPacketReader.readHeader(MultiPacketReader.java:44)
```

SQL

SQL query executed via the linked active transactions.

More Information

[Call Stack](#)
[SQL](#)

SQL format



```
select o1_0.id,o1_0.address,o1_0.city,o1_0.first_name,o1_0.last_name,p1_0.owner_id,p1_0.id,p1_0.birth_date,p1_0.name,p1_0.type_id,o1_0.telephone from owners o1_0 left join pets p1_0 on o1_0.id=p1_0.owner_id where o1_0.id=? order by p1_0.name
```

Removing any linked project

1. Go to **Management > Correlated Project Management**.
2. To delete a project to unlink from the **Correlated project list** section, select  on the utmost right of

the list.

3. When the confirmation message appears, select **Delete**.

The deleted project is excluded from the **Correlated project list** section.

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](#).

 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.

OpenMetrics Explorer

OpenMetrics Explorer is a feature that allows you to directly explore custom metrics based on Prometheus using PromQL, visualize them in various formats, and check metric states or gain insights.

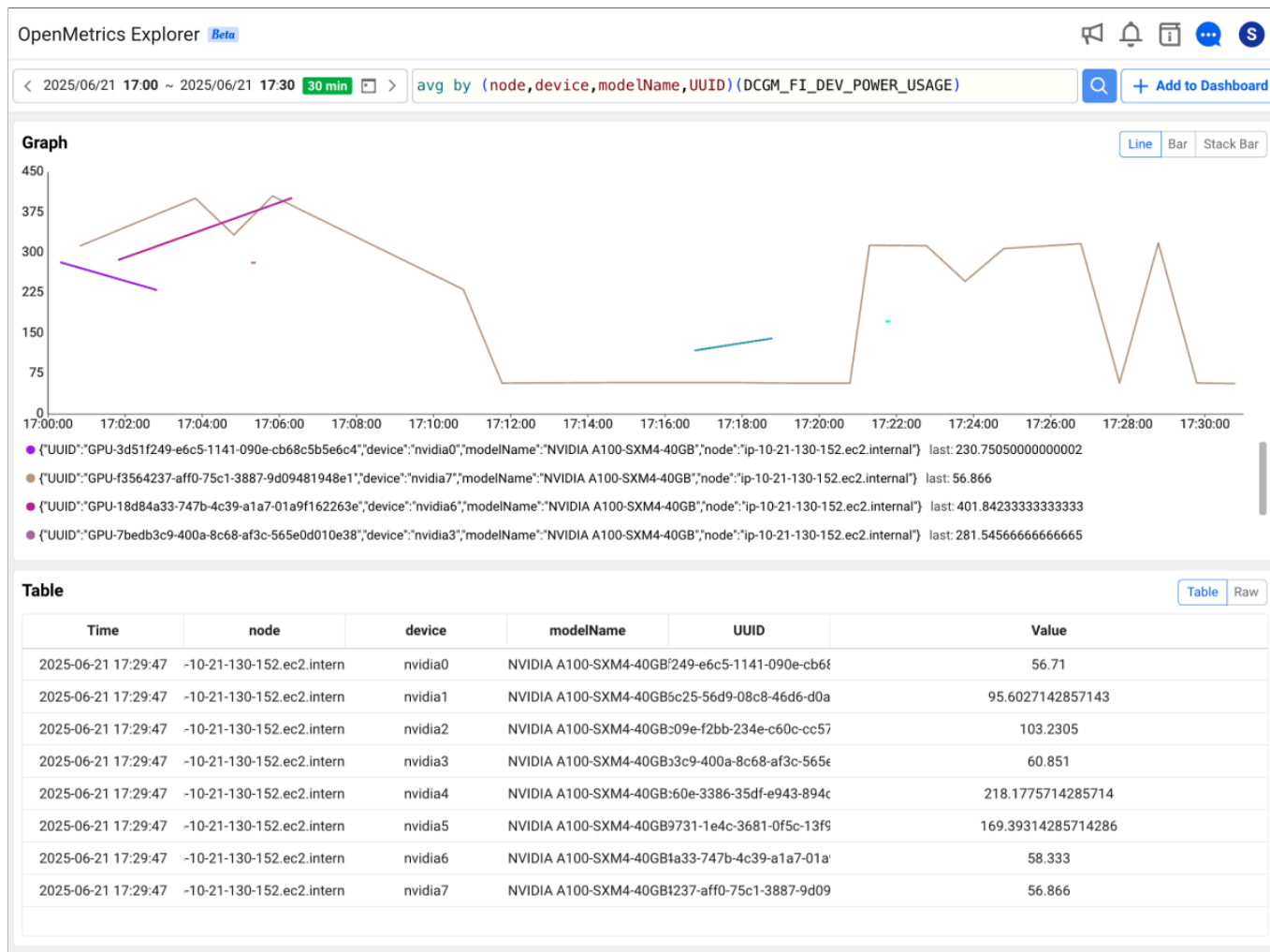
With this feature, you can freely query the metrics you need, extract meaningful data, or connect them to dashboards for continuous monitoring.

ⓘ To use OpenMetrics Explorer, you must install the Open Agent.

- See [Install WhaTap Operator](#).

Exploring with PromQL

In **OpenMetrics Explorer**, you can run PromQL queries and view the results with different visualization options.



Graph

- **Line:** Default time-series line chart
- **Bar:** Visualizes values as a bar chart
- **StackBar:** Stacked bar chart that shows accumulated values

Table / Raw

- **Table:** Displays time-series values in chronological order as a table
- **Raw:** View the Prometheus API response directly in JSON format

PromQL Syntax Support and Autocomplete

When you enter metrics such as `node_cpu_seconds_total` or `container_memory_usage_bytes`, the autocomplete feature suggests available metrics and operators.

It also supports label filtering in the form `{pod="example"}`.

Adding to Dashboard

You can save a queried PromQL as a dashboard widget and continuously monitor the desired metrics.

How to Add

1. In the **OpenMetrics Explorer** screen, click the **+ Add to Dashboard** button in the upper right corner.
2. In the **Add to Dashboard** window, select an existing dashboard or create a new one.
3. Click **Apply** to add the widget to the chosen dashboard.
 - You can enter a widget name and description in the added dashboard.
 - Added widgets can be freely repositioned in the Flexboard grid layout and edited with PromQL.