

PostgreSQL Monitoring

release date. 2026. 09. 11.

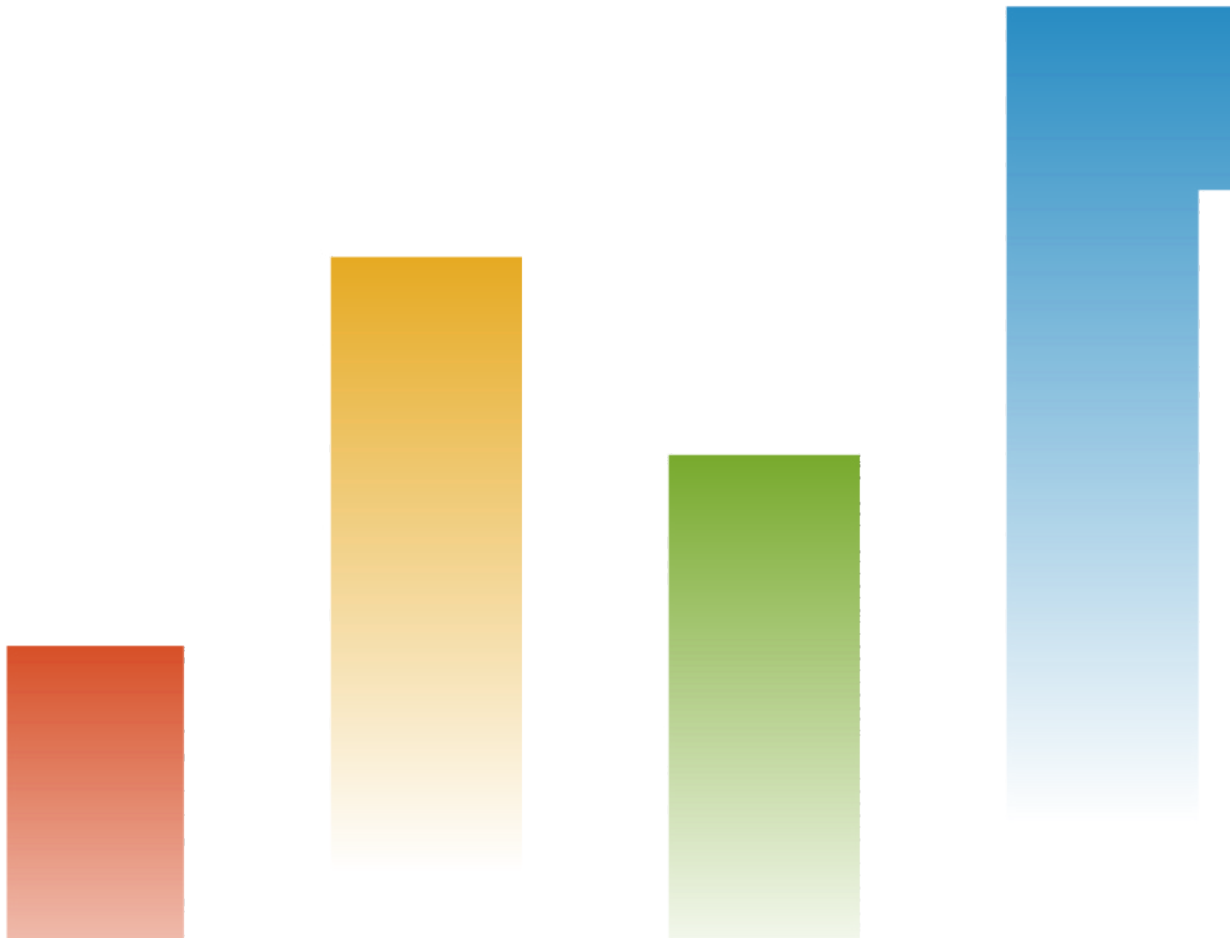


Table Of Contents

• PostgreSQL Monitoring	5
◦ Getting started	10
◦ Supported Environment	12
◦ Installation	
■ Agent Installation	
■ Manual Installation	19
■ Package Manager Installation	32
■ Docker Installation	39
■ Installation checking items	46
■ Troubleshooting	48
◦ Configuration	51
■ Quick agent configuration	54
■ DBX Agent Setting	58
■ XOS agent configuration	68
■ Cloud Agent Configuration	73
■ Installation and Usage	86
■ Log Collection Settings	92
■ Serverless Agent Settings	95
■ Serverless Log Collection Settings	114
◦ Management	121
◦ Dashboard	125
■ Instance List	127
■ Instance Map	130
■ Topology	138

■ Monitoring a Database Instance.....	141
■ Monitoring Multiple Instances	169
■ Slow Query	194
■ Flex Board	200
■ Using Flex board	203
■ Managing the widgets	213
○ Analysis	
■ Comparison between count trends	218
■ Lock tree.....	226
■ Session History	234
■ Wait analysis	242
■ Vacuum Analysis.....	246
■ Top Object	257
■ Top SQL Comparison	267
■ Database Parameter	275
○ Metrics.....	277
■ PostgreSQL metrics	279
■ Metrics Chart	287
■ Metrics Chart	310
■ Metrics Search	313
■ Metric Anomaly Detection	315
■ Metrics Trend Analysis.....	318
○ Stat/Report	
■ Top SQL.....	321
■ PG SQL Statistics.....	329
■ Database size.....	338

■ Report	344
◦ Log	347
■ Application	349
■ Live Tail	351
■ Log Explorer	364
■ Log Trend	371
■ Log Search	378
■ Log setting	393
■ Parsing the logs	428
◦ Alert	443
■ Event Configuration	0
■ Metrics	450
■ Composite Metrics	472
■ Anomaly Detection	479
■ Real-time logs	483
■ Composite log	496
■ Event Configuration	510
■ Event Notification Settings	514
■ Event Notification Format	536
■ Event History and Monitoring	544
■ Real-time Notifications	552
■ System Events	555
◦ Management	
■ Correlated project management	558
◦ Laboratory	566

PostgreSQL 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](#).

PostgreSQL Monitoring features

- **Realtime DB Monitoring**

It collects real-time DB performance metrics and the data of running clients (active sessions). You can check the operation status at a glance to immediately recognize the load occurrence status, and you can immediately recognize any problems through alarm settings.

- **Specific point analysis**

By using the metrics' short-term trends, long-term trends, and metric comparison functions, you can analyze problems at the past. The database parameter change history is traced.

- **Efficient structure even on a large scale**

Even when managing multiple DB servers, it provides an optimal analysis environment through unified performance data collection and efficient data storage structure.

- **Cloud DB monitoring available**

Structure for query-based monitoring data collection does not depend on the DB server type. DBs in AWS, GCP, Azure, and OCI environments can also be monitored, and monitoring for cloud resource metrics and integrated items is also possible.

- **Integrated Monitoring**

If you have installed all applications, servers, and database monitoring products, you can quickly analyze the causes of failures through linked views when detecting service problems.

- **WhaTap SaaS**

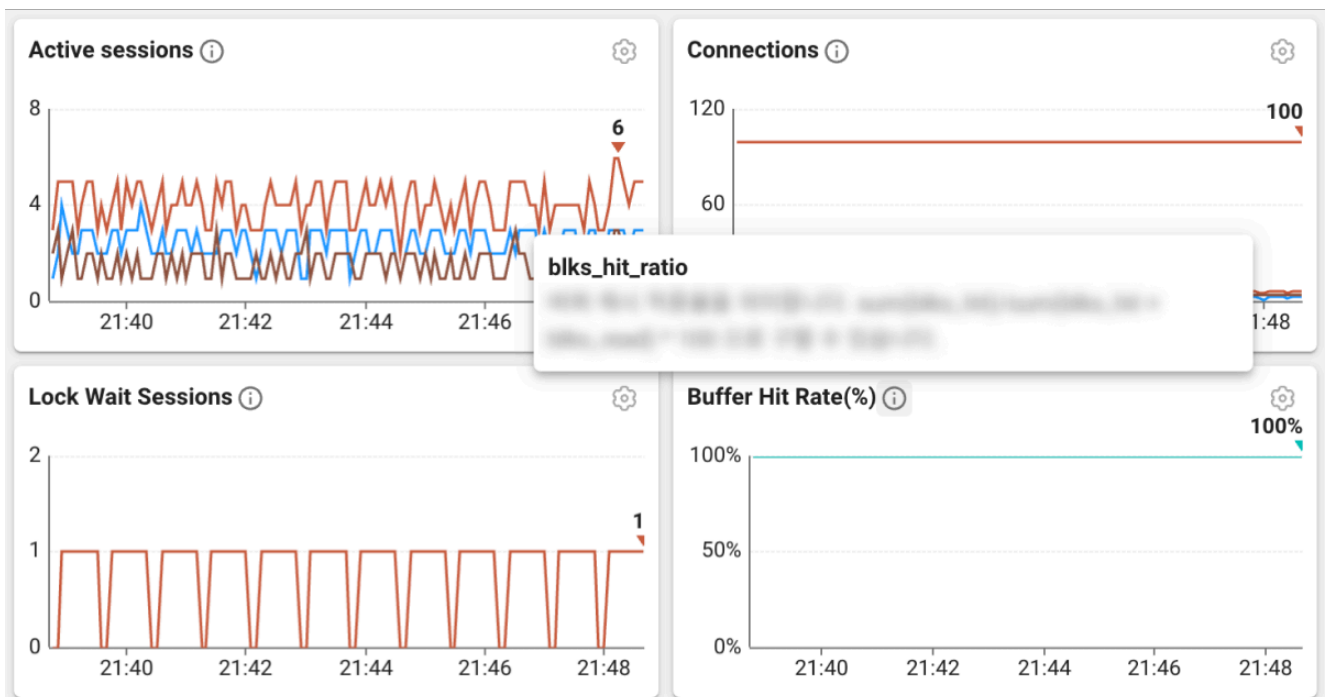
You can select from various database products, and freely set the period of use by monthly subscription. (available also with an installation type)

New features in Database Monitoring

The PostgreSQL Monitoring with enhanced charting and collecting features has been updated. Update your agent to get the latest features.

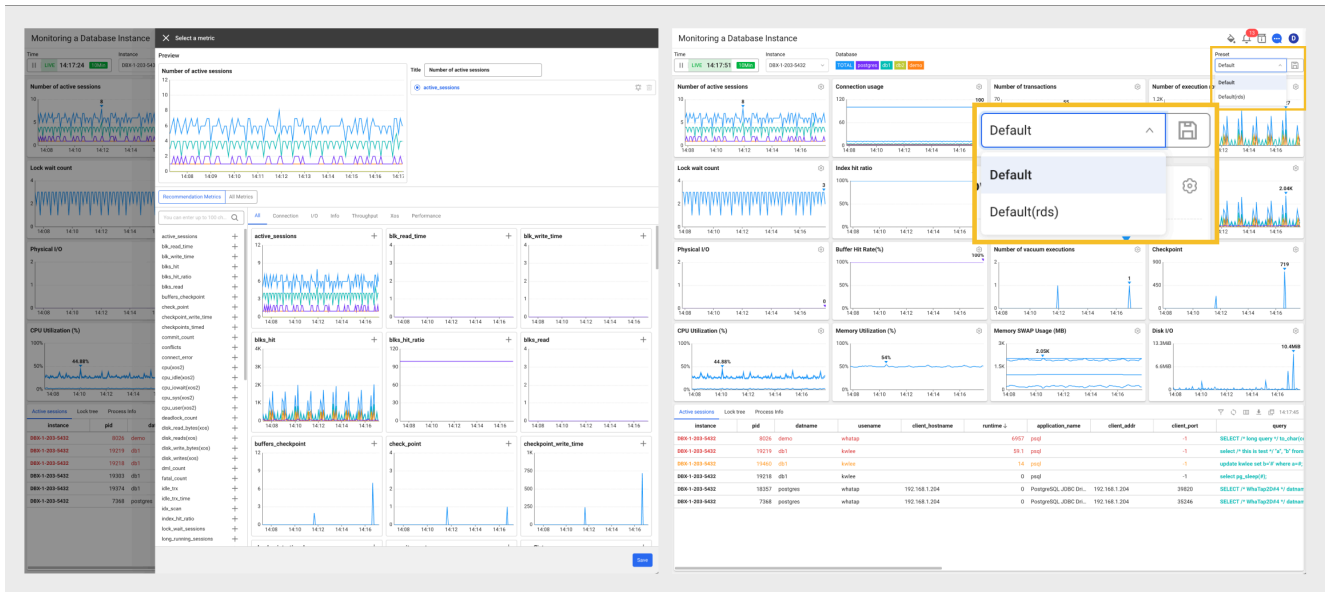
! The minimum version of the agent to use the PostgreSQL monitoring is 1.6.7. For more information about the agent update, see [the following](#).

- **Provision of DB metrics data**



Helps and tuning tips for metrics are added, and convenience is provided by categorizing and recommending the metrics so that users can easily understand numerous metrics. (multilingual support planned)

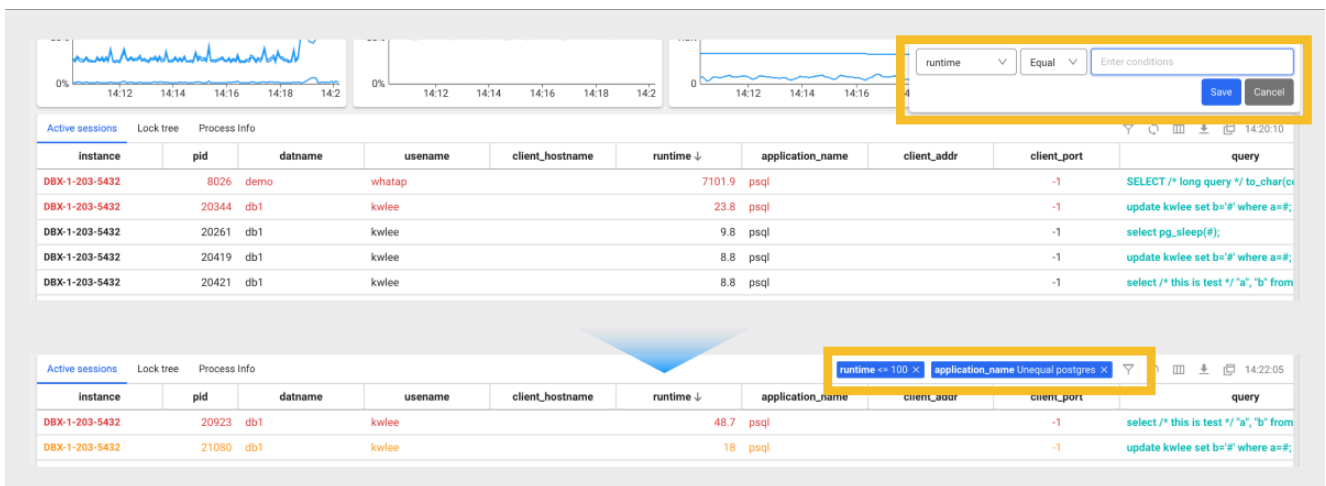
- Dashboard improvements



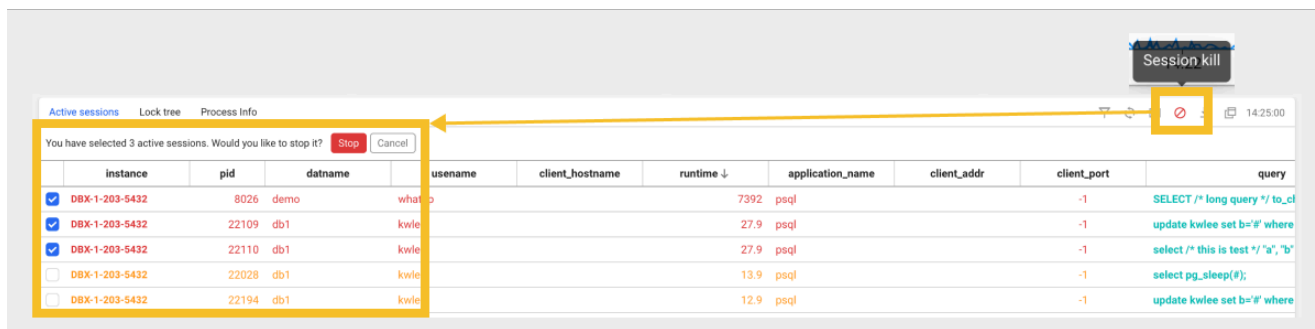
For dashboards, the function to add multiple metrics is provided to one chart. In addition, by adding presets, users can quickly check the desired metrics.

- UI improvements

Improvements in the instance monitoring filter

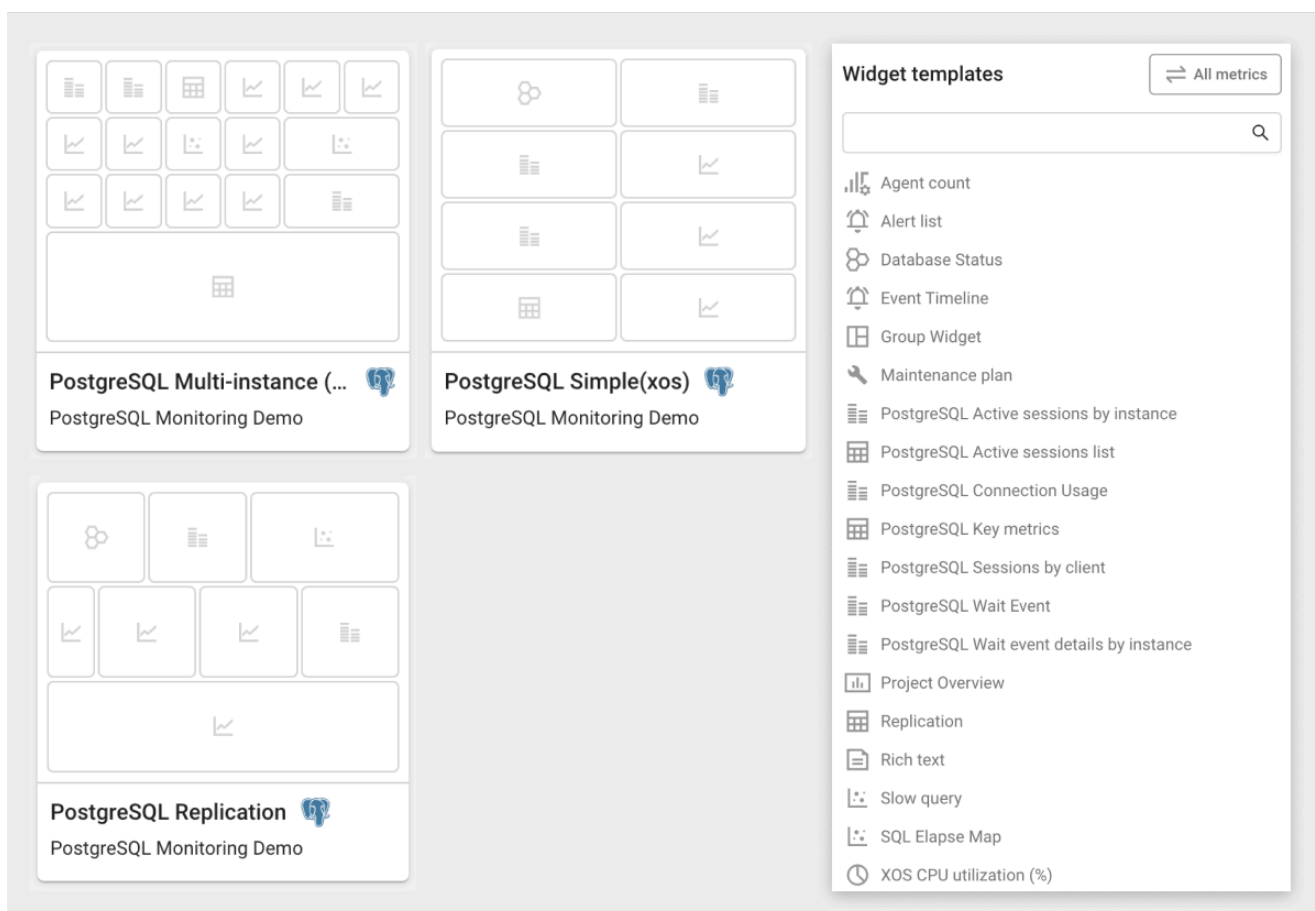


Session Kill UI improvements



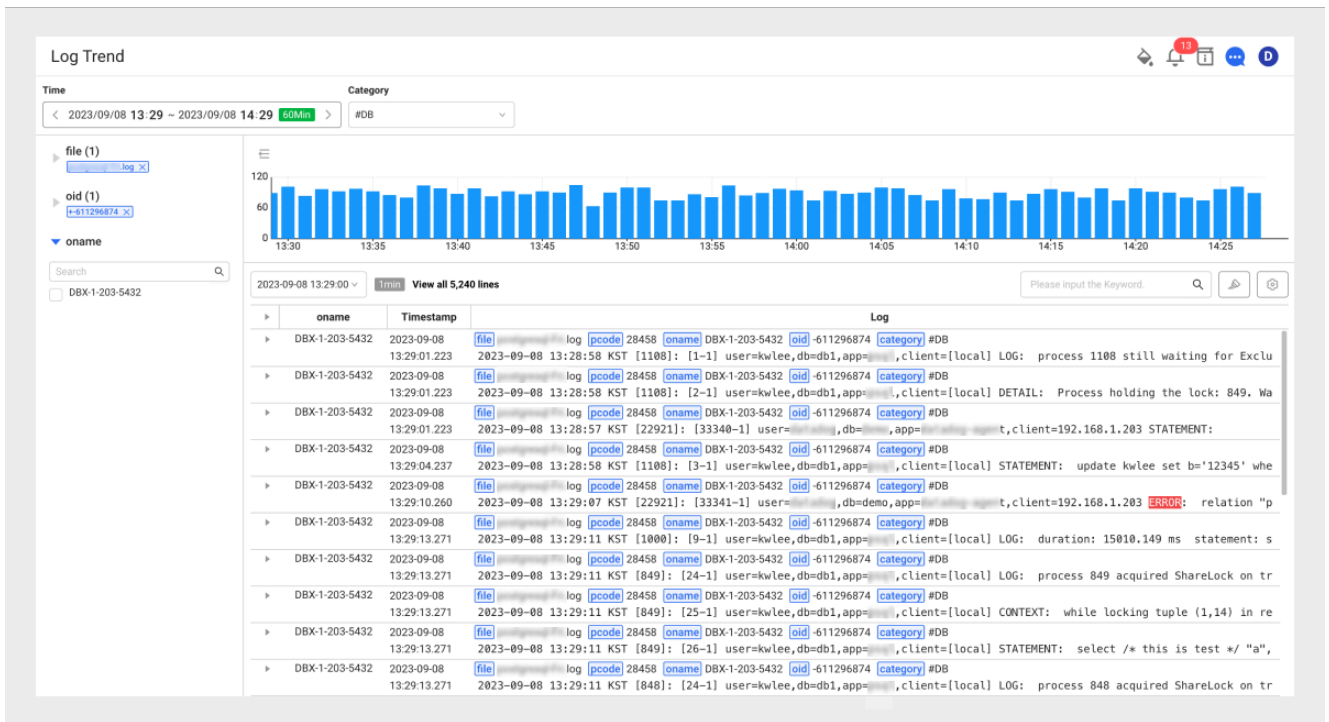
The instance monitoring filter has been improved, and the user interfaces for Session Kill have been improved.

- **Addition of Flex Board templates and widgets**



By converting many DB indicators into metrics, various templates and widgets have been provided.

- **Log**



Through **Live Tail**, **Log Trend**, **Log Search**, and **Log Alert**, you can search **DB logs** and can see their trend as well as the log-related alerts.

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.

Supported Environment

Before starting the PostgreSQL monitoring, check the following support environment.

Supported versions for the database

PostgreSQL 9.2 or later (e.g. EDB, Tmax OpenSQL) is supported.

- ! • The server to install the agent on must have Java 8 or later installed.
- The database server must allow inbound connections so that the DBX agent can access the DB port.
- It is required to open the outbound on the port 6600 so that the DBX agent can send data to the WhaTap server.

Agent minimum installation specifications

- **CPU:** Approximately 20 agents can be installed per core (may differ depending on the DB server load).
- **Memory:** 200 MB
- **Disk:** 100 MB (Agent space of 50 MB + Log file generation space of 50 MB)

! Minimum installation requirements based on the number of agents

Number of installed agents	CPU	Memory	Disk
10	1-core	2 GB	1 GB
20	1-core	4 GB	2 GB
40	2-core	8 GB	4 GB

Agent operation method

The DB agent basically operates in the following way:

- **Execution cycle:** 5 secs (default)
- **Number of queries performed:** 5 to 6
- **Speed:** It terminates within 1 second when the server load is normal.
- Up to 2 sessions (5-second collection session, other information collection session)

When a timeout occurs during collection query, the DB agent operates in the following way to minimize the load.

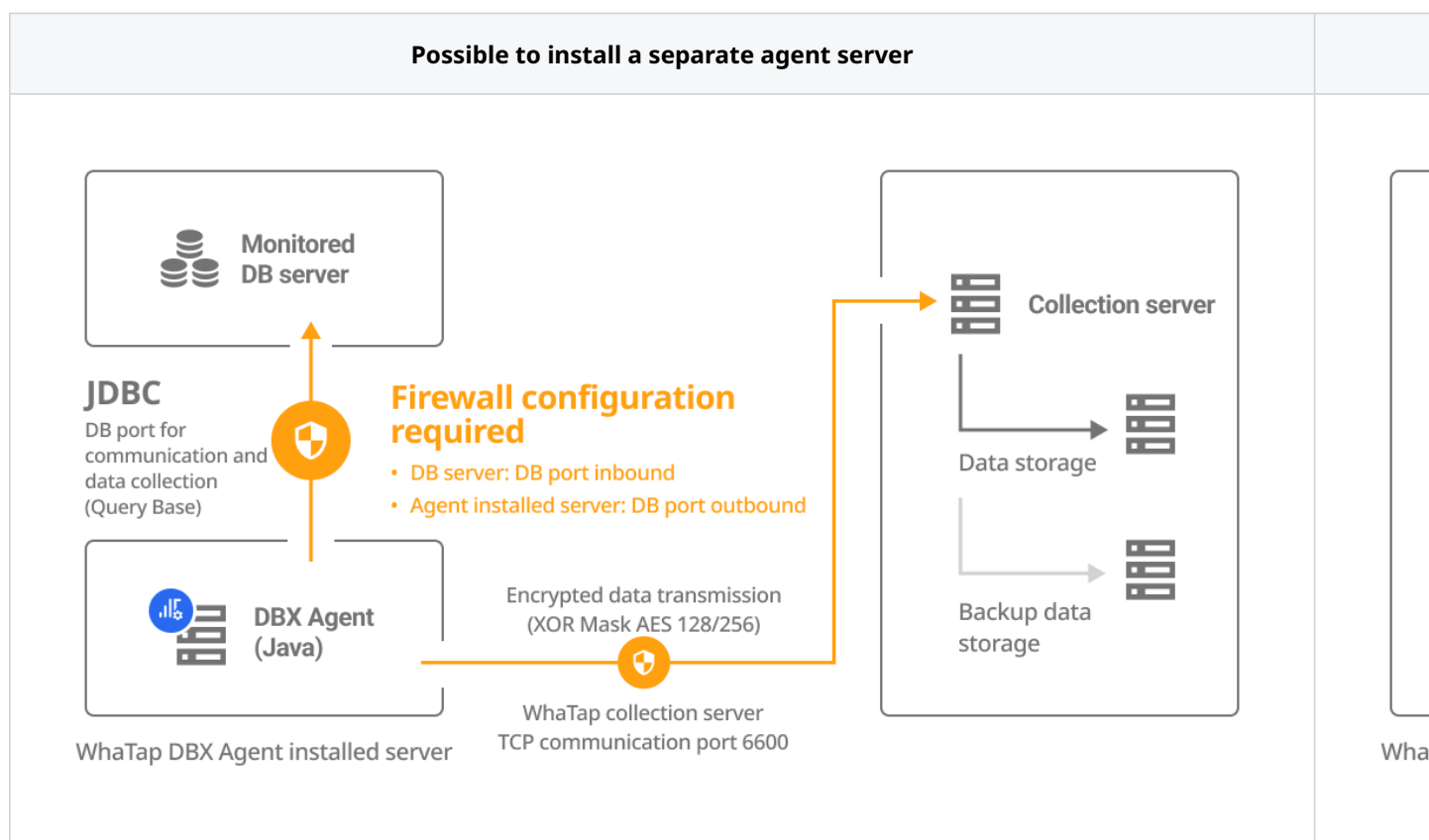
- Do not run a new collection query until the previous collection query has finished.
- If the timeout exceeds 12 seconds (exceeds 2 refresh cycles), the session is reconnected after termination.
- If timeout occurs 3 times or more, wait 10 seconds and try again.

ⓘ Depending on the load status of the DB server, it may take a long time or a timeout may occur until the result of a collection query appears on the WhaTap monitoring service screen.

Agent structure

The WhaTap database monitoring service consists of an agent and a collection server, and the data is encrypted for security between the agent and the collection server. It is made without an external library so there is no need to restart the system for installation and update.

- WhaTap database monitoring service can perform monitoring on a separate agent server without installing an agent on the database server. Monitoring can be applied flexibly according to the user environment.



- The XOS agent is an optional add-on agent that can monitor the process usage of the database server. To monitor the process usage of the database server, run a separate agent on the database server to collect data.
- The agent and server use the binary communication protocol. In addition, data is selectively encrypted to ensure data security while reducing the burden of encryption.

- ⓘ • The WhaTap agent must be installed on a server that can communicate over the network with the database server to be monitored.
- **JDK of Java 8** or later must be installed on the WhaTap DBX agent server.

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.

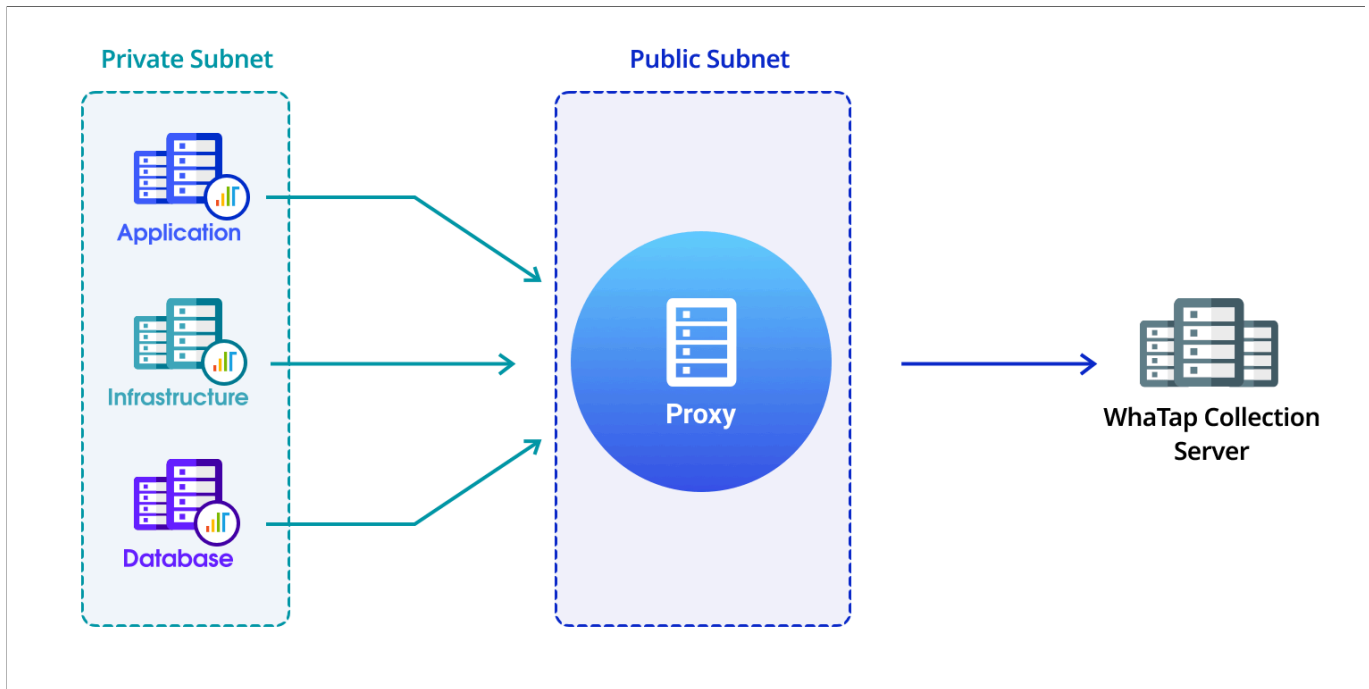
Firewall

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
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
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](#).

Mobile app

The WhaTap mobile app supports both Android and iOS environments. You can install the WhaTap app by following the link or scanning the QR code. For detailed information about the mobile app, see [the following](#).

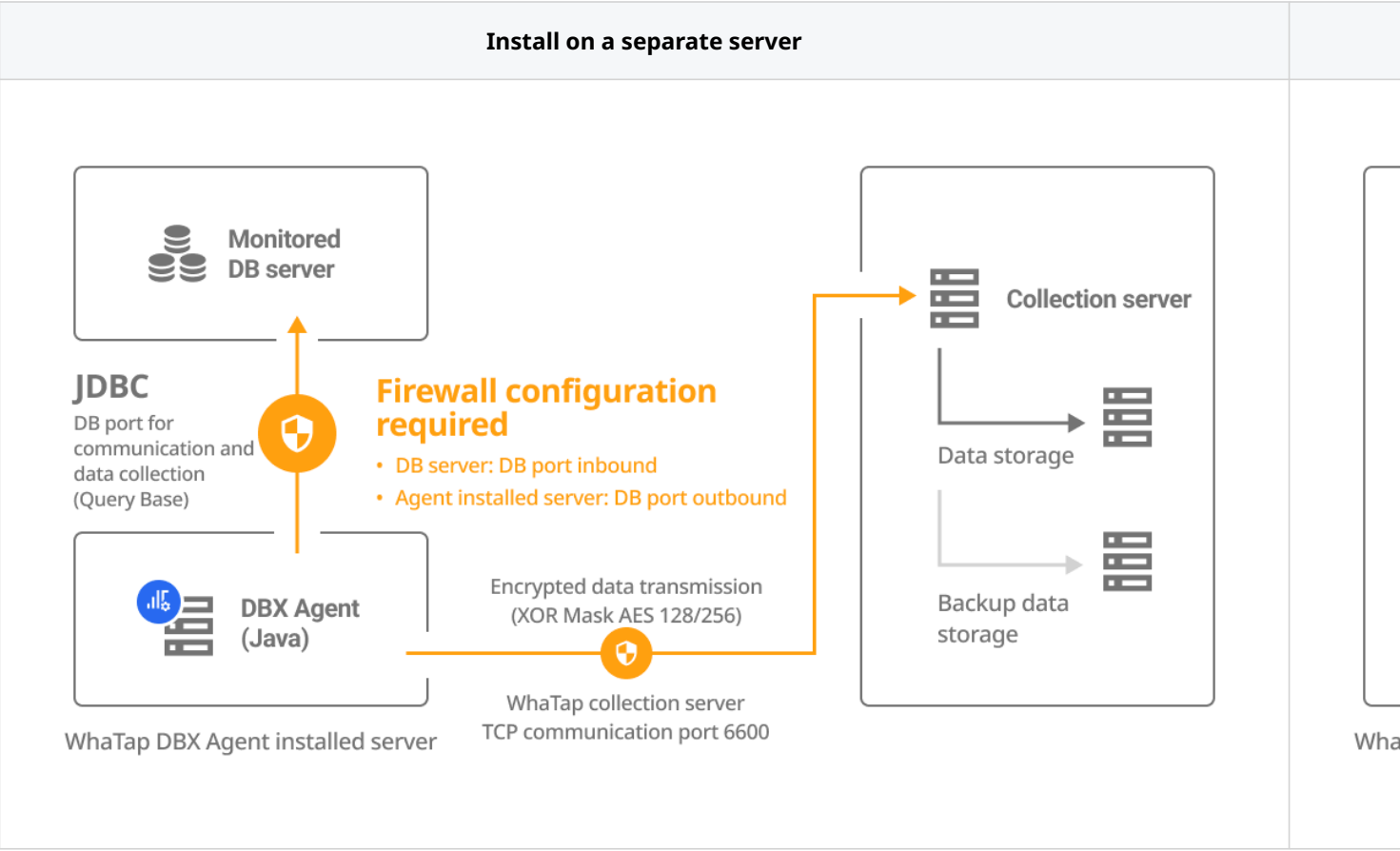
iOS	Android
 iOS 12 or later	 Android 5.0 or later

Manual Installation

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

Checking the configuration diagram

The DBX agent can be installed on a separate server or on the DB server. Select a method how to install the agent and check its configuration and firewall.



Account creation

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

- ❗ • If you already have an account with the required permissions, skip this step and proceed to the [next step](#).
- In the example code, `DB_User` is the database user account name. Change it to the user account name you are using.
- Enter your user password in `DB_Password` in the example code.

Assign additional roles depending on the features and operation environment.

- **PostgreSQL 10 or later**

SQL

```
create user DB_User with password 'DB Password';
grant pg_monitor to DB_User;
```

- **Below PostgreSQL 10**

SQL

```
create user DB_User with password 'DB Password';
alter user DB_User with superuser;
```

- **In case of using the kill session (version 10 or later)**

Database monitoring provides the **Kill Session** function that allows you to kill an active session. To use this function, set the role as follows:

SQL

```
grant pg_signal_backend to DB_User;
```


- In case of collecting the table size (Stat/Report > DB size > Table size (TOP 50))

SQL

```
grant usage on schema {schema_name} to DB_User;
```

- ① Execute the command for each individual DB.

- In case of collecting the object information (Analysis > Top Object)

SQL

```
grant select on all tables in schema {schema_name} to DB_User;
```

Support features for each role

- ① 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	Menu and feature	Details
Default role	grant pg_monitor to whatap;	Monitoring a Database Instance / Monitoring Multiple Instances / SQL Statistics	It is the same as grant pg_read_all_stats,pg_read_all_settings to DB_User; .
Role details	grant usage on schema {schema_name} to whatap;	DB size / Table size (TOP 50)	The roles for all schemas in the DB are required.
	grant select on all tables in schema {schema_name} to whatap;	Table layout query	-

Role	command	Menu and feature	Details
	grant pg_read_all_settings to whatap;	Parameter query	Specific parameter values may be invisible.
Session kill	grant pg_signal_backend to whatap;	Active sessions > Session kill	PostgreSQL 9.6 or earlier requires the superuser privilege.

Downloading the DBX agent

Download the DBX agent file. Use the following two methods.

- You can download it by using the 'wget' command.

BASH

```
wget -O whatap.agent.database.tar.gz "https://service.whatap.io/download/dbx_agent?type=postgresql&format=tar.gz"
```

- If you cannot download it with the command, select **Download** on the WhaTap monitoring service screen.

Download the DBX agent file and then unzip it.

BASH

```
tar -zxvf whatap.agent.database.tar.gz
```

ⓘ 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.

DBX Agent Setting

Follow the instructions on the agent installation screen to configure the agent. By entering the required items on the screen, the configuration and commands can be auto-completed.

✓ You can quickly and easily configure the agent by running the [setup.sh](#) file among the DBX agent configuration files. For more information, see [the following](#).

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 DB connection data.

whatap.conf

```
license={AccessKey}
whatap.server.host=13.124.11.223/13.209.172.35 # WhaTap server information
dbms=postgresql
db=postgres
db_ip={DB_Server_IP}
db_port={DB_Server_Port}
```

Downloading JDBC

Download the JDBC driver that matches the operating system and version of the database server to the DBX agent path where you unzipped the file. `/unzipped folder/jdbc`

ⓘ PostgreSQL 9.2 or later: <https://jdbc.postgresql.org/download>

ⓘ In `/unzipped folder/jdbc/README.md`, you can also see the JDBC driver installation paths for each database.

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.

• Linux Windows

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
```

CMD

```
uid.bat "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`.
- On Windows, the escape character (`\`) is not required for special characters in a password. However, it is required when using double quotes (`"`) within the password.

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.

• [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
```

Starting the monitoring

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

- [Linux](#) [Windows](#)

```
./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
```

```
start.bat
```

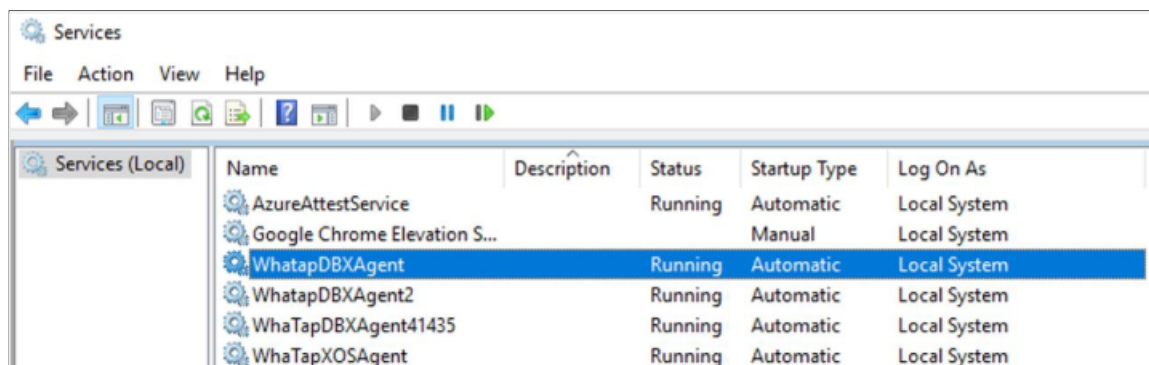
Check whether there are any errors in the execution log displayed on the screen, and check whether a chart appears normally on the dashboard screen (**Instance List**). If there is no problem, press Ctrl+C to terminate the execution, and then register it as a Windows service in the following order.

Registering a service

Execute the following command to register a service.

```
install_WindowsService.bat create WhatapDBXAgent
```

When a service is registered, you can check it in the service management tools as shown below. Find the service, right-click it, and then select the Start option to start the service.



❗ In **Control Panel > Windows Tools > Services** (services.msc), you can start or stop the **WhatapDBXAgent** service. Depending on the Windows version, the **service** path may differ.

Deleting a service

Execute the following command to delete a service.

```
install_WindowsService.bat delete WhatapDBXAgent
```

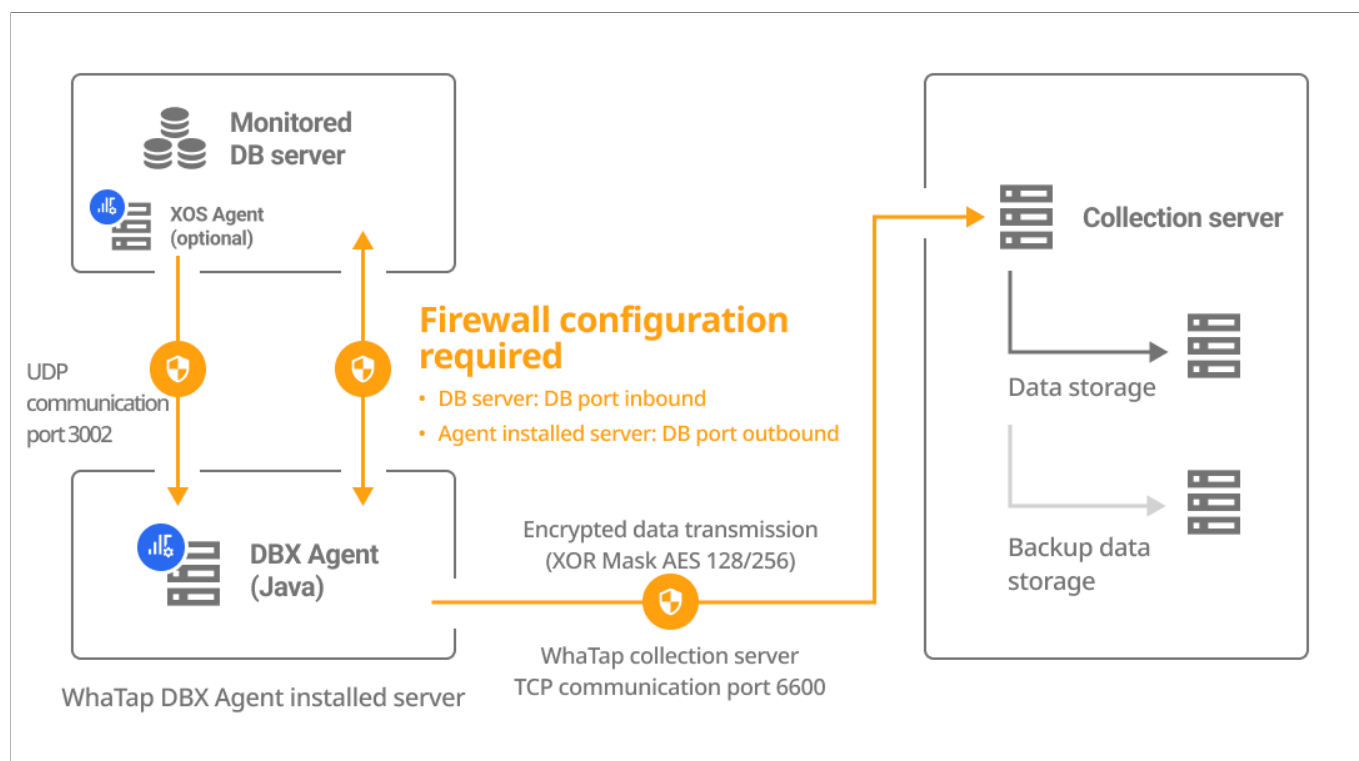
You completed installing the agent for database monitoring. In [the following](#), check the post-installation checklist.

Installing the additional agent (XOS) and applying other options

To additionally monitor the resources of the database server, run a separate XOS agent process on the database server to collect data.

- ❗ • It can be applied to only the OS environment running on the x86 architecture.
- The additional agent installation process is **optional**.
- For more information about the XOS agent configuration options, see [the following](#).

- To additionally collect and monitor monitoring metrics provided by cloud services on the dashboard of the database project, see [the following](#).



Configuring the whatap.conf file

Set the following options in the `whatap.conf` file in the path where the DBX agent has been installed.

`whatap.conf`

```
xos=1
xos_port=3002
```

Move the `xos` folder (`/unzip folder/xos/`) to the database server.

Configuring the xos.conf file

Set the following options in the `xos.conf` file in the `xos` path moved to the database server.

xos.conf

```
dbx_ip={DB_Agent_IP}
dbx_port=3002 # default 3002
cpu_limit=0
mem_limit=10240
```

- ✓ In **Agent Installation**, when you enter the DB data to **DB Agent IP** and **DB Agent Port**, the agent options are automatically generated.

Running the XOS agent

• [Linux](#) [Windows](#)

Run the XOS agent.

```
./start.sh
```

- ! • To transmit monitored data to the DBX agent, the port set to `dbx_port` (default 3002) must have been open. (**UDP Outbound**)
- To run the XOS agent in the background, run the `./startd.sh` file.

Run the agent with a batch file in the `xos` folder.

```
start.bat
```

After confirming that there are no errors in the execution log displayed on the screen and terminating the execution, register for the service in the following order.

! How to fix missing of `msvc120.dll` and `msvcr120.dll`

If you cannot run the xos agent because the `msvc120.dll` and `msvcr120.dll` files are missing, it is because Microsoft Visual C++ 2013 Redistributable 2013 is not installed or is corrupted. You can resolve this issue by installing the **Microsoft Visual C++ 2013 Redistributable Package**.

- ! Download and install the package that matches the system environment from the following link.
[Microsoft Visual C++ 2013 Redistributable \(x64\)](#)
- **vcredist_x64.exe:** For running 64-bit applications

• Service registration

Register in the service by running the [install_WindowsService.bat](#) file in the xos folder as follows. If you do not enter a name after the `create` option, the service is registered by default with the name of **WhatapXOSAgent**.

```
install_WindowsService.bat create WhatapXOSAgent
```

• Deletion of a service

```
install_WindowsService.bat delete WhatapXOSAgent
```

- ! To transmit monitored data to the DBX agent, the port set to `dbx_port` (default 3002) must have been open. (UDP Outbound)

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](#).

To additionally monitor the database server resources, set more options in the additional agent (XOS). For more information, see [the following](#).

Cloud Settings

To additionally collect and monitor monitoring metrics provided by cloud services on the dashboard of the database project, 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](#).

Package Manager Installation

This guide provides instructions for installing the WhaTap database monitoring agent using a package manager.

✓ Choosing an Installation Method

- Refer to this document for Debian/Ubuntu-based systems that use the `apt` package manager.
- For Docker environments, refer to [Docker Installation](#).
- For other environments, refer to [Manual Installation](#).

ⓘ Supported Databases

MySQL, PostgreSQL, SQL Server

To use the WhaTap monitoring service, after [Sign in](#), create a project and install the agent to the target server. For more information about registration as a member, see [the following](#).

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)

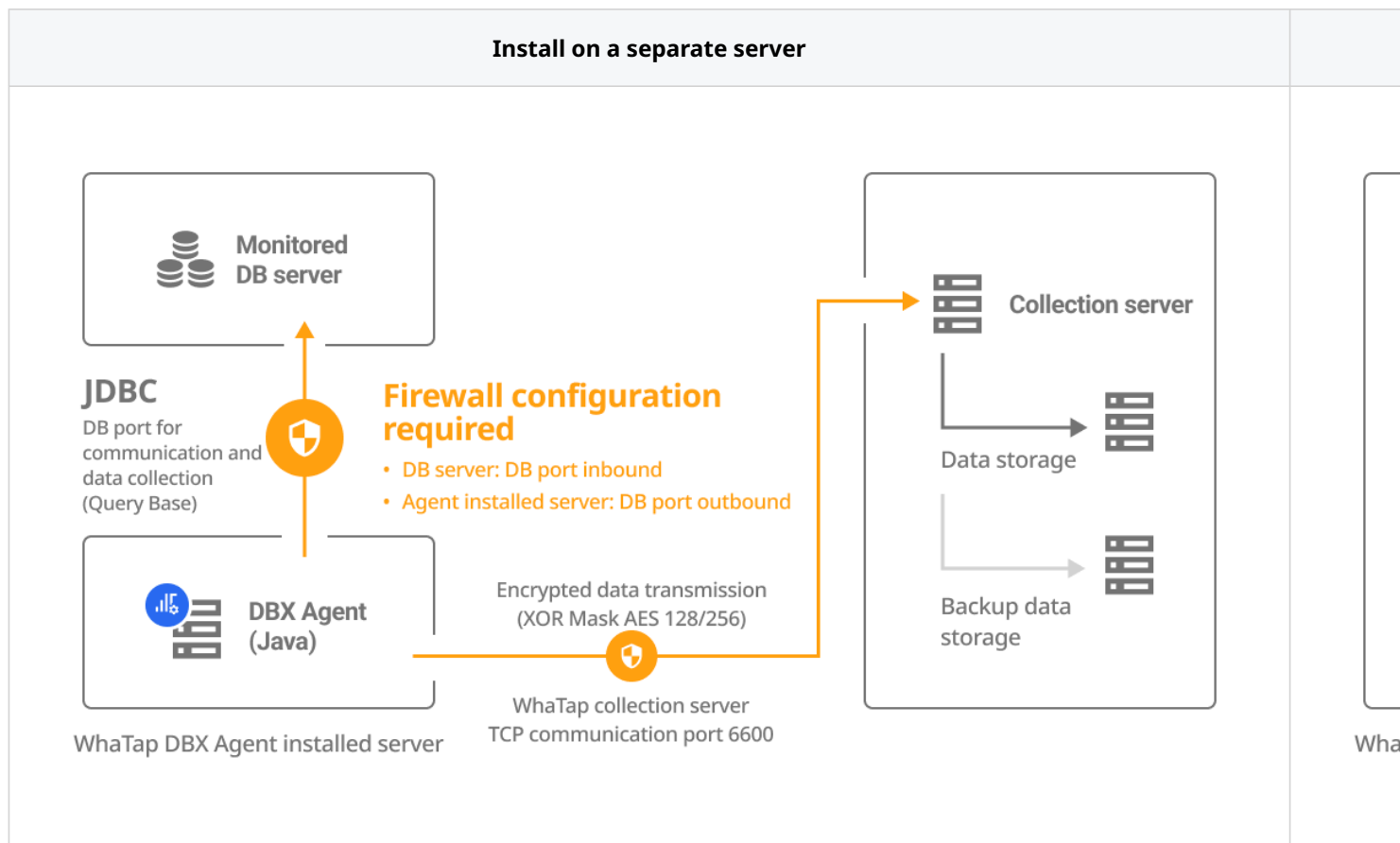
- **Project groups:** Group multiple projects for easier management. Select a group if applicable.
- **Project description:** Add additional details or a description for your project.

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.

Checking the configuration diagram

The DBX agent can be installed on a separate server or on the DB server. Select a method how to install the agent and check its configuration and firewall.



Account creation

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

- ❗ • If you already have an account with the required permissions, skip this step and proceed to the [next step](#).
- In the example code, `DB_User` is the database user account name. Change it to the user account name you are using.
- Enter your user password in `DB_Password` in the example code.

Assign additional roles depending on the features and operation environment.

- **PostgreSQL 10 or later**

SQL

```
create user DB_User with password 'DB Password';
grant pg_monitor to DB_User;
```

- **Below PostgreSQL 10**

SQL

```
create user DB_User with password 'DB Password';
alter user DB_User with superuser;
```

- **In case of using the kill session** (version 10 or later)

Database monitoring provides the **Kill Session** function that allows you to kill an active session. To use this function, set the role as follows:

SQL

```
grant pg_signal_backend to DB_User;
```

- In case of collecting the table size (Stat/Report > DB size > Table size (TOP 50))

SQL

```
grant usage on schema {schema_name} to DB_User;
```

- ① Execute the command for each individual DB.

- In case of collecting the object information (Analysis > Top Object)

SQL

```
grant select on all tables in schema {schema_name} to DB_User;
```

Support features for each role



- ① 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	Menu and feature	Details
Default role	grant pg_monitor to whatap;	Monitoring a Database Instance / Monitoring Multiple Instances / SQL Statistics	It is the same as grant pg_read_all_stats,pg_read_all_settings to DB_User; .
Role details	grant usage on schema {schema_name} to whatap;	DB size / Table size (TOP 50)	The roles for all schemas in the DB are required.
	grant select on all tables in schema {schema_name} to whatap;	Table layout query	-

Role	command	Menu and feature	Details
	grant pg_read_all_settings to whatap;	Parameter query	Specific parameter values may be invisible.
Session kill	grant pg_signal_backend to whatap;	Active sessions > Session kill	PostgreSQL 9.6 or earlier requires the superuser privilege.

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.

Installing the Agent

Adding the WhaTap Repository

Run the following commands to add the official WhaTap repository to your system.

```
curl -s https://repo.whatap.io/debian/release.gpg | gpg --dearmor | sudo tee /etc/apt/trusted.gpg.d/whatap-release.gpg > /dev/null
echo "deb [signed-by=/etc/apt/trusted.gpg.d/whatap-release.gpg] https://repo.whatap.io/debian unstable/" | sudo tee /etc/apt/sources.list.d/whatap.list
sudo apt-get update
```


Installing the Package

Install the WhaTap database monitoring agent with the following command.

```
sudo apt-get install whatap-db
```

Configuring and Starting the Agent

Create the configuration file and start the agent.

```
sudo bash -c 'cat > /usr/whatap/db/whatap.conf <<EOF
license=license_key
whatap.server.host=13.124.11.223/13.209.172.35
dbms=mysql
db_ip=127.0.0.1
db_port=3306
EOF'
cd /usr/whatap/db && sudo /usr/whatap/db/uid.sh whatap whatap && cd -
sudo service whatap-db restart
```

- ❗ You can add additional database options to the `whatap.conf` file.
- Modify the parameter values (database account ID and password) of the `uid.sh` script to match your actual environment.

Managing the Agent

Updating the Agent

To update the agent to the latest version, run the following commands.

```
sudo apt-get update
sudo apt-get install whatap-db
```

```
sudo service whatap-db restart
```

Stopping the Agent and Removing the Package

To remove the WhaTap database monitoring agent, run the following commands.

```
sudo service whatap-db stop  
sudo apt-get remove whatap-db
```

Checking Agent Logs

To check the agent logs, use the following command.

```
sudo service whatap-db log
```

Docker Installation

This guide provides instructions for installing the WhaTap database monitoring agent using Docker containers.

✓ Choosing an Installation Method

- Refer to this document if you are using a Docker environment.
- For Debian/Ubuntu systems, we recommend [Package Manager Installation](#).
- For other environments, refer to [Manual Installation](#).

ⓘ Supported Databases

MySQL, PostgreSQL, SQL Server

To use the WhaTap monitoring service, after [Sign in](#), create a project and install the agent to the target server. For more information about registration as a member, see [the following](#).

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)

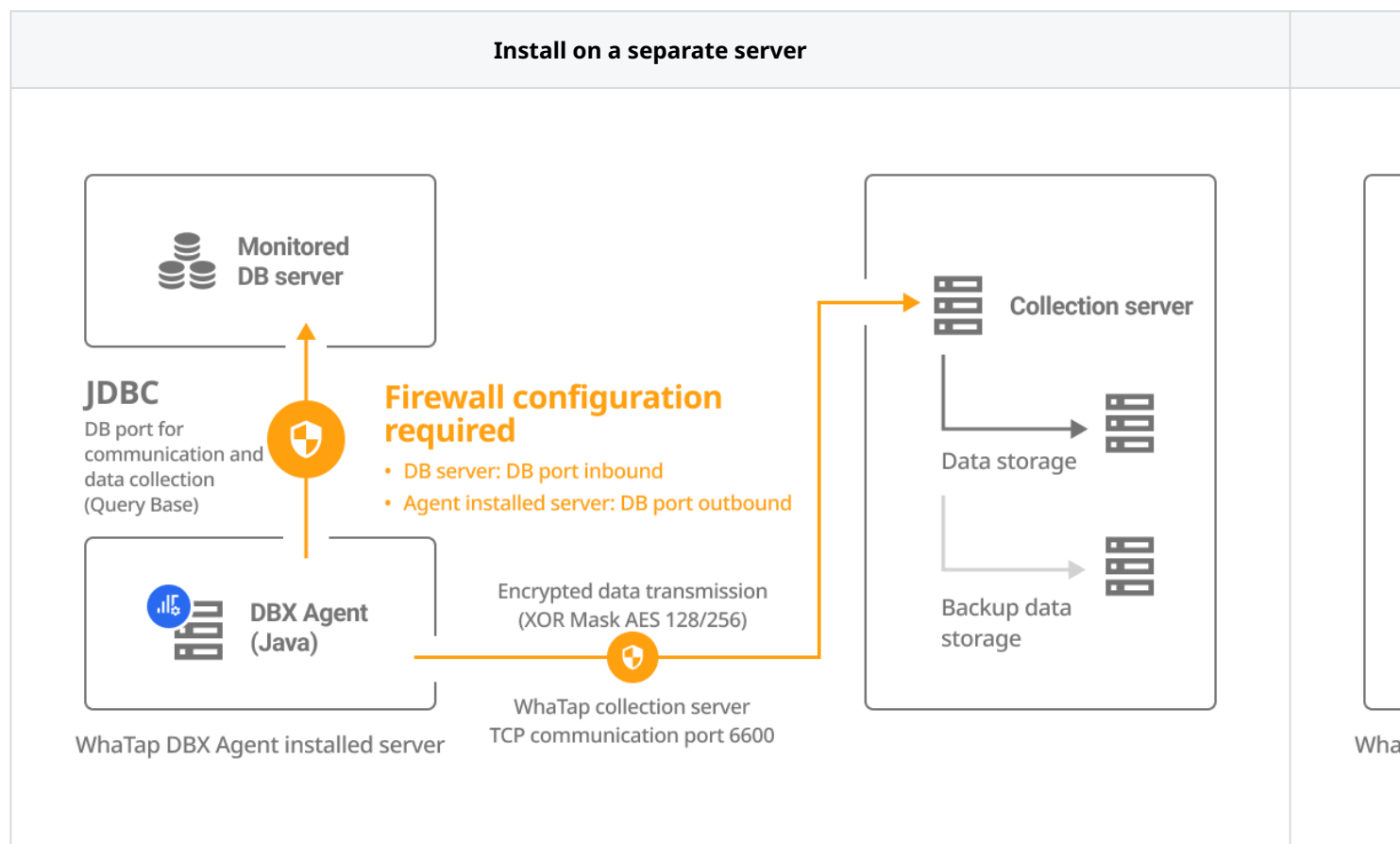
- **Project groups:** Group multiple projects for easier management. Select a group if applicable.
- **Project description:** Add additional details or a description for your project.

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.

Checking the configuration diagram

The DBX agent can be installed on a separate server or on the DB server. Select a method how to install the agent and check its configuration and firewall.



Account creation

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

- ❗ • If you already have an account with the required permissions, skip this step and proceed to the [next step](#).
- In the example code, `DB_User` is the database user account name. Change it to the user account name you are using.
- Enter your user password in `DB_Password` in the example code.

Assign additional roles depending on the features and operation environment.

- **PostgreSQL 10 or later**

SQL

```
create user DB_User with password 'DB Password';
grant pg_monitor to DB_User;
```

- **Below PostgreSQL 10**

SQL

```
create user DB_User with password 'DB Password';
alter user DB_User with superuser;
```

- **In case of using the kill session (version 10 or later)**

Database monitoring provides the **Kill Session** function that allows you to kill an active session. To use this function, set the role as follows:

SQL

```
grant pg_signal_backend to DB_User;
```

- In case of collecting the table size (Stat/Report > DB size > Table size (TOP 50))

SQL

```
grant usage on schema {schema_name} to DB_User;
```

- ① Execute the command for each individual DB.

- In case of collecting the object information (Analysis > Top Object)

SQL

```
grant select on all tables in schema {schema_name} to DB_User;
```

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	Menu and feature	Details
Default role	grant pg_monitor to whatap;	Monitoring a Database Instance / Monitoring Multiple Instances / SQL Statistics	It is the same as grant pg_read_all_stats,pg_read_all_settings to DB_User; .
Role details	grant usage on schema {schema_name} to whatap;	DB size / Table size (TOP 50)	The roles for all schemas in the DB are required.
	grant select on all tables in schema {schema_name} to whatap;	Table layout query	-

Role	command	Menu and feature	Details
	grant pg_read_all_settings to whatap;	Parameter query	Specific parameter values may be invisible.
Session kill	grant pg_signal_backend to whatap;	Active sessions > Session kill	PostgreSQL 9.6 or earlier requires the superuser privilege.

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.

Running the Agent

Run the following command to start the WhaTap database monitoring agent as a Docker container.

```
docker run -d --name whatap-db-agent \
-e WHATAP_CONF="
license=x6051ro88nrc9-x4hmr76jvlfng-z52cnv1dfa8n5c
whatap.server.host=13.124.11.223/13.209.172.35
dbms=mysql
db_ip=10.0.0.1
db_port=3306
connect_option=?useSSL=true&verifyServerCertificate=false&serverTimezone=SERVER_TIMEZONE" \
```

```
-e DB_USERNAME=mydbuser \  
-e DB_PASSWORD=mypassword \  
whatap/db-agent:latest
```

Configuration Items

Required Settings

- `license` : Project access key
- `whatap.server.host` : WhaTap collection server IP (13.124.11.223/13.209.172.35)
- `dbms` : Database type (mysql, postgresql, mssql)
- `db_ip` : Database server IP address
- `db_port` : Database server port number
- `DB_USERNAME` : Database account ID
- `DB_PASSWORD` : Database account password

Optional Settings

- `connect_option` : Database connection options
 - `useSSL=true&verifyServerCertificate=false` : When connecting to the database using SSL
 - `serverTimezone=SERVER_TIMEZONE` : When server timezone configuration is required for JDBC driver connection

ⓘ Modify the configuration values in the above command to match your actual environment.

Verifying Agent Operation

Check if the agent is running properly with the following command.

```
docker logs -f whatap-db-agent
```


Removing the Agent

Run the following command to stop and remove the WhaTap database monitoring agent container.

```
docker stop whatap-db-agent && docker rm whatap-db-agent
```

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.

Checking the logs

- whatap.log

All logs related to the agent startup and collection can be found in the `$WHATAP_HOME /logs/whatap.log` file.

Every day at 00:00 UTC, the existing `whatap.log` file is automatically renamed to `whatap_yymmdd.log_` and

stored by date, and a new [whatap.log](#) file is created to record subsequent logs.

Log files stored by date are automatically deleted after 7 days.

- [xos.log](#)

The logs related to the startup of XOS are recorded in the `$XOS installation path /xos yymmdd .log` file and are automatically created by date.

Checking the agent processes

You can check the status of the active WhaTap DBX agent process with the following command.

```
$ ps -ef | grep dbx
```

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](#).

DB roles and agent configuration

There are menus where data is not viewed depending on the DB roles and agent configuration. See the following to assign the required roles and set the agent options.

❗ Example shows a monitoring account creation as `whatap` :
`create user whatap with password 'DBPassword';`

Menu	DB options and roles	Agent CONFIG.	Remarks
Basic menu	<code>grant pg_monitor to whatap;</code>	-	DBX agent v1.6.15 or later
Dashboard > Slow query	DB parameter: <code>log_min_duration_statement=500</code> (e.g. SQL records taking 500 ms or more) DB role <code>grant PG_READ_ALL_SETTINGS to whatap;</code>	<code>xos.conf</code> <code>slow_query_interval=30</code> (Default 30 seconds) <code>whatap.conf</code> <code>slow_query_log=ON</code>	-

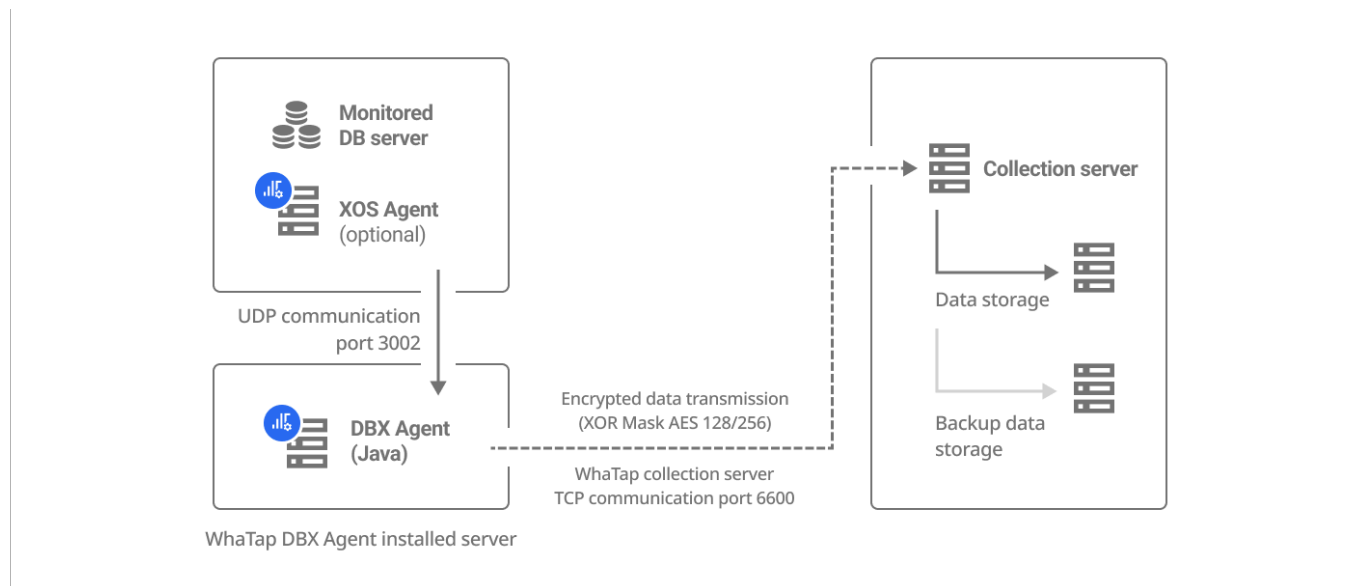
Menu	DB options and roles	Agent CONFIG.	Remarks
Instance monitoring > Session kill	<code>grant pg_signal_backend to whatap;</code>	-	-
Analysis > Top object	<code>grant select on all tables in schema {schema_name} to whatap;</code>	<code>whatap.conf</code> <code>pg_object=true</code>	DBX agent v1.6.15 or later The lists in the Bloating and Age tabs may not be viewed when the thresholds (size, ratio) are not exceeded even if the roles and agent options have been configured.
Stat/Report > Database size	<code>grant usage on schema {schema_name} to whatap;</code>	<code>whatap.conf</code> <code>tables=true</code>	It is the role required to view the table size under the Database size menu.
Object information (Object detail) modal window	<code>grant select on all tables in schema {schema_name} to whatap;</code>	-	Pop-up window that appears when the table is clicked in Top object , Database size , and SQL details (Plan). With no role, the column list does not appear.

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.

- **DBX agent (default agent)**

Query-based performance data is collected and transmitted to the server. Monitoring is possible through a separate agent server without installing agents directly on the database server to be monitored.

- **XOS Agent (additional agent)**

The additional agent is an optional add-on agent that can monitor the process usage of the database server. To monitor the process usage of the database server, run a separate agent on the database server to collect data.

❗ The XOS agent can be applied to only the OS environment running on the x86 architecture.

- **Network**

- Default agent: The WhaTap monitoring agent uses the external communication (TCP) port 6600 to transfer the collected monitoring data to the WhaTap collection server.
- Additional agent: The external communication (UDP) port 3002 is used between the server where the agent has been installed and the database server. If the internal port conflicts, the port can be changed using the `dbx_port` option.

Agent CONFIG.

For more information about agent installation, see the following.

Quick agent configuration

You can easily configure the agent by running the `setup.sh` file among the DBX agent configuration files.

DBX Agent Setting

It guides you to the options you can set in the `whatap.conf` file.

 XOS agent configuration

It provides a configuration method for additional monitoring of the database server resources.

 Cloud Agent Configuration

4 items

Quick Agent Configuration

You can quickly and easily configure the agent by running the [setup.sh](#) file among the DBX agent configuration files.

Unzipping the downloaded file

Unzip the downloaded DBX agent file.

BASH

```
tar -zxvf whatap.agent.database.tar.gz
```

Running setup.sh

Run the [setup.sh](#) file and then follow the on-screen instructions to configure the agent, enter the DB server information, download JDBC, create an account for monitoring, and set the roles. Enter the required data to automatically complete all settings.

BASH

```
./setup.sh
```

```
Hello. This is a monitoring configuration file helper. If an error occurs during the process, please take action according to the instructions.

**Proceed to the next step**
=====
1.Register WhaTap server information
2.Register Whatap server access key
3.Register DB server information (ip/port check)
4.Download related files (id/pwd check)
5.Register a monitoring account
=====

if you want to stop setup prees 'q' or 'quit'

## STEP1 Register WhaTap server information
Whatap > Management > Agent Installation > 1. Check Whatap Server ip and port
WhaTap Server Ip?
```

❗ To cancel the agent configuration process, enter 'q' or 'quit' and then press Enter.

STEP1 Register WhaTap server information

BASH

```
WhaTap Server Ip?  
WhaTap Server Port (default 6600)?
```

Enter the WhaTap's server information. In the **DBX Agent Setting** section of **Management > Agent Installation**, you can see the server IP and port number.

The default value of the WhaTap's server port is 6600. If you press Enter with no entry, the values are set to default.

STEP2 Register WhaTap server access key

BASH

```
WhaTap Project Access Key?
```

Enter the project access key. In the **Check access key** section of **Management > Agent Installation**, select **Copy** and then paste it.

If a correct access key is entered, the project code is output.

STEP3 Register DB server information (ip/port check)

BASH

```
Monitoring DBMS type?  
Monitoring DBMS Server Ip?  
Monitoring DBMS Server Port?
```

Enter the information of the database server for monitoring.

- **Monitoring DBMS type:** Select a type of the database displayed on the screen.

- **Monitoring DBMS Server Ip:** Enter the IP address of the database server.
- **Monitoring DBMS Server Port:** Enter the port number of the database server.

ⓘ After entering the IP address and port number of the database server, verify the connection status. If you enter incorrect values, you must enter the IP address and port number again.

STEP4 Download jdbc driver

The JDBC driver that matches the OS and database server version is automatically downloaded to the DBX agent path.

STEP5 Register a monitoring account

BASH

```
Monitoring Database Name?
Monitoring User ID?
Monitoring User \{user_id\'s Password?
Connecting DBMS Server with SSL (default false)?
```

It is the process to generate an encrypted UID for database connection. Enter the account created in the [Create account](#) section.

- **Monitoring Database Name:** Enter the name of the database to monitor.
- **Monitoring User ID / Monitoring User {user_id}'s Password:** Enter the account ID and password. Passwords are saved encrypted.
- **Connecting DBMS Server with SSL:** Enter `true` when connecting to the database server via the SSL protocol. The default value is `false`.

Running the agent

If you complete the agent installation process by using the `setup.sh` file, database information and settings in the `whatap.conf` file appear.

```

***** DATABASE INFO *****
jdbc:postgresql://192.168.1.203:5432/postgres
DatabaseMajorVersion = 10
DatabaseMinorVersion = 4
DatabaseProductName = PostgreSQL
DatabaseProductVersion = 10.4
DefaultTransactionIsolation = 2
DriverMajorVersion = 42
DriverMinorVersion = 1
*****

/Users/hjwang/Work/dbx_setup/./whatap.conf saved

***** whatap.conf *****
whatap.server.host=15.165.146.117
whatap.server.port=6600
license=x48re21p20h1a-x320c1x093d0a-x3ha4dp1298w23
dbms=postgresql
db_ip=192.168.1.203
db_port=5432
db=postgres
db_user=whatap
db_pwd={{TcKADwSL30C/4w-}}
db_ssl=false
*****

=====The file exists=====

Do you want to start the agent as a daemon? ([y]/n)y

The agent was started as a daemon.
Please check the log [/Users/hjwang/Work/dbx_setup/logs/whatap.log].

```

To use the agent like a daemon, enter 'y' in the last field.

Next steps

- After completing the agent configuration, see [the following](#) and then perform inspection according to the checklist.
- To monitor the resources of the database server by installing an additional agent (XOS), see [the following](#).

DBX Agent Setting

Write the settings for the DBX 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.

- **whatap.server.port** *String*

Default `6600`

Set the port of the collection server. Because only one port can be set, use the same port set in the "whatap_server_host" option.

- **dbms** *String*

`postgresql`

 If other option values are entered, data collection for monitoring may not work.

- **db** *String*

Database name.

- **db_ip** *String*

Set the IP address of the database server to be monitored.

- **db_port** *String*

Set the port used for DB communication.

- **aws_iam_auth** Boolean

Default `false`

This option connects using an AWS IAM authentication token instead of a DB password, for AWS RDS/Aurora (MySQL, MariaDB, PostgreSQL) environments. If set to `true`, the agent connects using an IAM authentication token; if `false` (default), it connects using the existing password-based method.

ⓘ Depending on the `aws_arn` option, you can also enter the AWS access key and secret key directly in `whatap.conf`.

Data collection

- **db_param_enabled** Boolean

Default `true`

It sets whether or not to collect the DB parameters.

- **conn_fail_count** Int

Default `15`

It sends a notification when the connection error occurs consecutively 15 times. You can adjust the notification trigger criteria by entering the desired number.

- **long_run_session_sec** Second

Default `10` (unit: second)

A session that has been running longer than the set time is treated as a long run session.

- **tables** Boolean

Default `true`

It collects the table size information. In **Stat/Report > DB size**, you can see the table size by selecting the database.

- **tables_hour** Int

Default 5

Set the time to collect data for storing the table size information. The default value is 5 and collection starts at 5 am.

- **tables_min_size** Int

Default 10

When collecting the table data, the table data smaller than the set value is not collected. The minimum collection value for the table is 10 MB by default based on `pg_total_relation_size`.

- **replication_name** String

If you set the replication name, it is considered that there is a Master/Slave relationship between DBs with the same `replication_name`.

As an option to define the replication set, if you set the `replication_name` option that has the same value for both master and slave nodes, the **Monitoring a Database Instance** displays a tree structure. It also provides a widget to monitor the replication status in **Flex Board** based on the option value.

- **skip_schema** String

Default: pg_catalog, sys, information_schema, pg_toast

It does not collect tables for the schema set with the option value.

- **skip_db** String

Default: edb, postgres

It does not collect tables for the database set with the option value.

- **slave_ip** string

Check the IP address with `select inet_server_addr();`. It must be one of the IP addresses found when searching `pg_stat_replication.client_addr` on the master node. If the `replication_name` option value is the same and it is a slave DB, specify `slave_ip`.

❗ In case of RDS, because the master/slave relationship is set with `pg_replication_slots.slot_name` instead of the IP address, the `slave_ip` option does not need to be set.

- **statements_min_row** Int

Default 10000 (row)

It is the data collection criteria used in the **PG SQL Statistics** menu. It queries `pg_stat_statements` and collects data if the rows value is greater than `statements_min_row`.

! For more information about the **PG SQL Statistics** menu, see [the following](#).

- **skip_whatap_session** Int

Default `0`

If you set `1`, the session in which the query to collect from WhaTap operates, does not appear in the active session list. (Default value: 0)

- **xview_min_sec** Int

Default `0`

When collecting xview, collection is performed only when the query execution time is longer than the set time. The default setting is to collect all queries whose execution time is longer than 0 seconds.

Enabling the top object

The following is the agent setting to view Top Object information. For more information about Top Object, see [the following](#).

- **pg_object** Boolean

Default `true`

Set whether or not to collect data from the Top object. To collect data, change it to `true`.

- **pg_object_hour** Int

Default `5`

Set the time to collect data from the Top object. The default value is `5` and collection starts at 5 AM.

- **table_bloat_ratio** Percentage

Default `50`

Data is collected when the table's bloating ratio is greater than the set value. The default value is 50%.

- **table_bloat_bytes** Byte

Default 10485760

Data is collected when the table's bloating bytes is greater than the set value. The default value is 10 MB.

- **index_bloat_ratio** Percentage

Default 50

Data is collected when the index's bloating ratio is greater than the set value. The default value is 50%.

- **index_bloat_bytes** Byte

Default 10485760

Data is collected when the index's bloating bytes is greater than the set value. The default value is 10 MB.

- **autovacuum_list_limit** Int

Default 50

Table data is collected according to the set value from the oldest age (`relfrozenxid`). The default value is 50.

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.

- **sqlmon** Int

Default value: 1

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

Supported version: 2.3.5 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
```

Slow Query Collection

To collect slow queries, first configure the database (DB). The WhaTap configuration method differs depending on the environment (on-premises or AWS RDS/Aurora).

❗ The slow query feature is supported only for **MySQL** and **PostgreSQL**.

❗ Check the database settings

Before configuring WhaTap, set the slow query related parameters in the PostgreSQL database (DB). Check that the slow query logging feature is enabled and proceed with the WhaTap configuration.

```
log_min_duration_statement=3000 # 3 seconds
```

On-premise environment

When running on your own server, use the WhaTap's DBX/XOS configuration method. **In addition to the DBX agent installation, be sure to install XOS on the DB server.** For the XOS agent configuration method, see [the following](#).

1. Add the `slow_query_log=on` option to the DBX `whatap.conf` file.
 - **slow_query_log** **Boolean**
It enables (`on`) or disables (`off`) the slow query logging feature. The default value in PostgreSQL is `on` .

2. If XOS has been installed, the file name is automatically recognized. If the file name is not automatically recognized, set `slow_query=` in the XOS `whatap.conf` file.

AWS RDS/Aurora Environment

Because in the AWS RDS/Aurora environment, the XOS agent cannot be installed, slow queries can be collected through **Cloud settings**.

Follow the guide provided in the **Management > Cloud Settings** menu to automatically create monitoring resources in the AWS environment and collect slow query data. No additional configuration is required. For more details, see [the following](#).

ⓘ PostgreSQL does not support the DBX agent configuration. The Cloud settings menu allows you to collect slow query data along with CloudWatch metric.

Applying the AES 256 encryption

The WhaTap DBX agent transfers the collected data to the server after encryption. 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 DBX agent has been 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](#)), the agent name is automatically determined by combining the IP address and the port number. For example, if the IP address is **10.11.12.13** and the port number is **3000**, the agent name is set to **DBX-12-13-3000**.

The key variables that determine the agent name are as follows:

Configuration	Description
<code>{type}</code>	The DBX value is used.
<code>{ip0}</code>	The first byte of the IPv4 address is used. (e.g. 10 in 10.11.12.13)
<code>{ip1}</code>	The second byte of the IPv4 address is used. (e.g. 11 in 10.11.12.13)
<code>{ip2}</code>	The third byte of the IPv4 address is used. (e.g. 12 in 10.11.12.13)
<code>{ip3}</code>	The fourth byte of the IPv4 address is used. (e.g. 13 in 10.11.12.13)
<code>{port}</code>	Database server port

ⓘ If the IP address is not numeric like RDS, the name is set in the following format: DBX- `{ip0}` - `{hash}` - `{port}` , where the `{hash}` value is the IP address that has been converted to 4-digit hexadecimal number.

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

To additionally monitor the resources of the database server, run a separate XOS agent process on the database server to collect data.

- ! • It can be applied to only the OS environment running on the x86 architecture.
- For more information about the XOS agent installation, see [the following](#).

Configuring the whatap.conf file

Set the following options in the [whatap.conf](#) file before running the XOS agent.

- **xos** Int

Default

Set whether or not to use the XOS agent. If the value is the XOS agent is not used for collection. Enter the value as or IP address of the XOS agent. If the IP address is specified, data is collected from only the XOS agent.

- **xos_port** String

Default

Set the port to communicate with the XOS agent. XOS agent can be communicated via UDP. If you have set a firewall on that port, disable it for UDP communication.

- **xos_cpu_iowait** Int

Default

Set whether the CPU utilization collected from the XOS agent is to include the iowait value. If the value is set to , the iowait value is included. If set to , the value is not included.

- **xos_mem_buffcache** Int

Default

Set whether the memory utilization collected from the XOS agent is to include the buffcache value. If the

value is set to `1`, the buffcache value is included. If set to `0`, the value is not included.

Configuring the xos.conf file

Copy the `xos` directory to the database server, and then enter the agent server information and set additional options in the `xos.conf` file.

- **dbx_ip** `String`

Enter the IP address of the server in which the DBX agent has been installed.

- **dbx_port** `String`

Default `3002`

Set the port to communicate with the DBX agent. DBX agent can be communicated via UDP. If you have set a firewall on that port, disable it for UDP communication.

- **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`

It displays the disk utilization, free sizes, and total sizes for each instance. If the value is `0`, the disk utilization is not collected.

- **port_check** `String`

The XOS agent attempts to connect directly to the port specified as an option. If it fails, the port list is sent to the `db_port_check` category.

`xos.conf`

```
port_check=192.168.122.113:3333,192.168.122.113:3334
```

If access cannot be made via the specified port, you can check it after selecting the `db_port_check` category in **Analysis > Metrics Search**.

-	Time	Oid	Tags	Fields	
			oname	error_count 	port_list
1	2025-01-17 18:39:41	1064156037	DBX-5-112-1521	2	192.168.122.113:3333,192.168.122.113:3334
2	2025-01-17 18:39:41	1064156037	DBX-5-112-1521	2	192.168.122.113:3333,192.168.122.113:3334
3	2025-01-17 18:39:51	1064156037	DBX-5-112-1521	2	192.168.122.113:3333,192.168.122.113:3334
4	2025-01-17 18:39:51	1064156037	DBX-5-112-1521	2	192.168.122.113:3333,192.168.122.113:3334

- **process_check** String

You can monitor the running status of processes. Enter the process name as an option. You can register multiple items using comma (,) as the delimiter.

```
xos.conf
```

```
process_check=pmon,smon,tnslsnr,melong,haha
```

When the process containing the string is not monitored, the data is sent to the `db_process_check` category. For example, if there have no **melong** and **haha** processes, you can check it after selecting the `db_process_check` category from the **Analysis > Metrics Search** menu.

-	Time	Oid	Tags	Fields	
			oname	error_count 	process_name
1	2025-01-17 18:35:06	1064156037	DBX-5-112-1521	2	melong,haha
2	2025-01-17 18:35:06	1064156037	DBX-5-112-1521	2	melong,haha
3	2025-01-17 18:35:12	1064156037	DBX-5-112-1521	2	melong,haha
4	2025-01-17 18:35:12	1064156037	DBX-5-112-1521	2	melong,haha

① You can use the `process_check` option to send an alert when a process running in the database ends. For more information, see [the following](#).

- **process_check_interval** Second

Default `6`

The `process_check` option sets the time interval for collecting the process execution status.

- **iplist_interval** Second

Default `0` (Off)

Set the time interval for collecting the IP list for the database server. You can set the time interval in seconds (minimum of 3 seconds).

① If the `xos_dbip=true` option is set in the `whatap.conf` file, it is sent to the `db_ip_list` category. You can check it after selecting the `db_ip_list` category from the **Analysis > Metrics Search** menu.

- **slow_query** String

It collects slow queries from the database. If XOS is installed on the monitored DB but does not automatically recognize the file name, set `slow_query=` in the XOS `whatap.conf` file.

- **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/postgresql/logs/postgresql.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
```

- The `file` option is available in DBX Agent 1.6.7 or later.
- For more information about the **Log** menus, see [the following](#).

Running the XOS agent

Run the following batch file on the database server.

```
$ ./start.sh

# or

$ ./startd.sh # When running in the background
```

Cloud Agent Configuration

This document is a guide to configure the WhaTap Cloud Agent in the AWS and NCP environments. It is configured to be immediately applicable to the operating environment, including key settings, field descriptions, roles configuration, and execution commands.

Agent CONFIG.

The following explains how to configure the AWS Aurora and RDS, and NCP Cloud DB for PostgreSQL.

Configure AWS Aurora and RDS

config.yaml

```
input:
  csp: "aws" # Enter the Cloud Service Provider.
  namespace: "rds"
  region: "us-east-1" # Set the region where the instance is located.
  instances: # The specified instances are collected always. Instances listed under this property are collected at all times, regardless
    of the autoscaling setting.
    - name: "mysql-rds"
      slow_query: true # To use the slow query page, set it to true. This feature is available only in MySQL and PostgreSQL instances.
  clusters:
    autoscale:
      enabled: false # If the autoscale is enabled, the autoscaled instances are added or deleted for collection.
      interval: 60 # Autoscale checking cycle for the specified clusters (unit: second)
    names:
      - "database-cluster-name"
  metrics: # Enter the metrics to collect.
    - "CPUUtilization"
    - "FreeStorageSpace"
    - "FreeableMemory"
    - "ReadLatency"
    - "WriteLatency"
    - "ReadIOPS"
    - "WriteIOPS"
```

```

- "NetworkReceiveThroughput"
- "NetworkTransmitThroughput"
- "FreeLocalStorage"
logs:
  enabled: false # Can be enabled/disabled. Select true/false.
  groups: # Add the desired AWS log groups.
    - "/aws/rds/cluster/database-cluster-name/error"
    - "/aws/lambda/MyLambda"

output: # Enter the WhaTap information to receive the collected metrics.
license: "abcdefg-hijgkjgjk-zxcvnbmbmc"
host: "127.0.0.1"

```

Field	Description
<code>csp</code>	Indicates <code>aws</code> (Amazon Web Services) as a cloud service provider.
<code>namespace</code>	Namespace of the service to be collected - Fixed value: Enter <code>rds</code> when configuring Aurora and RDS.
<code>region</code>	AWS Region code - e.g. <code>us-east-1</code> (Virginia Region)
<code>instances</code>	It always collects the specified instances regardless of whether they are auto-scaled, as long as they are in the list of instances to collect.
<code>instances[].name</code>	RDS InstanceName - e.g. <code>mysql-rds</code>
<code>instances[].slow_query</code>	Whether to collect slow queries for the instance. It is ** available only in MySQL and PostgreSQL instances.** - <code>true</code> : To be collected - <code>false</code> : Not to be collected
<code>clusters</code>	RDS Cluster-related settings
<code>clusters.autoscale.enabled</code>	Whether to collect cluster autoscale - <code>true</code> : Automatically add/remove/collect instances

Field	Description
	- <code>false</code> : Only the instances directly entered in <code>instances</code> are included in the collection targets.
<code>clusters.autoscale.interval</code>	Polling interval for detecting autoscaling status (unit: second)
<code>clusters.names</code>	List of cluster names to check for autoscaling status
<code>metrics</code>	List of metrics to collect - e.g. <code>CPUUtilization</code> , <code>FreeStorageSpace</code> , etc - Note. Shortcut to the official Amazon CloudWatch Metrics documentation for Amazon RDS
<code>output.license</code>	WhaTap's license key to transmit the collected data
<code>output.host</code>	WhaTap server host address (IP) to transmit the collected data to

Cross-account monitoring (multi AWS account)

-  A single cloud agent can monitor RDS/DocumentDB across **multiple AWS accounts**. When using multiple accounts (separate by environment, partner accounts, etc.), you do not need to install an agent per account. One agent uses STS AssumeRole to query resources in other accounts.

Item	Description
Supported targets	AWS RDS (Aurora, standalone), DocumentDB
Authentication method	EC2 Instance Profile + STS AssumeRole

Choosing a configuration method

There are two methods for cluster configuration, and they **cannot be used simultaneously**.

- `names` : Single-account only (queries only the account of the EC2 where the agent is installed, using its IAM Role)
- `account_configs` : Cross-account support (queries multiple accounts via STS AssumeRole)



account_configs configuration

config.yaml

```
input:
  csp: "aws"
  namespace: "rds"
  region: "ap-northeast-2"
  clusters:
    autoscale:
      enabled: true
      interval: 300
  account_configs:
    # Local account (omitting assume_role_arn uses the local account)
    - clusters: ["local-aurora-cluster"]
      instances: ["standalone-mysql-1"]

    # Another account A
    - assume_role_arn: "arn:aws:iam::111111111111:role/DBXCRole"
      clusters: ["partner-cluster-1", "partner-cluster-2"]

    # Another account B
    - assume_role_arn: "arn:aws:iam::222222222222:role/QARole"
      clusters: ["qa-cluster"]
      instances: ["qa-standalone-pg"]
  metrics:
    - "CPUUtilization"
    - "FreeStorageSpace"
    - "FreeableMemory"
    - "ReadLatency"
    - "WriteLatency"

output:
  license: "your-license-key"
  host: "127.0.0.1"
```

Table | account_configs fields



Field	Required	Description
<code>assume_role_arn</code>	N	STS AssumeRole ARN. If omitted, the local account is used.
<code>clusters</code>	Y	Aurora/DocumentDB cluster names. Member instances are automatically discovered.
<code>instances</code>	N	Standalone RDS instance names not belonging to a cluster.

! Difference between `clusters` and `instances`

	<code>clusters</code>	<code>instances</code>
Target	Aurora clusters, DocumentDB clusters	Standalone RDS instances without a cluster
Behavior	Auto-discovers member instances by cluster name	Directly specifies instance names
Autoscale	Automatically reflects instance additions/deletions	N/A
Example	Aurora MySQL cluster	Standalone RDS MySQL instance

Configuration example 1: Single account + standalone instance

```
clusters:
  autoscale:
    enabled: true
    interval: 300
  account_configs:
    - clusters: ["prod-aurora"]
      instances: ["legacy-mysql-standalone"]
```



Configuration example 2: 3 accounts (local + partner + QA)

```
clusters:
  autoscale:
    enabled: true
    interval: 300
  account_configs:
    - clusters: ["prod-cluster", "staging-cluster"]

    - assume_role_arn: "arn:aws:iam::111111111111:role/DBXCRole"
      clusters: ["partner-prod", "partner-dev"]

    - assume_role_arn: "arn:aws:iam::222222222222:role/QADBXCRole"
      clusters: ["qa-cluster"]
      instances: ["qa-standalone-pg"]
```



- `names` and `account_configs` **cannot be used simultaneously**. A validation error occurs at agent startup.
- ARN format is strictly validated. `arn:aws:iam::<12-digit number>:role/<name>` — be careful not to omit the double colons (`::`).
- Each `account_config` requires at least 1 cluster.
- `autoscale.enabled: true` is required for cluster discovery to work.

NCP Cloud DB for PostgreSQL Setup

config.yaml

```
input:
  csp: "ncp" # Enter the Cloud Service Provider.
  namespace: "vpc_postgresql"
  region: "KR" # KR | SGN | JPN | USWN | DEN
  accessKey: "YOUR_ACCESS_KEY"
  secretKey: "YOUR_SECRET_KEY"
```

```

instances: # Specify the name of the DB Server to monitor in the Cloud DB for PostgreSQL (VPC) list.
  - name: "instance-name"
    domain: "pg-xxxxx.vpc-pub-cdb-kr.ntruss.com" # Enter the private or public domain that can be accessed.
metrics: # Enter the metrics to collect.
  - "cpu_user"
  - "mem_used"
  - "disk_usage"
  - "nic_total"
  - "cpu_load_1"
  - "tup_inserted"

output: # Enter the WhaTap information to receive the collected metrics.
license: "4544ee0few5-a880410bb205-2987709519af"
host: "127.0.0.1"

```

Field path	Description
<code>csp</code>	Indicates using the Naver Cloud Platform (<code>ncp</code>) as the cloud service provider.
<code>namespace</code>	It is used as the delimiter for monitoring targets or group name. - e.g. <code>vpc_postgresql</code>
<code>region</code>	Cloud region code - Available values: <code>KR</code> , <code>SGN</code> , <code>JPN</code> , <code>USWN</code> , <code>DEN</code>
<code>accessKey</code>	Access key for API calls to Naver Cloud - Note. How to Create an Access Key and Secret Key
<code>secretKey</code>	Secret key for API calls to Naver Cloud
<code>instances</code>	List of PostgreSQL instances to monitor - Each instance contains the <code>name</code> and <code>domain</code> fields.
<code>instances[].name</code>	Cloud DB for PostgreSQL (VPC) instance name - It can be checked in DB Server > DB Server Name of the NAVER CLOUD PLATFORM console. - e.g. <code>whatap-001-70ld</code>

Field path	Description
<code>instances[].domain</code>	Domain address that can be accessed from the instance - It can be checked in DB Server > Private or Public Domain of the NAVER CLOUD PLATFORM console. - Enter the Private or Public domain address that is being accessed by the DBX agent. - e.g. <code>pg-33rr40.vpc-pub-cdb-kr.ntruss.com</code>
<code>metrics</code>	List of metrics to collect - Note. Official Documentation for Cloud DB for PostgreSQL (VPC) metrics
<code>output</code>	WhaTap information to receive the collected metrics
<code>output.license</code>	WhaTap's license key to transmit the collected metrics
<code>output.host</code>	WhaTap server host address (IP) to transmit the collected data to

How to Issue an Access Key / Secret Key

1. Log in to the [NAVER CLOUD PLATFORM Console](#).
2. From the top menu, go to My Page > Account Management.
3. Click Authentication Key Management in the menu.
4. Click the [Create Key] button to generate a new Access Key and Secret Key.

Role setting

For the agent to operate normally, set the required roles for each service.

AWS Aurora and RDS roles

The Cloud Agent calls the AWS API to perform the following features.

- Check the monitoring targets by querying the AWS Aurora and RDS instance and clusters.
- Collecting CloudWatch metrics

IAM inline policy

Add the JSON policy as **inline policy** for IAM user or role. If you are running the agent on EC2, grant the policy to the IAM role associated with the EC2 instance.

- This policy is applied to both **Aurora and RDS services**.
- The resources are configured to apply to all Aurora and RDS resources. If needed, the resource ARNs can be restricted depending on the security policy.
- The agent can use the read-only APIs only. The roles that can affect the database are not included.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "RDSPermissions",
      "Effect": "Allow",
      "Action": [
        "rds:DescribeDBInstances",
        "rds:DescribeDBClusters",
        "rds:DescribeDBClusterParameters",
        "rds:DescribeOptionGroups"
      ],
      "Resource": "*"
    },
    {
      "Sid": "CloudWatchMetricsPermissions",
      "Effect": "Allow",
      "Action": [
        "cloudwatch:GetMetricData",
        "cloudwatch:GetMetricStatistics",
        "cloudwatch:ListMetrics"
      ],
      "Resource": "*"
    },
    {
      "Sid": "CloudWatchLogsPermissions",
      "Effect": "Allow",
      "Action": [
        "logs:DescribeLogGroups",
```

```

    "logs:DescribeLogStreams",
    "logs:FilterLogEvents",
    "logs:GetLogEvents"
  ],
  "Resource": [
    "arn:aws:logs:*:*:log-group:/aws/rds/*"
  ]
}
]
}

```

Cross-account IAM permissions

- ⓘ Additional IAM permissions are required to configure cross-account monitoring using `account_configs`.

Agent EC2 Instance Profile

Grant the EC2 running the agent permission to assume the role of the target account.

```

{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "sts:AssumeRole",
      "Resource": [
        "arn:aws:iam::111111111111:role/DBXCRole",
        "arn:aws:iam::222222222222:role/QARole"
      ]
    }
  ]
}

```

Target account Cross-Account Role

Create a role in the target account that allows access from the agent account.

- **Trust Relationship:** Allow the agent account



```
{
  "Principal": {
    "AWS": "arn:aws:iam::<agent-account-ID>:role/DBXC-EC2-Instance-Role"
  },
  "Action": "sts:AssumeRole"
}
```

- **Permission Policy:** RDS/CloudWatch read permissions

```
{
  "Effect": "Allow",
  "Action": [
    "rds:DescribeDBInstances",
    "rds:DescribeDBClusters",
    "cloudwatch:GetMetricData",
    "logs:FilterLogEvents"
  ],
  "Resource": "*"
}
```

NCP Cloud DB for PostgreSQL permissions

The Cloud Agent calls the NCP Cloud Insight API to collect metrics and configuration data from the NCP Cloud DB for PostgreSQL instances. For this purpose, Access Key and Secret Key are required.

How to create an Access Key and Secret Key

1. Log in to the [NAVER CLOUD PLATFORM Console](#).
2. From the top menu, go to **My Page > Account Management**.
3. Select **Authentication Key Management** from the menu.
4. Click the **[Create Key]** button to create new Access Key and Secret Key.

Pricing information

 The Cloud Monitoring Agent is free, but note that separate charges may apply depending on the use of Cloud services.

AWS Pricing

Cloud monitoring agents are provided without separate SaaS fees, and the monitoring feature can be used simply by installing the agent. However, because the metrics are collected through the API for Cloud services such as AWS CloudWatch, **separate charges may apply depending on the CloudWatch usage.**

This agent basically collects metrics by **calling the GetMetricData API every 1 minute**. Note that the charge may increase depending on the call frequency and collection targets.

For more information about the pricing policy, see the [Amazon CloudWatch pricing in AWS official documentation](#) page.

NCP Pricing

Cloud monitoring agents are provided without separate SaaS fees, and the monitoring feature can be used simply by installing the agent. The Cloud Monitoring Agent calls the Cloud Insight API every 5 minutes to collect metrics at 1-minute granularity.

The Cloud Insight service of the Naver Cloud Platform (NCP) is currently provided free of charge, but may become paid depending on the future policy. If the Cloud Insight service becomes paid, charges may apply based on the number of API calls. Monitor the usage accordingly.

For more information about the pricing policy, see the [Official pricing guide of NAVER Cloud Platform](#) page.

-  • `names` and `account_configs` **cannot be used simultaneously**. A validation error occurs at agent startup.
- ARN format is strictly validated. `arn:aws:iam::<12-digit number>:role/<name>` — be careful not to omit the double colons (`::`).



- Each `account_config` requires at least 1 cluster.
- `autoscale.enabled: true` is required for cluster discovery to work.

Installation And Usage

DBX for Cloud, DBXC consists of two main components.

- **dbxc-agent**: A background service (agent) that monitors cloud metrics
- **dbxc-ctl**: A CLI tool to control the agent (start, stop, check status, view logs)

Configuration File Structure

DBXC defines runtime configurations through a YAML configuration file.

Supported Formats

DBXC supports both single-project and multi-project agents.

- YAML format (.yaml, .yml)
- Single file: Run a single agent
- Directory: Run multiple agents

Automatic Search Order for Configuration Files

1. File or directory specified with the `-c` option
2. `config.yaml` , `config.yml` (default files)
3. All `.yaml` , `.yml` files in the current directory

Example Configuration Files

Directory tree example

```
> tree
.
├── configs
```

```

|   ├── project-1.yml
|   └── project-2.yml
└── dbxc-agent
    ├── dbxc-ctl
    ├── dbxc.log
    └── dbxc.pid

> ls -al
total 68456
drwxr-xr-x  8 kyw  staff   256  7 18 20:23 .
drwxr-xr-x  9 kyw  staff   288  7 18 20:10 ..
drwxr-xr-x  4 kyw  staff   128  7 18 20:23 configs
-rwxr-xr-x  1 kyw  staff 20174450  7 18 20:09 dbxc-agent
-rwxr-xr-x  1 kyw  staff  8565570  7 18 20:09 dbxc-ctl
-rw-----  1 kyw  staff 5574908  7 18 20:30 dbxc.log
-rw-r--r--  1 kyw  staff    5  7 18 20:23 dbxc.pid

```

yaml example

```

input:
  csp: "aws" # Enter the cloud service provider.
  namespace: "rds"
  region: "us-east-1"
  instances: # Always collect the specified instances. If you specify an instance inside a cluster here, it will always be collected regardless of autoscale.
    - name: "mysql-rds"
      slow_query: true # Set this to true if you want to use the slow query page.
  clusters:
    autoscale:
      enabled: false # When autoscale is enabled, scaled instances are added to or removed from the collection targets.
      interval: 60 # Autoscale check interval for the specified clusters (unit: seconds).
    names:
      - "database-cluster-name"
  metrics: # Enter the metrics to collect.
    - "CPUUtilization"
    - "FreeStorageSpace"
    - "FreeableMemory"
    - "ReadLatency"
    - "WriteLatency"
    - "ReadIOPS"

```

```

- "WriteIOPS"
- "NetworkReceiveThroughput"
- "NetworkTransmitThroughput"
- "FreeLocalStorage"
logs:
  enabled: false # Can be enabled/disabled. Select true/false.
  groups: # Add the desired AWS log groups.
    - "/aws/rds/cluster/database-cluster-name/error"
    - "/aws/lambda/MyLambda"

output: # Enter the WhaTap information to receive the collected metric data.
license: "abcdefg-higjgkjgjk-zxcvnbmbmc"
host: "127.0.0.1"

```

Agent Execution and Control Commands

The CLI provides various commands for starting the agent, checking its status, and viewing logs.

Start/Stop

```

# Start agent
dbxc-ctl start -c project.yaml    # Specific config file
dbxc-ctl start --config configs   # Multiple projects (directory)
dbxc-ctl start                   # Automatically search for default config file

# Stop agent
dbxc-ctl stop

# Restart agent
dbxc-ctl restart -c project.yaml

```

Check Status

```

# Quick status check
dbxc-ctl status
dbxc-ctl describe

```

View Logs

```
# View logs in real time
dbxc-ctl logs -f

# View logs for a specific project only
dbxc-ctl logs -c project.yaml

# Filter logs by level
dbxc-ctl logs --level error

# View the last 100 logs
dbxc-ctl logs --tail 100
```

If Agent Startup Fails

```
# Check logs
dbxc-ctl logs --level error
```

Check Agent Status

```
# Check all statuses
dbxc-ctl describe

# Check logs for a specific agent
dbxc-ctl logs -c specific-config.yaml -f
```

✓ Common Troubleshooting

- **Port conflict:** Check if the default port 53200 is in use
- **Configuration file errors:** Verify YAML syntax
- **Permission issues:** Check file read permissions
- **PID file lock:** Force stop and delete the PID file

Technical Details

Detailed command structure and internal system behavior of DBXC.

Command Structure

dbxc-ctl supports the following commands:

- **start**: Start DBXC agent
- **stop**: Stop DBXC agent
- **restart**: Restart DBXC agent
- **logs**: View logs
- **status or describe**: Check agent status
- **version**: Check version information
- **help**: Display help

HTTP API Endpoints

The agent provides HTTP endpoints:

- `http://localhost:53200/health` : Health check
- `http://localhost:53200/status` : Status information
- `http://localhost:53200/logs` : Log viewing

Logging System

- File name: dbxc.log
- Maximum size: 10MB per file
- Maximum number of backup files: 7 (dbxc.log.1, dbxc.log.2, ... dbxc.log.7)
- Retention period: 14 days
- Compression: Automatic compression enabled (.gz for old logs)

- Timezone: Local time

When the log file reaches 10MB, it rotates automatically. Up to 7 backup files are retained, and logs older than 14 days are deleted. Old logs are compressed to save disk space.

Log Collection Settings

Database Cloud Agent can periodically collect various logs, including RDS and Aurora instances, using the AWS CloudWatch Logs API.

This document explains how the API affects costs and provides guidance on configuring collection settings appropriately.

Using CloudWatch Logs API and Cost Impact

The DBXC Agent uses the CloudWatch Logs **FilterLogEvents** API to query already stored logs.

The cost incurred is determined mainly by two factors:

First, the **amount of data retrieved**. When the API is called, it fetches logs from CloudWatch Logs within the specified time range and limit (`regular_fetch_limit`). The larger the volume of data retrieved, the higher the cost due to AWS internal processing and data transfer.

Second, the **number of API calls**. If the interval is short or there are many target log groups, frequent API calls occur periodically. Since API calls themselves are billed, frequent calls at short intervals may accumulate costs even if the data volume is small.

As a result, in log collection using FilterLogEvents, the **amount of retrieved data** has a greater impact on costs. However, using **short intervals with many target groups** makes API call costs also significant.

Therefore, for efficient operation, avoid fetching unnecessarily large volumes of logs at once with `regular_fetch_limit` , and adjust the interval according to your environment to control the number of calls.

ⓘ Official References

- AWS API Reference: [FilterLogEvents](#)
- AWS Pricing: [CloudWatch Logs Pricing](#)

Log Collection Mechanism

The DBXC Agent periodically calls the FilterLogEvents API to fetch new logs from specified log groups.

At each call, it queries only logs between the last collection time and the current time, managing collection status separately for each log group.

The last collection time is stored in a local state file, so even after restarting the agent, collection resumes from the previous point.

Recommended Settings by Environment

All settings require specifying the actual log group list in `logs.groups`, and log collection requires `logs:FilterLogEvents` and `logs:DescribeLogGroups` permissions.

General Production Environment (Default)

```
logs:
  groups:
    - "/aws/rds/instance/prod-mysql/error"
    - "/aws/rds/instance/prod-mysql/general"
  interval: 60
  regular_fetch_limit: 1000
```

This is the agent's default configuration, applied even if not explicitly defined in the config file. It prevents unnecessary bulk log retrieval at service startup and stably collects up to 1,000 logs per cycle afterward. With a good balance of real-time performance, stability, and cost efficiency, this setting suits most production environments.

Real-time Oriented Environment

```
logs:
  groups:
    - "/aws/rds/cluster/prod-aurora/audit"
    - "/aws/rds/cluster/prod-aurora/error"
  interval: 30
  regular_fetch_limit: 200
```

Collecting logs every 30 seconds minimizes latency while reducing memory usage by limiting the amount retrieved at once. However, since the number of API calls increases, this setup may raise API call costs and should

be used only for services requiring strong real-time capabilities.

Cost-optimized Environment

```
logs:
  groups:
    - "/aws/rds/instance/mysql/error"
  interval: 300
  regular_fetch_limit: 500
```

Collecting logs every 5 minutes significantly reduces API call frequency and lowers costs. However, it introduces up to a 5-minute delay in log reflection, making it unsuitable for environments where real-time monitoring is critical.

Tuning Guidelines

If API costs are higher than expected, increase the interval or reduce `regular_fetch_limit`. If log loss is a concern, reduce the interval. If memory usage is high, reduce `regular_fetch_limit` or decrease the number of target log groups.

IAM Permission Configuration

```
{
  "Effect": "Allow",
  "Action": [
    "logs:FilterLogEvents",
    "logs:DescribeLogGroups"
  ],
  "Resource": "arn:aws:logs:*:*:log-group:/aws/rds/*"
}
```

Serverless Agent Settings

Home > Select Project > **Management** > **Serverless Agent Settings**

ⓘ The serverless agent is a lightweight collector designed to gather only cloud metrics in a serverless execution environment (such as Lambda or Functions) without requiring a dedicated server or program installation.

It can be easily deployed via AWS CloudFormation or Azure Resource Manager Template, enabling quick setup with minimal management overhead.

This agent provides the same functionality as [Cloud Agent Settings](#), but is more suitable for single-instance environments where you only need to collect metrics.

However, it does **not support log collection**. If you operate multiple instances or require log collection (e.g., via CloudWatch Logs), it is recommended to use the Cloud Agent Settings instead.

It provides the feature to set up additional collection and monitoring for monitoring metrics provided by cloud services on the dashboard of the database project. Through the **Serverless Agent Settings** menu, you can monitor the resource status of the cloud database server in real time to maximize the operational efficiency.

Cloud settings allow you to collect CloudWatch metrics and slow query data. Collection of slow queries is supported only for MySQL and PostgreSQL.

- The configuration process is intuitive and easy, so even non-developers can quickly perform it.
- After completing the configuration, you can immediately check and correct errors through visual feedbacks.
- By creating and configuring roles directly on the CSP console, you can enable granular control over security groups and policies.

ⓘ After configuration, extra charge may be incurred in the cloud depending on the usage, separate from the WhaTap fee. Check the pricing policy of the cloud service in use before installation.

AWS CloudWatch configuration

You can monitor the status and resource usage of the database running in the AWS cloud environment.

Selecting the service to which you want to add metrics

1. Select the service you want to add metrics to.


AWS CloudWatch


Install an AWS CloudFormation template to collect Amazon CloudWatch metrics.


Azure Monitor

Install the Azure Resource Manager template to collect cloud metrics.

In the first section, select **AWS CloudWatch**. Install the **AWS CloudFormation** template to collect the Amazon CloudWatch metric.

Entering parameters

Enter required items before proceeding with the **AWS Cloud Formation** configuration.

2. Enter parameters.

DB identifier

Instance



Interval

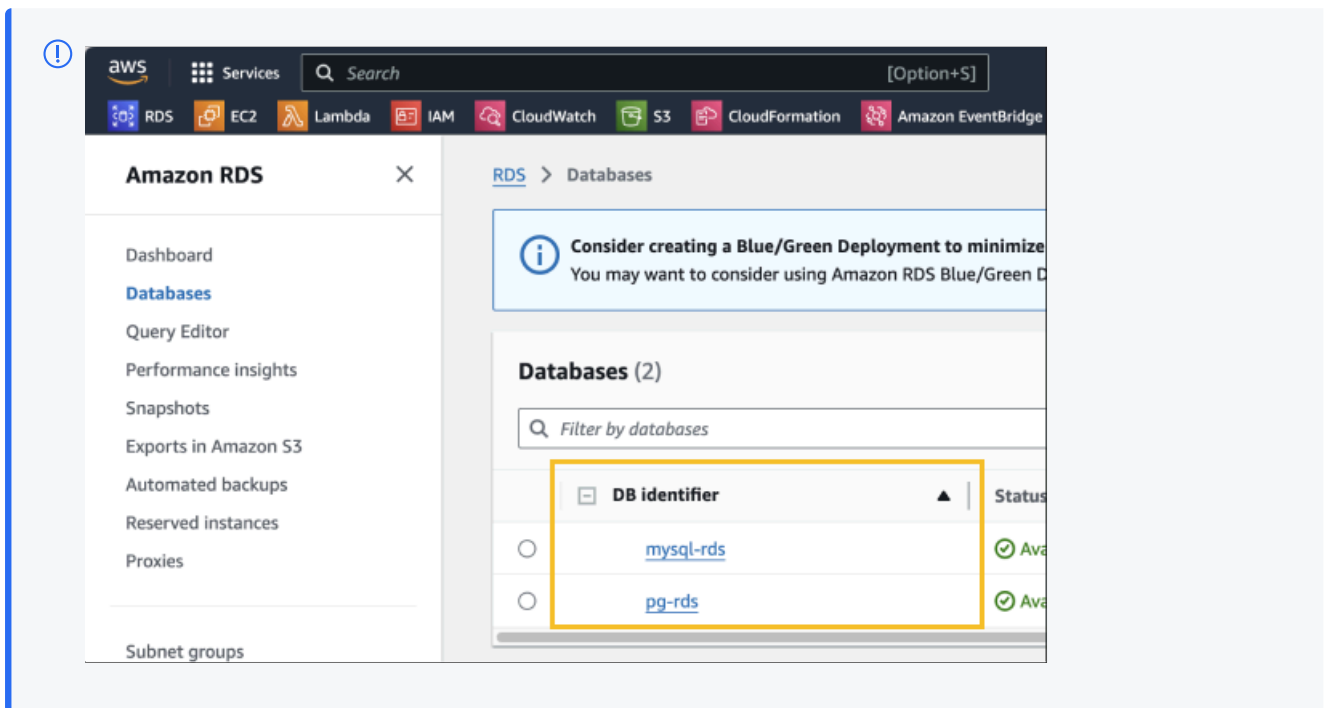


Region ⓘ

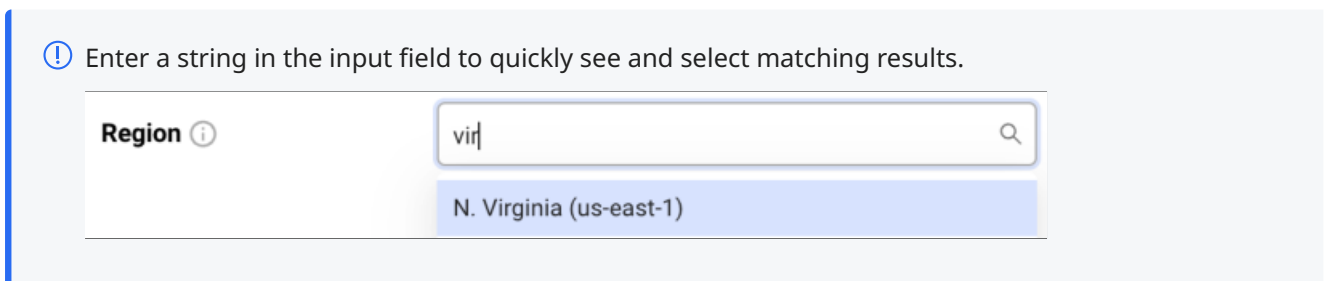


- **DB Identifier:** Enter the identifier of the AWS database instance to monitor.

ⓘ Enter the **DB identifier** value of the Database instance created by the Amazon RDS service.



- **Instance:** Select an instance (agent) to monitor metrics collected from databases running in the AWS cloud environment.
- **Interval:** Select a metric collection interval. (60 sec / 300 sec)
- **Region:** Select a region applied to the desired AWS database to monitor. If a different region is entered, you cannot view the metrics.



Configuring the AWS CloudFormation

Install the **AWS CloudFormation** template provided by WhaTap. This template allows you to collect the CloudWatch metric through WhaTap. If you select **AWS CloudFormation Setting**, the installation starts and the

configuration process is performed.

3. Set AWS CloudFormation.

Collect cloud metrics by installing the AWS CloudFormation template provided by WhaTap. Please refer to the guide link below to install and set up AWS CloudFormation for metrics collection.

[AWS CloudFormation Setting >](#)

1. When you select **AWS CloudFormation Setting**, the **AWS CloudFormation** configuration screen appears.
2. Check the values for the items automatically filled in the **Stack name** and **Parameters** sections.
3. Select **Create stack** located at the bottom of the screen.

- For more information about **AWS CloudFormation**, see [the following link](#).
- For the **Amazon CloudWatch** metrics about **Amazon RDS**, see [the following link](#).

Setting the required roles

To create a **CloudFormation** stack, set the following AWS roles:

Guide to AWS roles (JSON)

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "cloudformation:CreateStack",
        "cloudformation:UpdateStack",
        "cloudformation>DeleteStack",
        "cloudformation:DescribeStacks",
        "cloudformation:DescribeStackResources",
        "cloudformation:GetTemplateSummary",
```



```
"cloudformation:DescribeStackEvents",
"cloudformation:ListStacks",
"cloudformation:ListStackResources"
],
"Resource": "*"
},
{
  "Effect": "Allow",
  "Action": [
    "lambda:CreateFunction",
    "lambda:UpdateFunctionCode",
    "lambda:UpdateFunctionConfiguration",
    "lambda:InvokeFunction",
    "lambda:DeleteFunction",
    "lambda:GetFunction",
    "lambda:GetFunctionConfiguration",
    "lambda:ListFunctions"
  ],
  "Resource": "*"
},
{
  "Effect": "Allow",
  "Action": [
    "logs:CreateLogGroup",
    "logs:CreateLogStream",
    "logs:PutLogEvents",
    "logs:DescribeLogGroups",
    "logs:DescribeLogStreams",
    "logs:GetLogEvents",
    "logs:FilterLogEvents"
  ],
  "Resource": "*"
}
{
  "Effect": "Allow",
  "Action": [
    "events:PutRule",
    "events:PutTargets",
    "events:RemoveTargets",
```



```

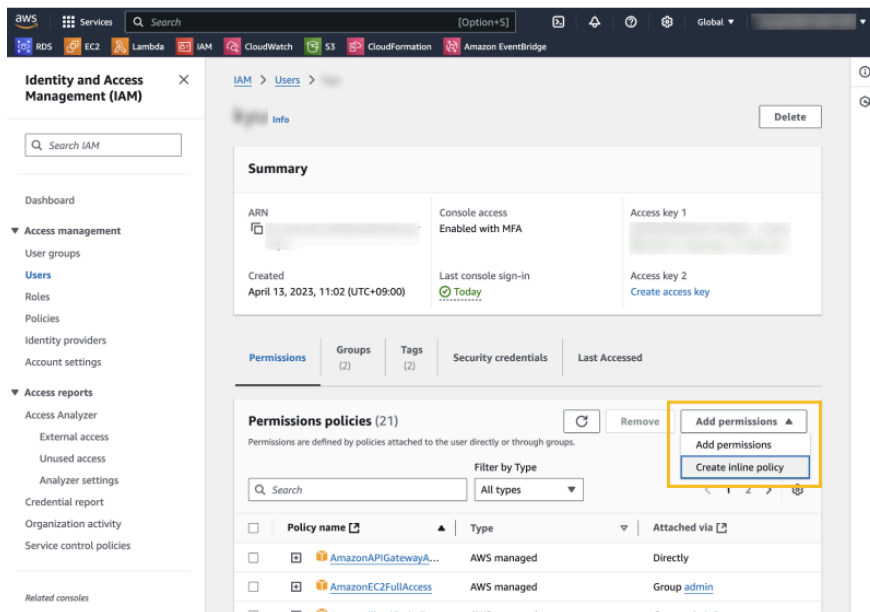
    "events:DeleteRule",
    "events:DescribeRule",
    "events:EnableRule",
    "events:DisableRule"
  ],
  "Resource": "*"
},
{
  "Effect": "Allow",
  "Action": [
    "iam:CreateRole",
    "iam:AttachRolePolicy",
    "iam:PutRolePolicy",
    "iam:GetRole",
    "iam:DeleteRole",
    "iam:PassRole",
    "iam:ListRoles"
  ],
  "Resource": "*"
},
{
  "Effect": "Allow",
  "Action": [
    "cloudwatch:GetMetricData"
  ],
  "Resource": "*"
},
{
  "Effect": "Allow",
  "Action": [
    "s3:GetObject"
  ],
  "Resource": "arn:aws:s3:::repo.whatap.io/agent/db/WhaTapRDSMonitoring.zip"
},
{
  "Effect": "Allow",
  "Action": [
    "scheduler:GetSchedule",
    "scheduler:ListSchedules",

```

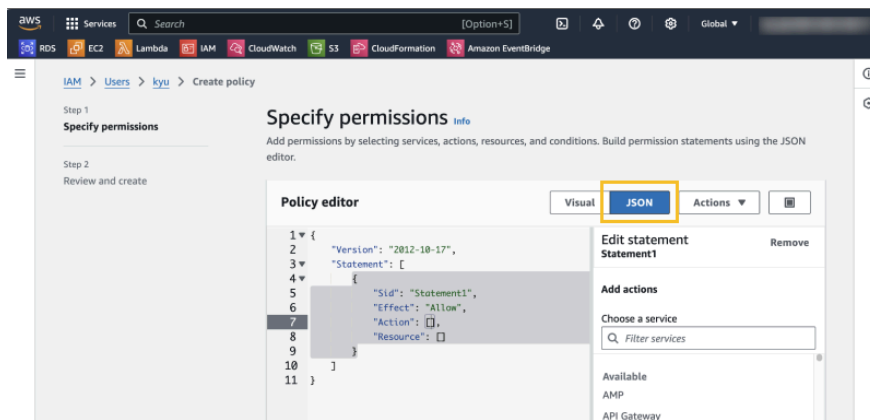



```
"scheduler:CreateSchedule"
},
"Resource": "*"
}
}
}
```

1. Copy the JSON content from **Guide to AWS roles**.
2. In IAM, select a **user** that will create the **CloudFormation** stack.
3. Select **Create inline policy**.



4. After selecting **JSON**, paste the copied roles to register the policy.

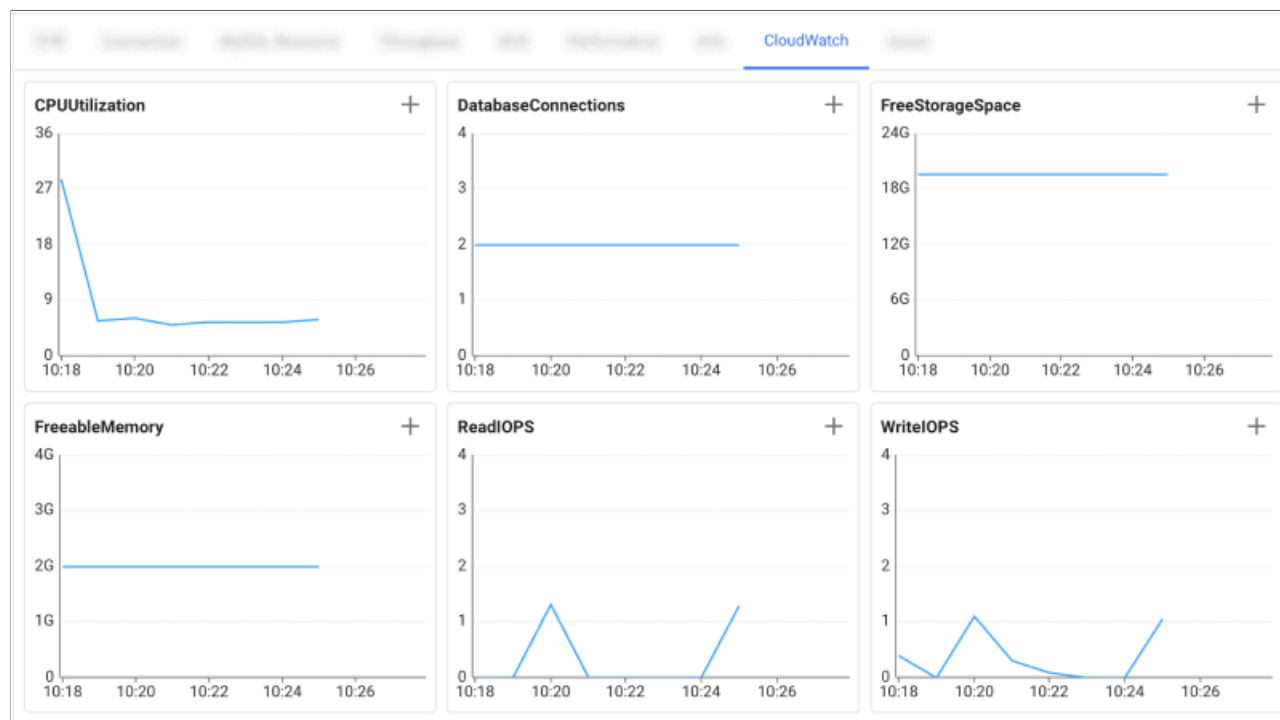


Checking the collection metrics

After completing the cloud data connection, the **Added cloud services** section is generated on the screen. You can see a list of instances (agents) connected to the cloud database and the metrics being collected.

Added cloud services		
Appears when cloud metrics have been collected within the last 5 minutes.		
installed agent	cloud service	Collection indicators
	AWS CloudWatch	CPUUtilization, DatabaseConnections, FreeStorageSpace, FreeableMemory, ReadIOPS, WriteIOPS
	AWS CloudWatch	CPUUtilization, DatabaseConnections, FreeStorageSpace, FreeableMemory, ReadIOPS, WriteIOPS

The metrics collected from **AWS CloudWatch** fall into the **CloudWatch** category.



You can check in the following paths.

- **Dashboard > Monitoring a Database Instance:** You can select metrics by selecting the ⚙ button on the widget.
- **Dashboard > Monitoring Multiple Instances:** You can select metrics by adding custom widgets.

- For more information on how to change the metrics of a widget in the **Monitoring a Database Instance** menu, see [the following](#).
- For more information on how to add custom widgets in the **Monitoring Multiple Instances** menu, see [the following](#).
- In **Analysis > Metrics Search**, select the **Category** entry with `db_aws_rds`. You can check the related tags and fields.

Azure Monitor configuration

You can monitor the status and resource usage of the database running in the Azure cloud environment.

Selecting the service to which you want to add metrics

1. Select the service you want to add metrics to.

 **AWS CloudWatch**
 Install an AWS CloudFormation template to collect Amazon CloudWatch metrics.

 **Azure Monitor** ✓
 Install the Azure Resource Manager template to collect cloud metrics.

In the first section, select **Azure Monitor**. Install the **Azure Resource Manager** template to collect metrics provided by Azure Monitor.

Entering parameters

Enter required items before configuring the **Azure Resource Manager**.

2. Enter parameters.

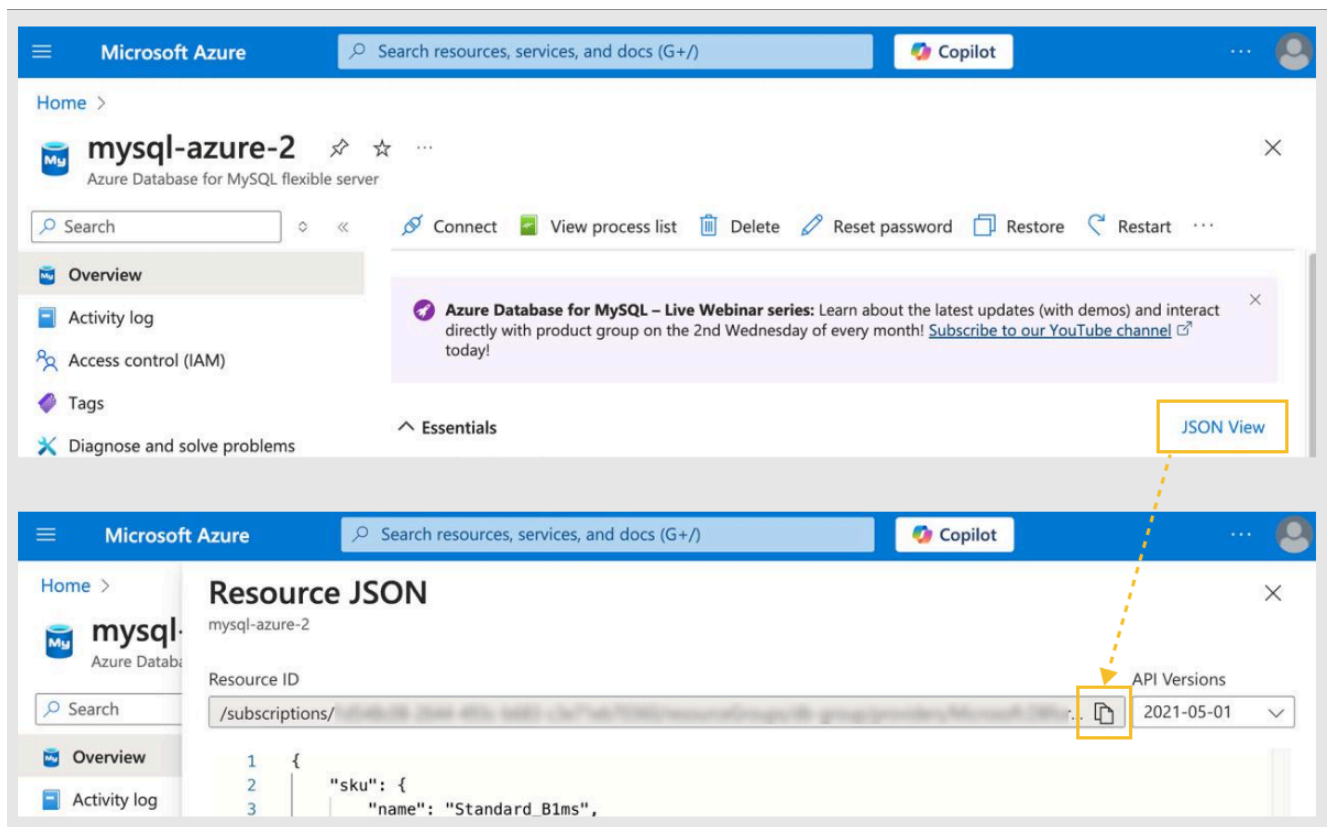
Instance

Resource ID ⓘ

resourceID must be checked and entered in the monitored database console.

JSON generation

- **Instance:** Select an instance (agent) to monitor metrics collected from databases running in the Azure cloud environment.
- **Resource ID:** Enter the resource ID of the database to monitor. This information can be viewed on the monitored database console. See the following image.



Enter the parameters and then select **Create JSON**. It generates the JSON to configure the **Azure Resource Manager**. Copy the generated JSON content.

Configuring the Azure Resource Manager

Install the **Azure Resource Manager** template provided by WhaTap. This template allows you to install the Function App that collects Azure Monitor metrics by the WhaTap server. If you select **Azure Resource Manager Setting**, the installation starts and the configuration process is performed.

3. Set Azure Resource Manager.

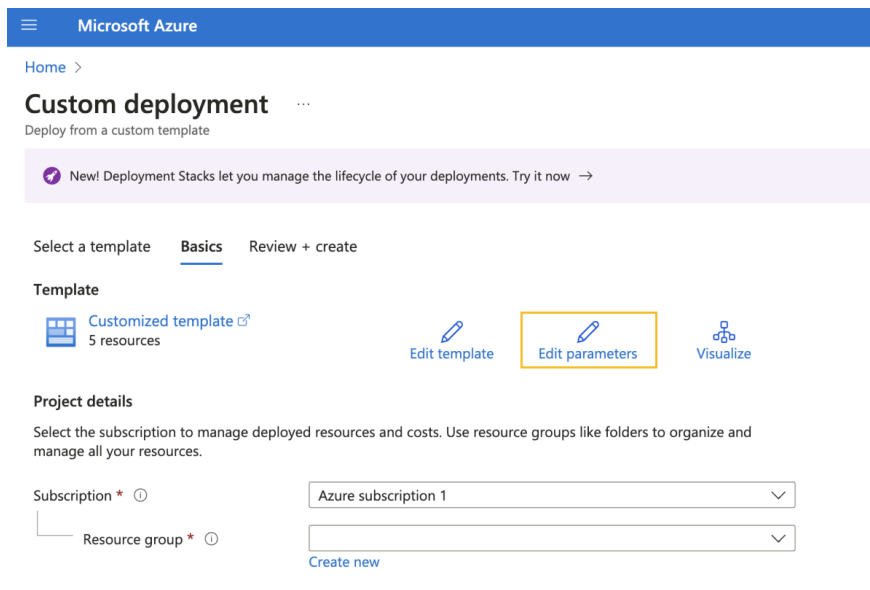
You can use the Azure Resource Manager (ARM) template provided by WhaTap to install a Function App that collects Azure Monitor metrics to the WhaTap server. Please refer to the guide below to set up metrics collection through ARM template and install the Functions App.

[Azure Resource Manager Setting >](#)

1. If you select **Azure Resource Manager Setting**, the template creation screen for Microsoft Azure

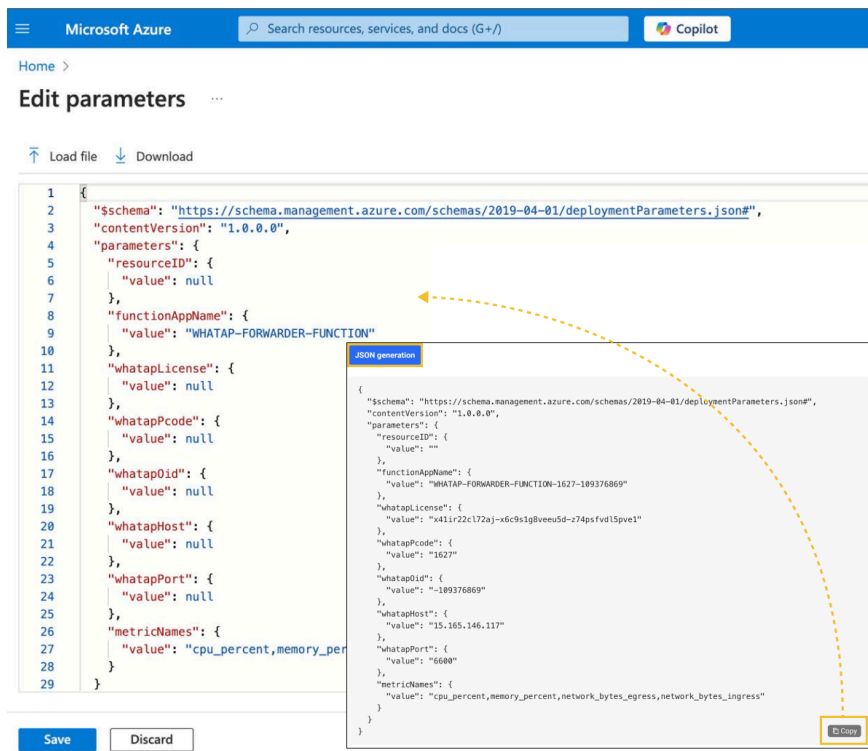
appears.

2. Select **Edit Parameters**.



The screenshot shows the Microsoft Azure portal interface for a custom deployment. The top navigation bar is blue with the Microsoft Azure logo. Below it, the breadcrumb 'Home >' is visible. The main heading is 'Custom deployment' with a subtitle 'Deploy from a custom template'. A purple banner below the heading says 'New! Deployment Stacks let you manage the lifecycle of your deployments. Try it now →'. The 'Select a template' section has three tabs: 'Select a template', 'Basics' (which is active), and 'Review + create'. Under the 'Basics' tab, there is a 'Template' section showing a 'Customized template' with '5 resources'. Below this, there are three buttons: 'Edit template', 'Edit parameters' (which is highlighted with a yellow box), and 'Visualize'. The 'Project details' section follows, with instructions to 'Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.' It contains two dropdown menus: 'Subscription *' (set to 'Azure subscription 1') and 'Resource group *' (empty). A 'Create new' link is visible below the resource group dropdown.

3. Paste the JSON format parameter created through the **Create JSON** button above.



4. When you select **Save** at the bottom of the screen, the parameters are automatically entered.
5. Select **Resource group**. For **Resource group**, select the group that contains the database to be monitored.

Azure

6. Select **Review + Create**.
7. In the following screen, select **Create**.

The deployment is processed and all processes are completed.

- ! • For more information about **Azure Resource Manager**, see [the following link](#).
- For information about metrics supported in Azure Monitor, see [the following link](#).

Setting the required roles

Guidance on roles

In order for WhaTap's Function App to properly use the monitoring, the **Monitoring Reader** role must be assigned to the corresponding resource group or subscription. However, assigning this role itself is a task that requires authorization. It requires the **Microsoft.Authorization/roleAssignments/write** role, which is not included in general user roles.

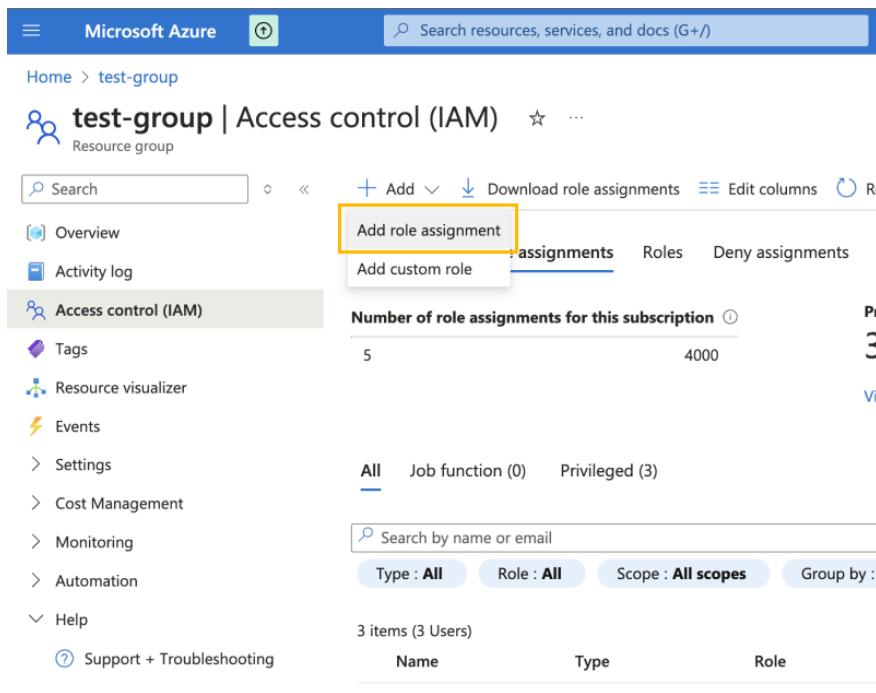
Therefore, to assign the **Monitoring Reader** role, you must have the **Role Based Access Control Administrator** or **User Access Administrator** role, which includes the advanced roles.

- If the **Monitoring Reader** role was already assigned to the resource group or subscription, you can skip this step.
- For more information about the Azure roles assignment, see [the following link](#).

Assigning the roles

For the minimum scope, guidance is provided based on the resource group that is a lower level than subscription.

1. On the group dashboard that is to contain the resources, select **Access control (IAM) > Add > Add role assignment** in order.



2. On the **Add role assignment** screen, select the **Privileged administrator roles** tab.

Microsoft Azure  Search resources, services, and docs (G+/)

Home > test-group | Access control (IAM) >

Add role assignment

A role definition is a collection of permissions. You can use the built-in roles or you can create your own custom roles. [Learn more](#)

Job function roles **Privileged administrator roles**

Grant privileged administrator access, such as the ability to assign roles to other users.

 Can a job function role with less access be used instead?

Search by role name, description, permission, or ID

Type : All Category : All

Name ↑↓	Description ↑↓	Type ↑↓	Category ↑↓
Owner	Grants full access to manage all resources, including ...	BuiltInRole	General
Contributor	Grants full access to manage all resources, but does ...	BuiltInRole	General
Access Review Operat...	Lets you grant Access Review System app permissio...	BuiltInRole	None
Role Based Access Co...	Manage access to Azure resources by assigning role...	BuiltInRole	None
User Access Administr...	Lets you manage user access to Azure resources.	BuiltInRole	General

Showing 1 - 5 of 5 results.

Review + assign Previous Next

3. Select **Role base Access Controle Administator** or **User Access Administrator**.
4. Select **Next** at the bottom of the screen.
5. In the **Assign access to** section, select **User, group, or service principal**, and then select **Select Members** to add users (**User**).

Microsoft Azure

Search resources, services, and docs (G+)

Home > test-group | Access control (IAM) >

Add role assignment

Role **Members** Conditions Review + assign

Selected role User Access Administrator

Assign access to ☒ User, group, or service principal
☐ Managed identity

Members + Select members

Name	Object ID	Type
Thomas.Henderson	a1b2c3d4-5678-9010-1111-222222222222	User

Description Optional

Review + assign Previous Next

6. Select **Next** at the bottom of the screen.
7. Select **Recommended** items and then click **Review + assign**.

Microsoft Azure

Home > test-group | Access control (IAM) >

Add role assignment

Role Members **Conditions** Review + assign

Selected role
User Access Administrator

What user can do

- ☐ Allow user to only assign selected roles to selected principals (fewer privileges)
- ☒ Allow user to assign all roles except privileged administrator roles Owner, UAA, RBAC (Recommended)
- ☐ Allow user to assign all roles (highly privileged)

Condition

Preconfigured (All roles except privilege built-in roles)

Exclude roles:
Owner, User Access Administrator, Role Based Access Control Administrator

Review + assign Previous Next

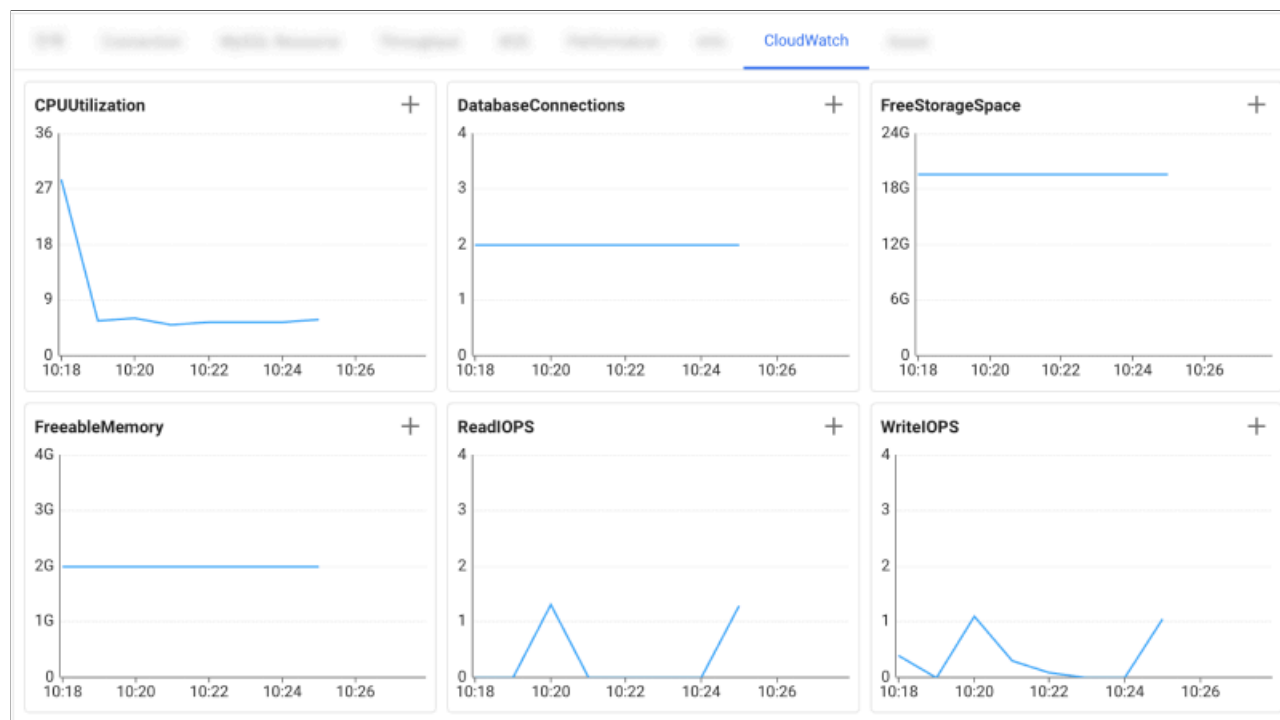
! To select the **fewer privileges** items, allow only the **Monitoring Reader** role.

Checking the collection metrics

After completing the cloud data connection, the **Added cloud services** section is generated on the screen. You can see a list of instances (agents) connected to the cloud database and the metrics being collected.

Added cloud services		
Appears when cloud metrics have been collected within the last 5 minutes.		
installed agent	cloud service	Collection indicators
My-123-monitor	Monitor	cpu_percent, memory_percent, network_bytes_egress, network_bytes_ingress
My-123-monitor	Monitor	cpu_percent, memory_percent, network_bytes_egress, network_bytes_ingress

Metrics collected from **Azure Monitor** are included in the [Azure](#) category.



You can check in the following paths.

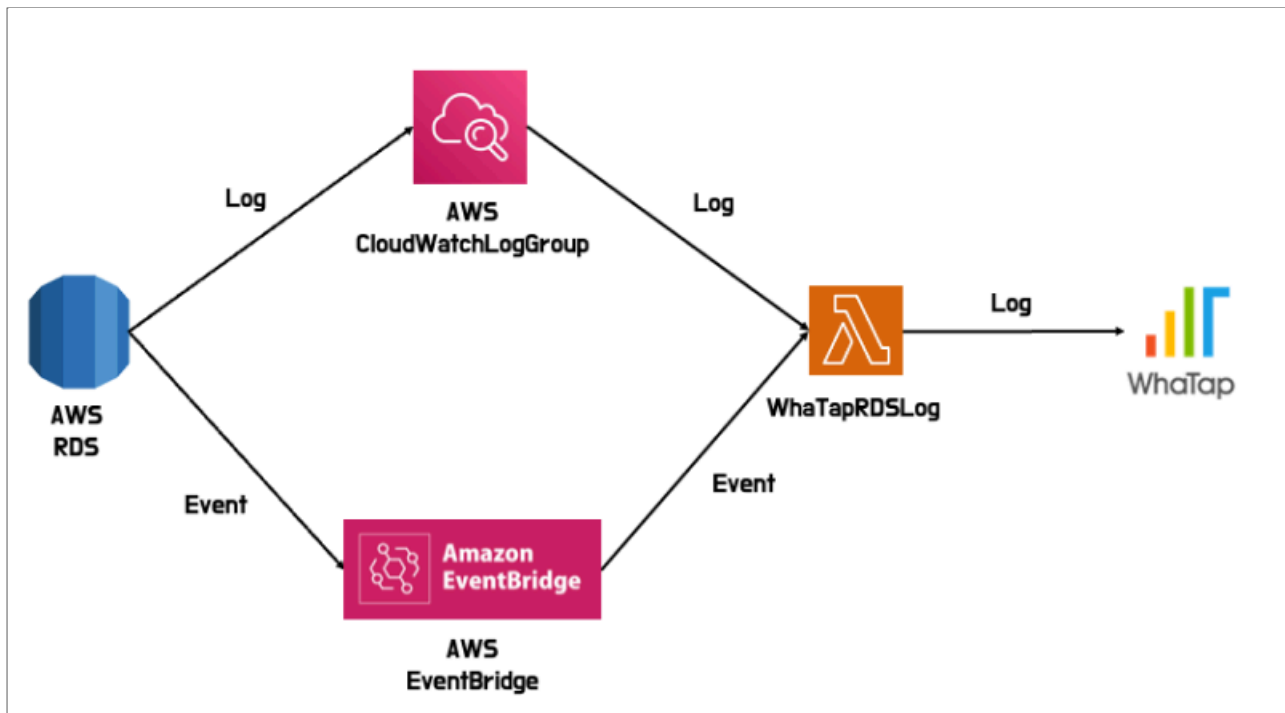
- **Dashboard > Monitoring a Database Instance:** You can select metrics by selecting the ⚙ button on the widget.
- **Dashboard > Monitoring Multiple Instances:** You can select metrics by adding custom widgets.

- For more information on how to change the metrics of a widget in the **Monitoring a Database Instance** menu, see [the following](#).
- For more information on how to add custom widgets in the **Monitoring Multiple Instances** menu, see [the following](#).
- In **Analysis > Metrics Search**, select **Category** as `db_azure_database`. You can check the related tags and fields.

Serverless Log Collection Settings

Applying to the AWS RDS logs

Configuration for monitoring AWS RDS logs



AWS provides various information for RDS monitoring. To check information, directly access the AWS console. However, WhaTap's **AWS RDS Log** provides logs and events from RDS in **Log** under Database Monitoring.

WhaTap's **AWS RDS Log** can be installed through **AWS CloudFormation**. For more information about **AWS CloudFormation**, see [the following link](#).

Checking the roles

To install WhaTap's **AWS RDS Log** via **AWS CloudFormation**, first check the required IAM roles.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "cloudformation:CreateStack",
        "cloudformation:UpdateStack",
        "cloudformation>DeleteStack",
        "cloudformation:DescribeStacks",
        "cloudformation:DescribeStackEvents",
        "cloudformation:DescribeStackResource"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "s3:CreateBucket",
        "s3:PutBucketVersioning",
        "s3:PutBucketPublicAccessBlock"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "lambda:CreateFunction",
        "lambda:UpdateFunctionCode",
        "lambda:UpdateFunctionConfiguration",
        "lambda>DeleteFunction",
        "lambda:AddPermission"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "iam:CreateRole",
        "iam:GetRole",
        "iam:AttachRolePolicy"
      ]
    }
  ]
}

```

```

],
  "Resource": "*"
}
]
}

```

Installation and configuration

1. It can be installed using the **AWS CloudFormation** template provided by WhaTap. Access the following link and then enter the following items in the **Parameters** section.

[AWS CloudFormation Install](#)

- **Host:** Address of the WhaTap collection server. It can be viewed in **Management > Agent Installation** under the WhaTap project. e.g. 13.124.11.223 / 13.209.172.35
- **MemorySize:** WhaTapRDSLog's memory size. Enter a value between 128 MB and 10,240 MB.
- **Pcode:** Project code of a WhaTap project. Check the **pcode** in **Management > Project Management**.
- **ProjectAccessKey:** Access key of a WhaTap project. Check the **Project access key** in **Management > Project Management**.
- **TimeOut:** Maximum time in which the installed WhatTapRDSLog is maintained while running.
- **UseReservedConcurrency:** Whether or not to use the reserved concurrency for WhatTapRDS.
- **ReservedConcurrency:** Maximum number of instances that WhatapRDS can run at the same time when using the reserved concurrency.

2. Access the following link for log monitoring configuration details.

[AWS CloudFormation Setting](#)

3. In the **Parameters** section, enter the following:
 - **AwsRdsNames:** List of names of RDS clusters and instances to receive events
 - e.g. database-1, seoul-pro-db-01, seoul-pro-db-01-writer ...
 - Enter **none** if you do not want to create an event rule.

- **RdsLogGroupName (1-9):** RDS CloudWatchLogGroup name from which you want to receive logs
 - e.g. /aws/rds/seoul-pro-db-01
 - Enter `none` if you do not want to create a subscription filter.

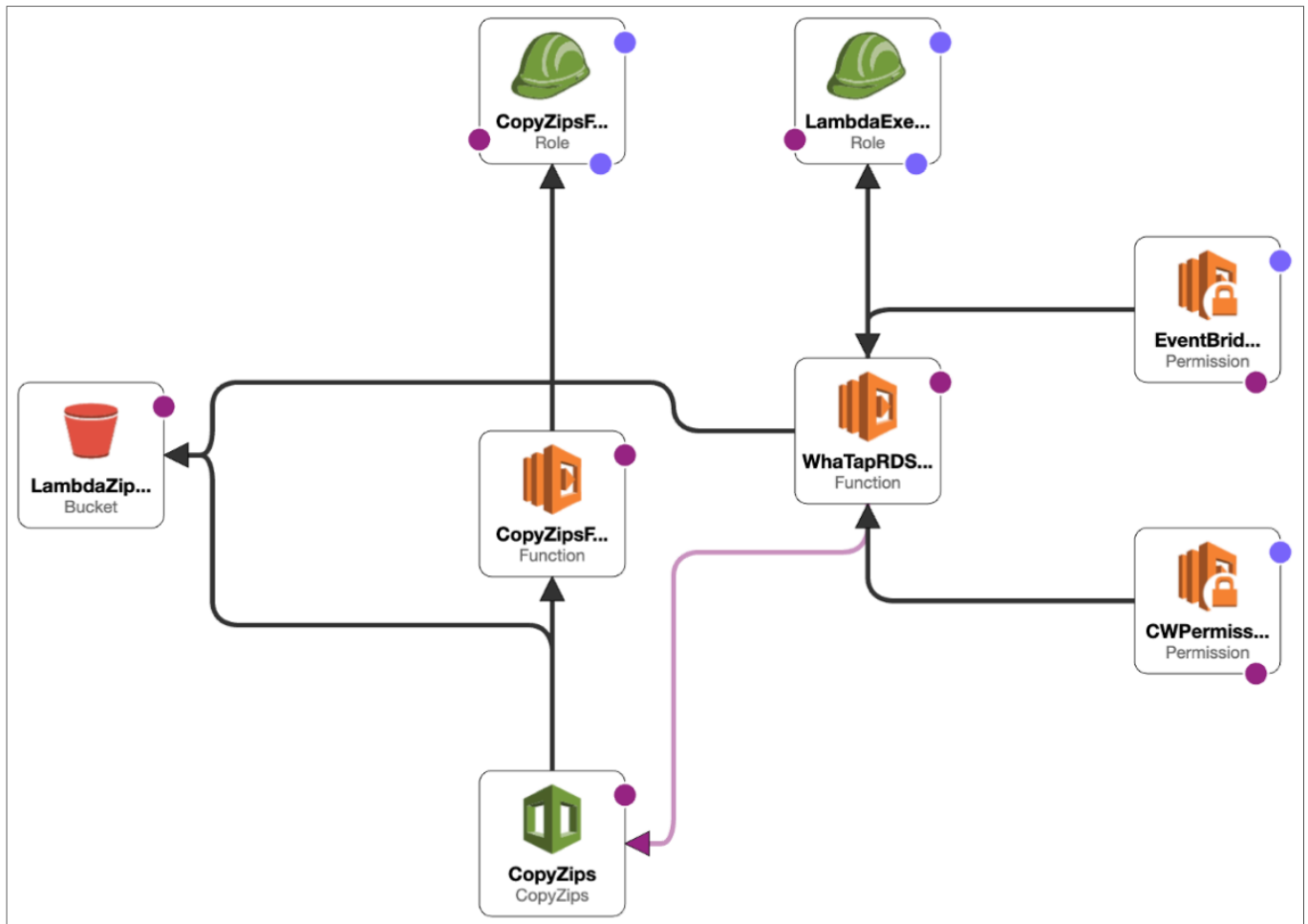
! You can connect up to 9 CloudWatch log groups.

- **WhaTapRDSLogArn:** WhataTapRDSLogArn generated in the previous step

Outputs (1)		
Search outputs		
Key	Value	Description
WhaTapRDSLogArn	arn:aws:lambda:ap-northeast-2:511937781722:function:whataprdseventforwarder-WhaTapRDSLog-nLFyYmp1Od3s	WhaTapRDSLog's Arn

Checking the created resourced

Check the resources installed through **AWS CloudFormation**.



- **S3**

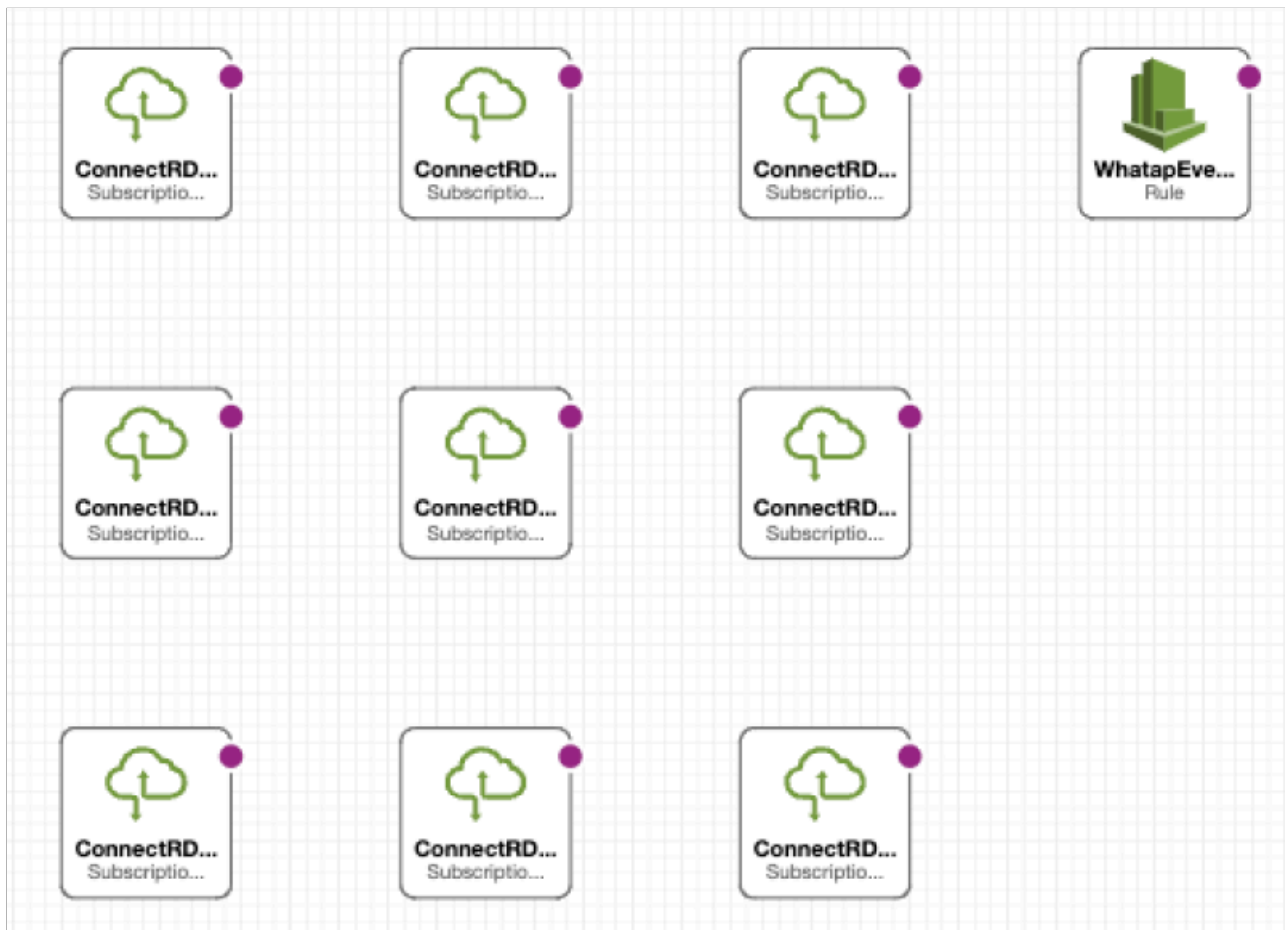
- Bucket: AWS S3 bucket (CopyZips) that stores executable files of WhaTap **AWS RDS Log**

- **Lambda**

- Function
 - CopyZipsFunction: AWS Lambda function that moves executable files from the WhaTap AWS S3 bucket to your environment
 - WhaTapRDSLog: AWS Lambda function for AWS RDS monitoring
- Permission
 - CWPermission: Permission to run WhataTapRDSLog when logs are created in RDS CloudWatchLogGroup

- EventBridgePermission: Permission to run WhataTapRDSLog in EventBridge when RDS events occur
- Role
 - Role required for CopyZipsFunction to get files from WhaTap's S3 and put them into CopyZips
 - Role required for WhatTapRDSLog to leave logs in CloudWatchLogGroup

Checking the filters and rules



• WS CloudWatchLogGroup

SubscriptionFilter (1-9): Filter that executes WhatapRDSLog when logs come into the CloudWatch LogGroup of RDS.

- **AWS EventBridge**

Rule: Rule that triggers WhatapRDSLog when RDS events occur among AWS events

Management

Agent Update

Updating to the latest version

Home > Select Project > **Management** > **Agent Update**

Agent Update

Agent update guide

When you click the 'Update' button, the latest agent file will be downloaded to the agent installation location and automatically restarted. The update process takes approximately 1 minute. Once complete, the agent version will change to the latest version.

ⓘ The agent installed in the Windows environment does not support a self-update. [Show more](#)

ⓘ If the update is not performed properly, please refer to [Whatap Docs](#).

Agent list

Latest agent version 2.3.3 [Update all agents to the latest version](#) [Previous version](#)

no.	id	ip	name	agent path	version	update
1	704779695	192.168.1.100	Agent-0001	C:\Program Files\Whatap\agent\agent.jar	2.2.8 20241212	Update
2	1652207373	192.168.1.100	Agent-0002	C:\Program Files\Whatap\agent\agent.jar	2.2.8 20241212	Update
3	755606486	192.168.1.101	Agent-0003	C:\Program Files\Whatap\agent\agent.jar	2.1.4 20241101	Update

When you update the agent, the latest version of the agent file is automatically downloaded to the agent installation path ([agent path](#)) and it is restarted. The update takes about 1 minute. During an update, the download and update success messages appear on the screen step by step. After an update, the agent is automatically changed to the latest version.

- For the agents that can be updated, the **Update** button is enabled.
- To update agents individually, select **Update** to the right of each agent.
- To update all agents at a time, select **Update all agents to the latest version**.

✔ If you move to another menu while updating an agent, you cannot check its progress. While updating an agent, you cannot update another agent at the same time.

ⓘ The agents installed in the Windows environment do not support the automatic update. Manually reinstall the latest agent or place the latest [whatap.agent.dbx-*.jar](#) file in the previous agent folder and

❗ register the service again.

1. Delete Windows service

```
install_WindowsService.bat delete WhatapDBXAgent # existing service name
```

2. Create Windows service

```
install_WindowsService.bat create WhatapDBXAgent # existing service name
```

In case of update failure

In case of no agent restart

If the agent is not automatically changed to the latest version after display of a successful download message, manually restart the agent. Run the [stop.sh](#) and [startd.sh](#) batch files sequentially in the agent installation path ([agent path](#)).

Manual download of the agent

If downloading is not possible, proceed with agent installation in the following order:

1. Go to **Management > Agent Installation**.
2. Select **Download** in 2 of the **Install guide** section to download the latest version of the agent.
3. Unzip the downloaded file.
4. Upload only the [whatap.agent.dbx-X.Y.Z.jar](#) file to the agent installation path ([agent path](#)).
5. Run the [stop.sh](#) and [startd.sh](#) batch files to restart the agent.

Returning to the previous version

1. On the upper right of the screen, select **Previous version**.
2. The list of downloadable agents appears.
3. Select a desired version.
4. Select **Download** for the agent to return to the selected version.

5. Download the file to the installation path of the selected agent.
6. Delete the `whatap.agent.dbx-X.Y.Z.jar` file of the higher version from the agent installation path.
7. Run the `stop.sh` and `startd.sh` batch files to restart the agent.

ⓘ Because the batch file is set to run the highest version of the agent in the installation path, delete the higher version of JAR file to run the previous version of the agent.

ⓘ The minimum version of the DBX agent to use the PostgreSQL monitoring is 1.6.7.

Agent deletion

Deleting from the server where the agent have been installed

- [Linux](#) ~~Windows~~

Among the Java processes running on the server, the process containing `whatap.agent.dbx` string is terminated. Run `stop.sh` in the path where the agent has been installed. Afterwards, delete the agent file and user roles registered in the agent.

Among the Java processes running on the server, find the process running with the jar file of the agent file name and then stop it. If the batch file is running in the foreground, end the batch execution script. 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.
- You can check the active sessions (including queries), lock information, and process information. To apply both the public cloud and private cloud, the products are configured for query-based monitoring.
- 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 a Database Instance

While you can monitor only one instance in **Monitoring a Database Instance**, **Monitoring Multiple Instances** allows you to see and compare the metrics for multiple instances on a screen. You can see the status of instances through grouped values for various key instance metrics. Using the graph charts, you can easily see the trend of instance metric values and which instances have higher values. It provides information of the longest lasting active session. For more information, see [the following](#).

Slow Query

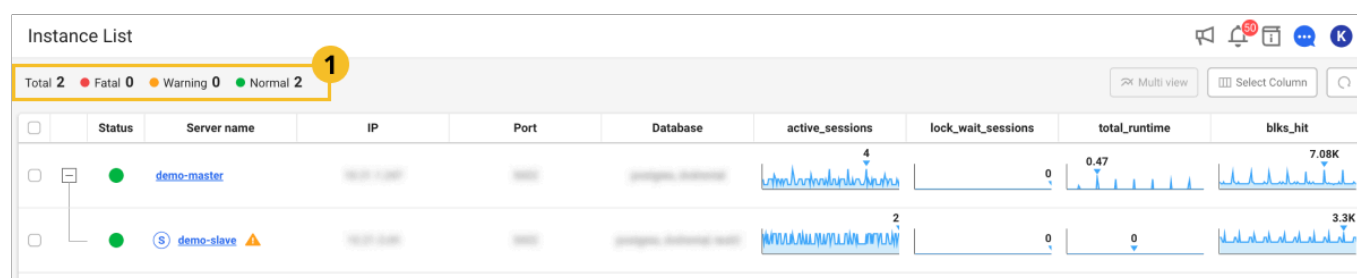
It provides the response time distribution by SQL. Through the response distribution, you can intuitively check the status of delayed SQL and identify the cause of failure by specifying the target. For more information, see [the following](#).

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.

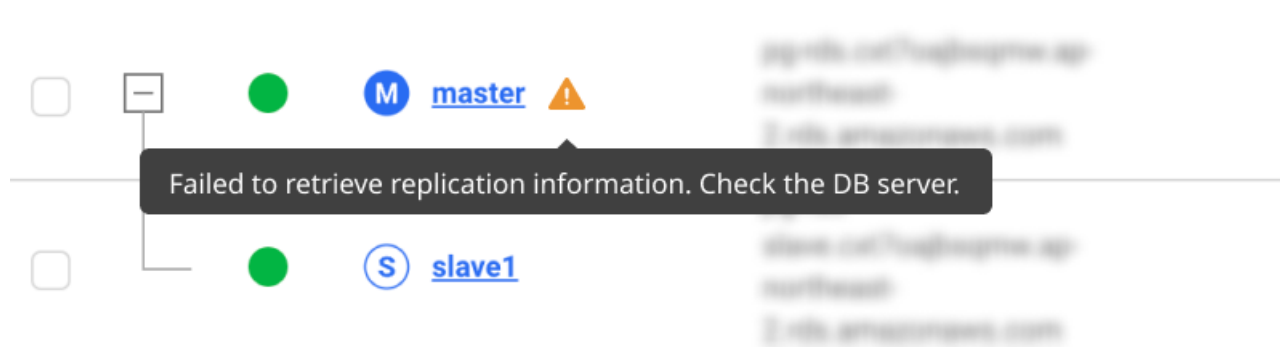
! For more information about **Alert**, see [the following](#).

- 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.

① 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](#).
- In the **Instance list**, **M**, **S**, and **C** displayed in the **Server name** column mean Master, Slave, and Cluster respectively. It is displayed in case of the DB composed as replication.

If the replication instance is abnormal, the  icon appears.



- 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](#).

① For more information about the fixed area on the top of the screen, see [the following](#).

- ✓ In **Instance list**, to change **Server name**, use the `object_name` option in the agent configuration file (`whatap.conf`). For more information, 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

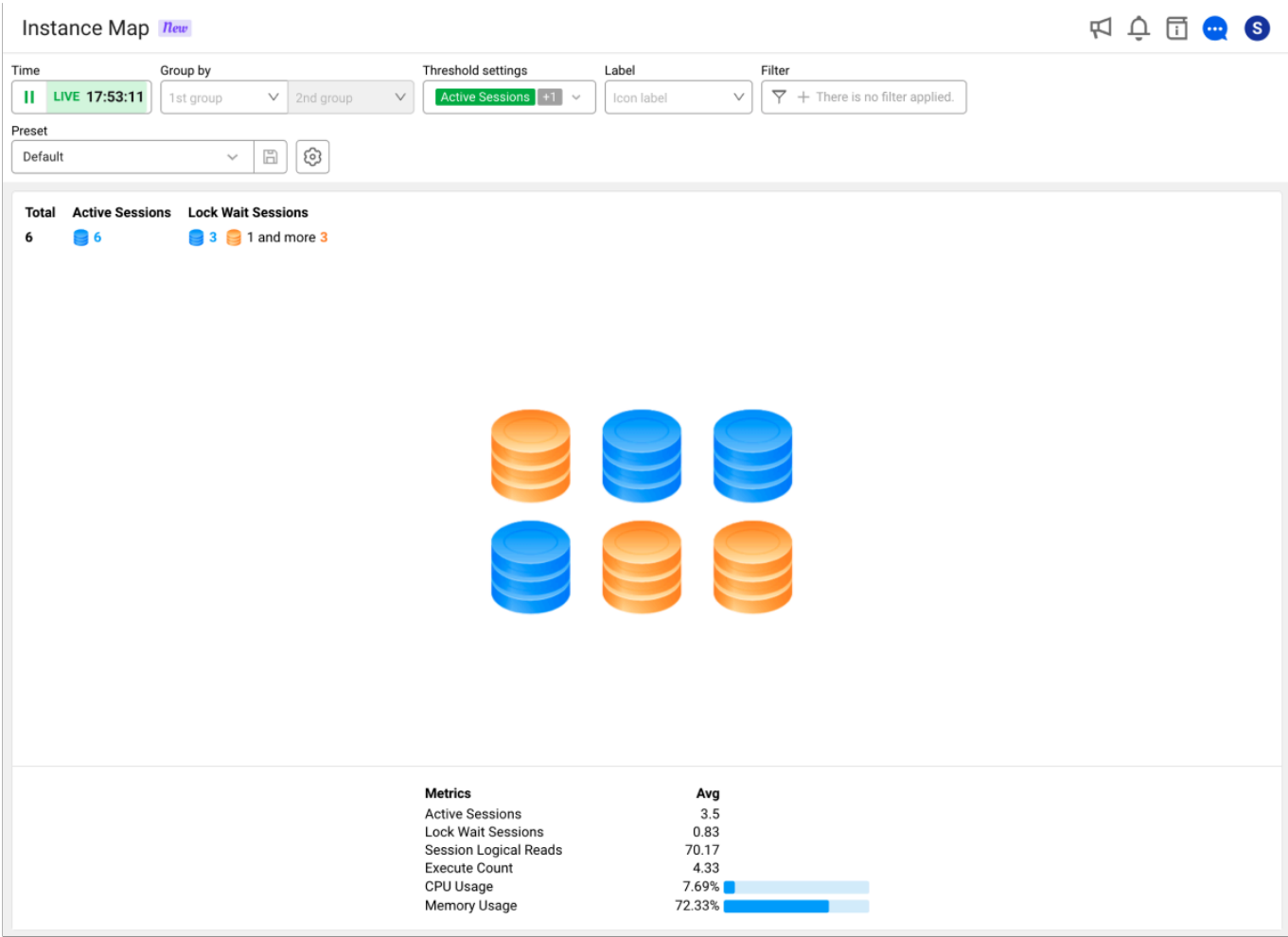
You can display or hide the performance metrics columns in the instance list. Select **Select Column** on the upper right of the screen. Check or uncheck the desired items. You can display or hide the columns.

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.

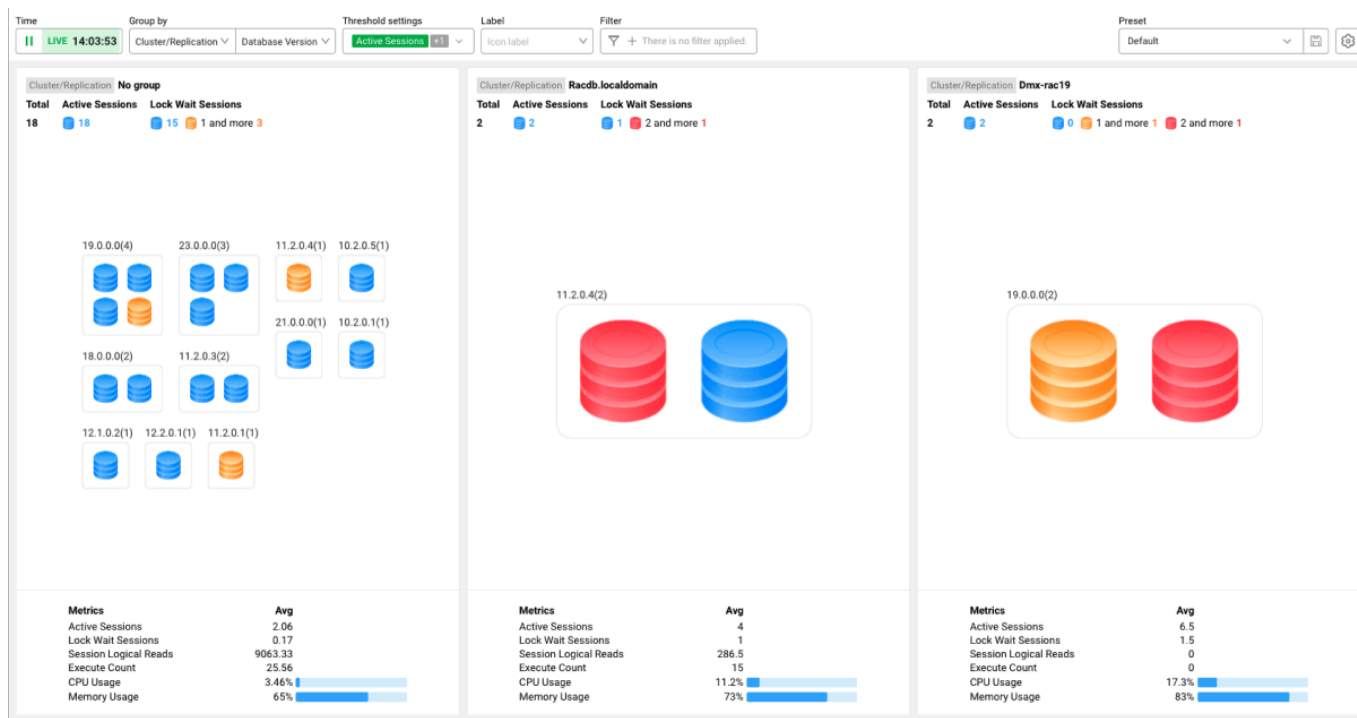
By setting thresholds on key metrics of database servers, you can grasp the status of each instance at a glance and monitor active sessions in real time. You can also navigate to screens such as Instance Monitoring or Multi-Instance Monitoring to quickly analyze databases when an issue occurs.



! 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.

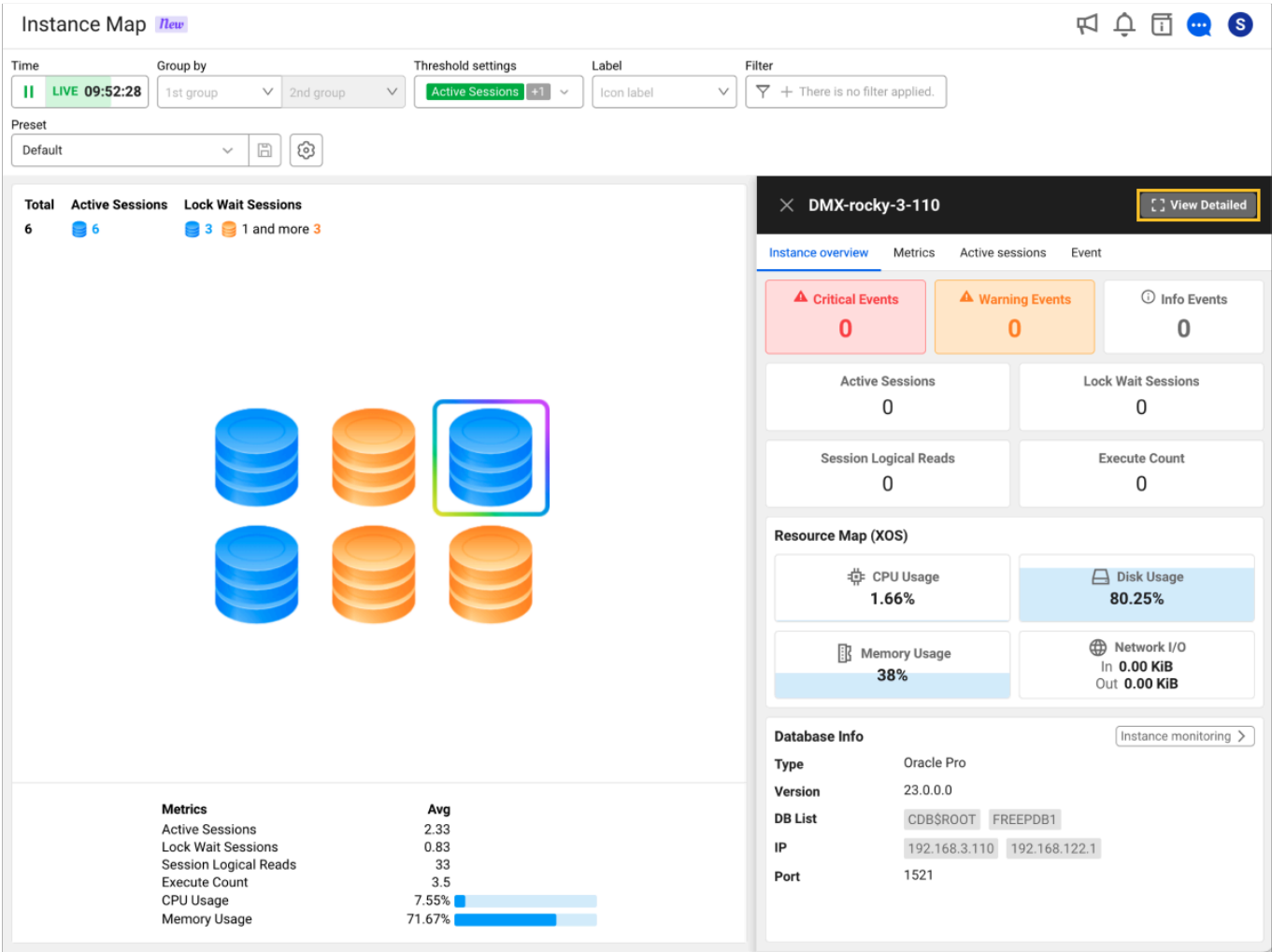
Bottom Section

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.



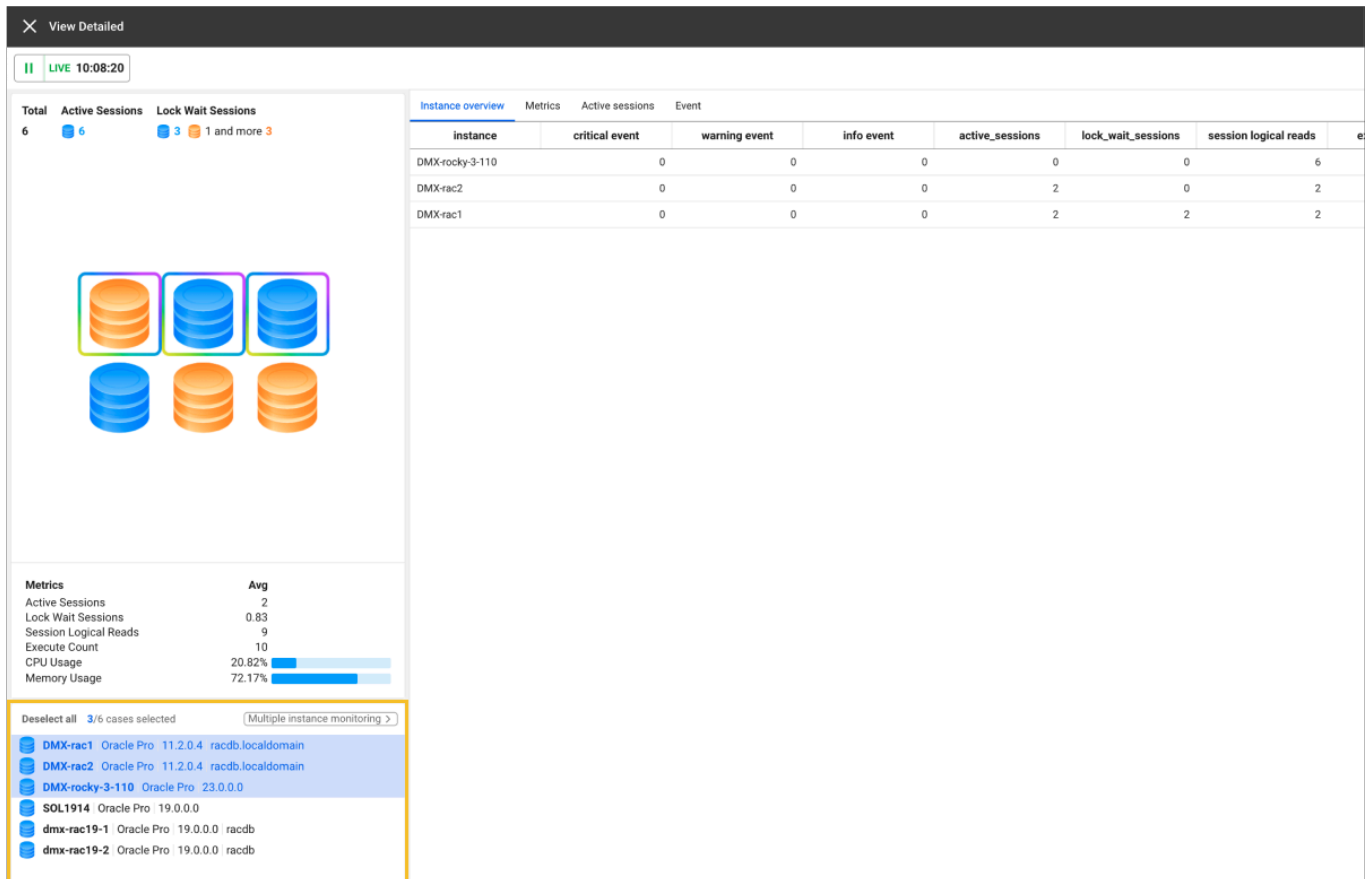
Instance Overview

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.

Topology

The Topology screen visualizes the replication and cluster structure of a database as a graph. You can grasp the connection relationships between the master and its replicas (slave, standby), the health status of each node, and whether there is replication lag or disconnection on a single screen.

! Supported platforms

The Topology screen supports the replication structure of the following platforms.

- MySQL (Replication)
- PostgreSQL (Replication)

The Topology menu is exposed for the two platforms above. Other platforms are planned for support.

Entering the screen

Home > select a project, then select **Topology** from the left menu.

Screen composition

The Topology screen consists of a graph area and auxiliary panels.

- **Summary card:** Displays summary information about the replication structure, such as the total number of nodes, the number of clusters, and the number of nodes with issues.
- **Legend:** Explains the node status colors and roles, and the meaning of edges (connection lines).
- **Issue panel:** Lists items that need attention, such as replication lag or disconnection. When you select an item, the screen moves to the corresponding node. The panel can be collapsed or expanded.
- **Minimap:** Shows the position of the area you are currently viewing within the entire graph.
- **Controls:** Provide zoom in/out and fit-to-screen functions.

How to read nodes and edges

- **Node:** Represents a single database instance. The node color indicates the health status, and the role indicator shown on the node indicates its role in the replication structure, such as master or replica.
- **Edge (connection line):** Represents the replication relationship from the master to a replica. The degree of replication lag is shown by the line color, and a disconnected relationship is shown as a dashed line. Synchronous and asynchronous replication are distinguished by arrows.
- **Cluster:** Displays the nodes belonging to the same replication group grouped in a box. The box shows the group name and platform together.

! The node colors in the replication topology are displayed based on the events (alerts) that occurred on each node.

2D view and 3D view

You can switch between the **2D** and **3D** views with the view toggle button at the top of the screen. In the 3D view, you can rotate the viewpoint by dragging and zoom in/out with the wheel. Both views support dark mode.

Checking detailed information

When you select a node, edge, or cluster, a detail panel (Drawer) appears on the right side of the screen. The selected target is highlighted, and unrelated targets are dimmed. When you hover the mouse pointer over each element, you can check the summary information as a tooltip.

AI diagnosis

When you run **AI diagnosis** in the node or cluster detail panel, the AI analyzes the replication structure based on the replication status at the selected point in time and provides the results. The analysis is displayed in the order of the progress step, analysis content, and diagnosis result (verdict).

ⓘ AI diagnosis is available per node or cluster on the platforms that support replication topology (currently MySQL and PostgreSQL). Other platforms are planned for support later.

Flex board widget

You can add the **Database Topology** widget to the integrated Flex board. In the widget, you can view the replication topology of multiple projects together on a single screen.

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

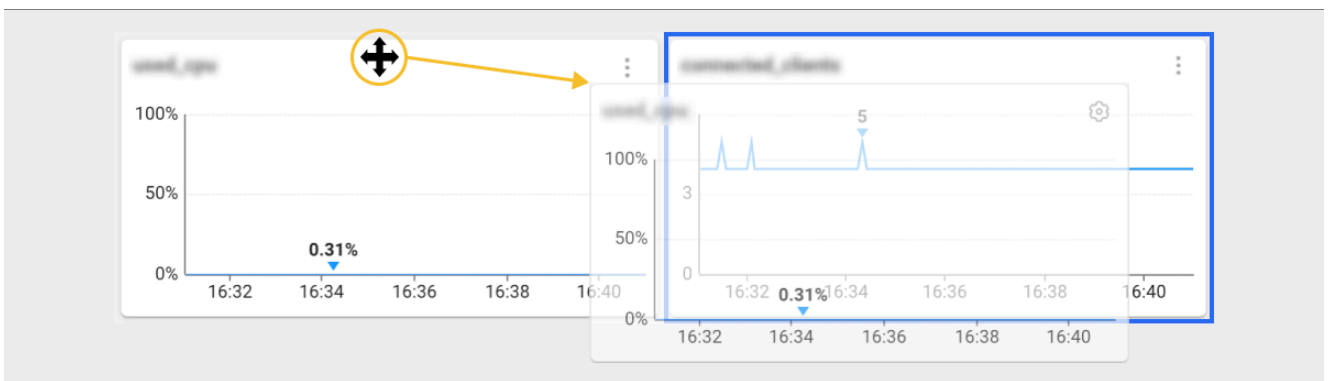
By default, the collected data for the last 10 minutes is retrieved in real time. If you select values in **Time** and **Instance**, the collected data is automatically applied to the **2** graph chart and **4 Active sessions** table.

- 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.

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** 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**

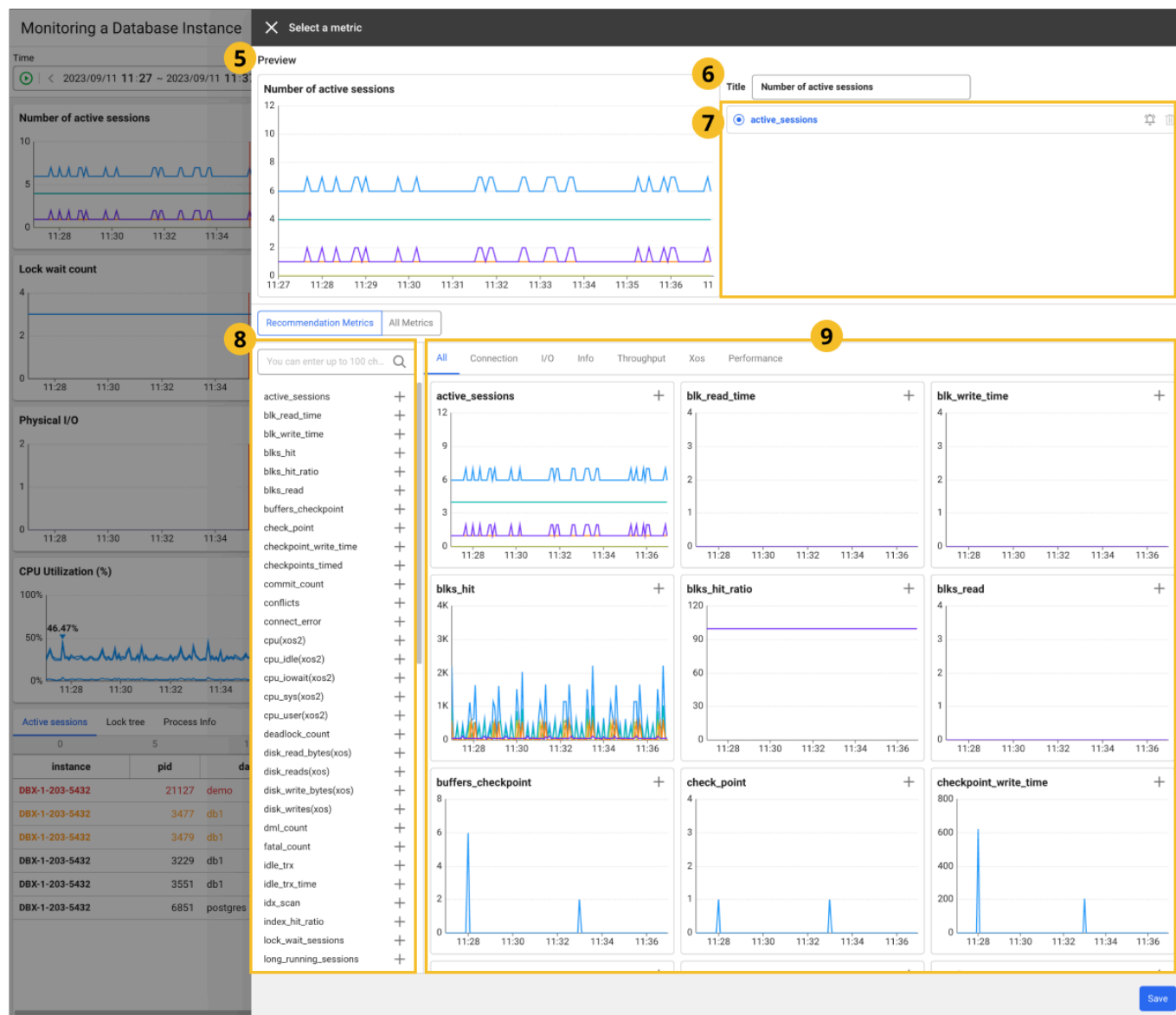
You can query the resource usage of the database server in real time.

✓ 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.
- The **Process Info** can be checked only when the XOS agent has been installed. For more information, see [the following](#).

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.

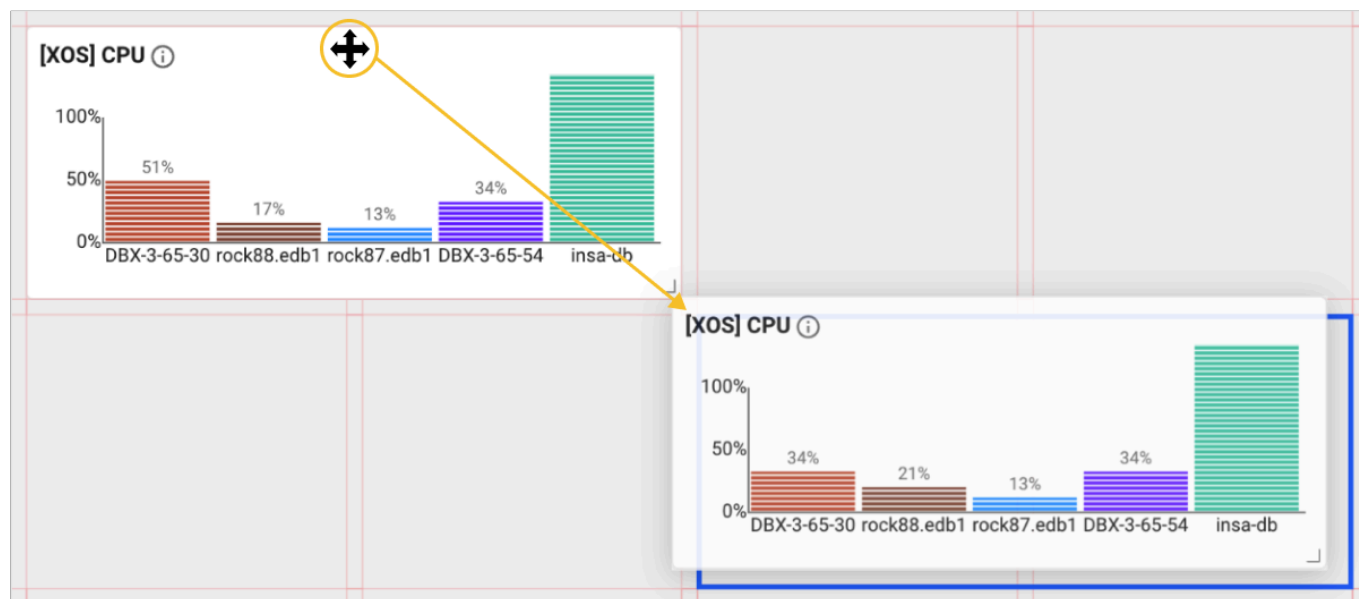
Resizing a widget



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

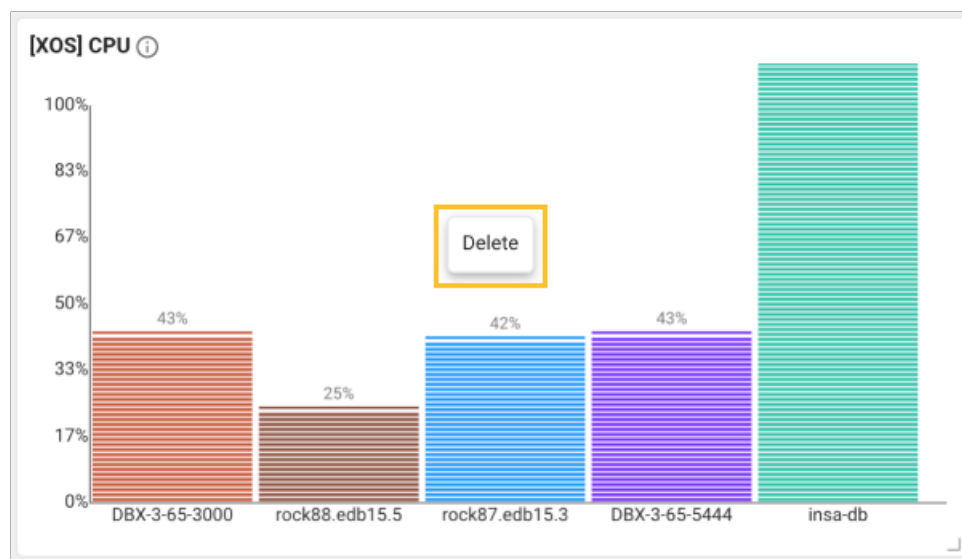
Grids with a uniform horizontal-to-vertical ratio appear and the size of the widget can be adjusted in each grid.

Moving a widget



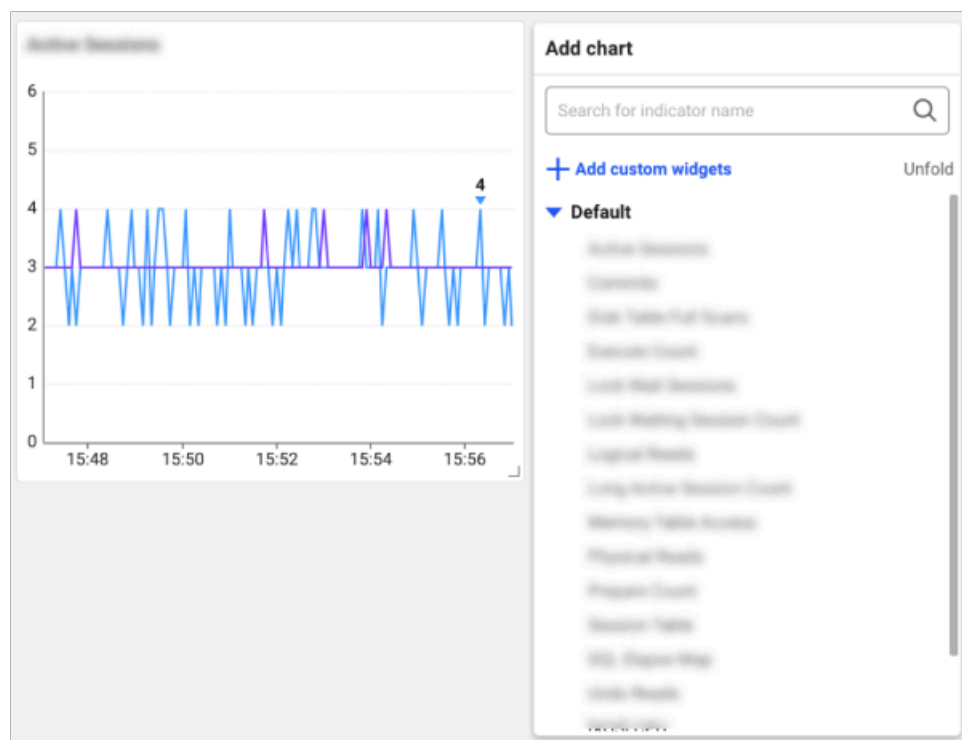
When you move the mouse cursor to the top of the widget, the cursor shape changes to **+**. At this time, you can move the widget by dragging with your left mouse button and move to the desired position.

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.

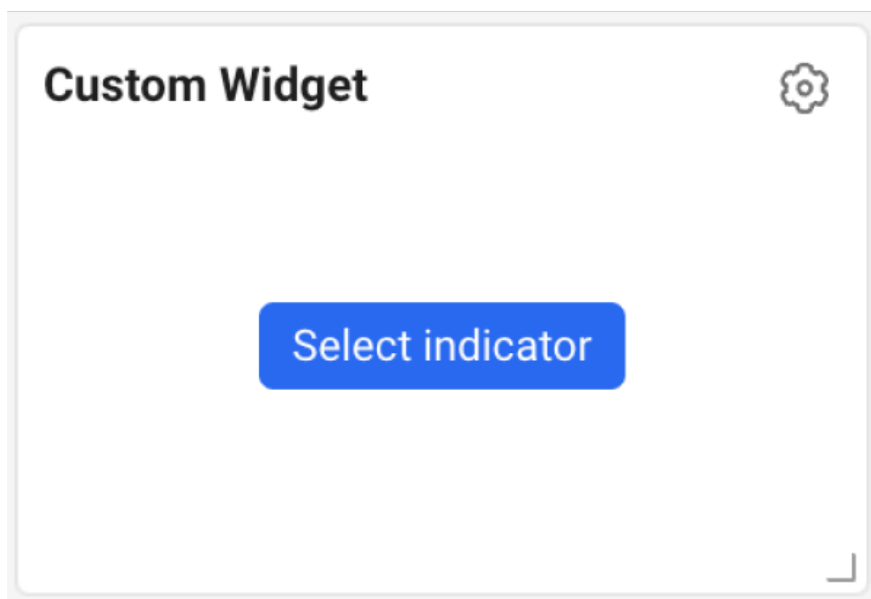
Custom widget

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.

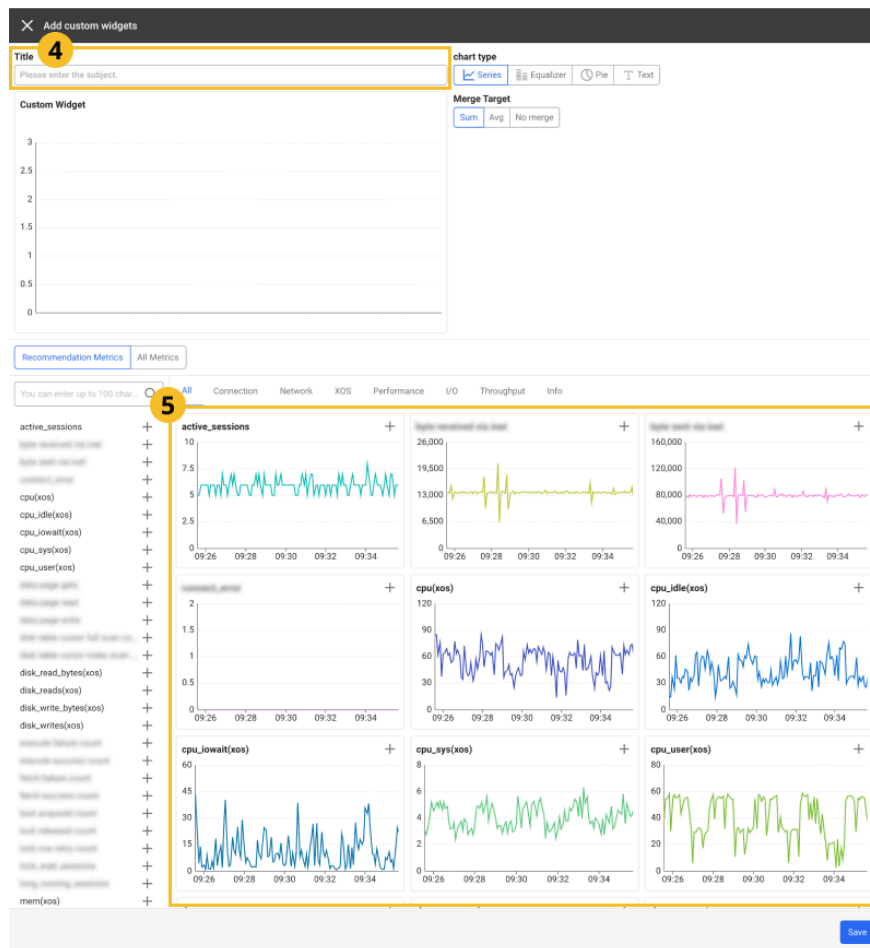
Add

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

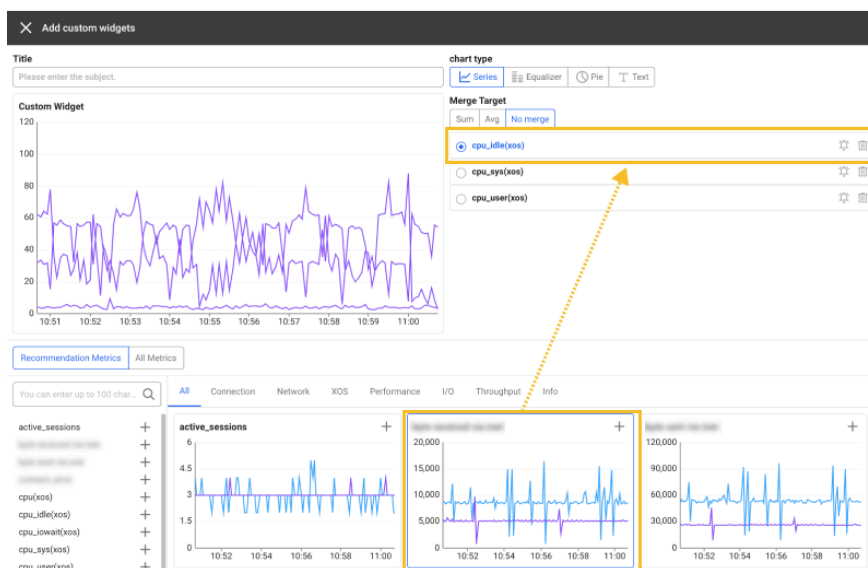
In **chart type** of the **Add Custom Widgets** window, you can select the following types.

-  **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:

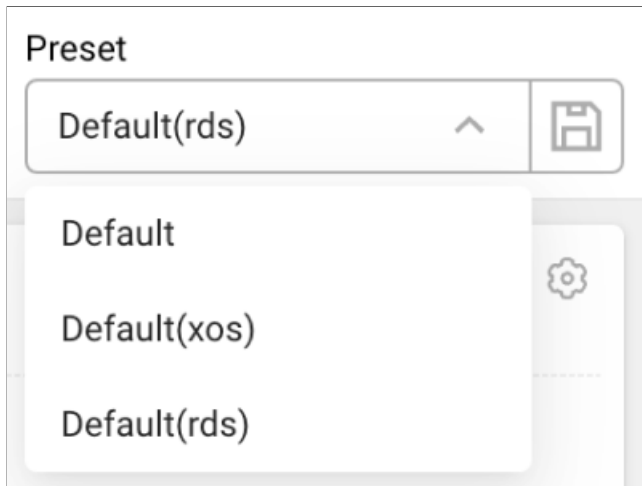
- : You can see the features and information for major widgets. (multilingual support planned)
-  or : You can enlarge or reduce the range of the vertical axis.
- : You can view the widget's data on a wider screen.
- : A detail window appears where you can search the data of the widgets separately by 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 following presets are provided by default, and the [Default](#) presets cannot be changed.

- [Default](#): It consists of major DB metrics.
- [Default\(xos\)](#): It consists of key DB metrics and XOS metrics (CPU and memory utilization, memory SWAP usage, disk I/O, etc.).
- [Default\(rds\)](#): It consists of key DB metrics and RDS CloudWatch metrics (CPU utilization, free memory space, free storage space, network, etc.).



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.

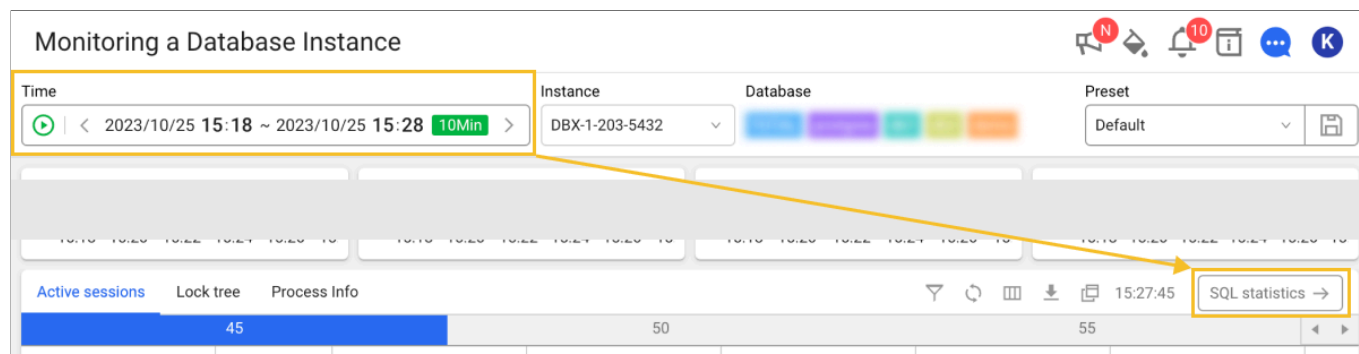
- : Using the column header entries in the table, the list can be filtered.
- : The list of tables is refreshed.
- : The column header entries in the table can be displayed or hidden.
- : The active session can be stopped.

- ❗ 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 **II** 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

pid

datname

cpuusage

client_addr

username

runtime ↓

state

wait_event

query

query_param

application_n

DBX-1-203-5432

10087

0.33%

192.168.1.204

whatap

0

active

SELECT /* WhaTap2D#4 */ datname,pid,extrac... 'idle'

Session detail (pid: 10087)

II

LIVE

08:33:47

Kill Session

Session history >

Session Info

pid

10087

query_hash

348448832

wait_event

datname

postgres

username

whatap

wait_event_type

query_start

2024-03-11 08:33:40

application_name

PostgreSQL JDBC Driver

runtime

0

backend_start

2024-03-11 05:31:50

state

active

backend_type

client backend

backend_xmin

2093748

xact_start

2024-03-11 08:33:40

client_addr

192.168.1.204

state_change

2024-03-11 08:33:40

client_hostname

192.168.1.204

client_port

52804

query_param

'idle',26096

SQL Text

Plan

Default Bind

Reset Bind

Formatting

SELECT
/* WhaTap2D#4 */
datname,
pid,
extract(
epoch
from
current_timestamp - query_start
) as runtime,
extract(

Process Info

cputime

14

cpuusage

0.33%

elapsed

10912

vsize

344528

rss

45160

state

active

ioread

23885

iowrite

0

- The session information is viewed in real time by default. If you select **II** 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 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](#).
- [Process Info](#): You can see the process information details of the selected session.

View SQL Details

To check the SQL query information of the executing session, select the **query** (or **sql_text**) column in the **Active sessions** section. The **SQL details** window appears. You can check the SQL query statements and plan information.

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

query

START_REPLICATION ...

SELECT /* ... */

SQL details View SQL Statistics →

Query (hash: 1098839306) Default Bind Reset Bind ☒ Formatting Plan demo whatap Text JSON 🔍

```

SELECT
/* long query */
to_char(count(*), 'DD.MM.YYYY') as
when,
'n' as departure,
'c' as arrival,
's' as status,
'o' as seat_no
FROM
bookings.ticket_flights a,
bookings.boarding_passes b
union all
SELECT
to_char(f.scheduled_departure, 'DD.MM.YYYY') AS
when,
f.departure_city || ' ' || f.departure_airport || ' ' AS departure,
f.arrival_city || ' ' || f.arrival_airport || ' ' AS arrival,
f.status,
bp.seat_no
FROM
ticket_flights tf
JOIN flights_v f ON tf.flight_id = f.flight_id
LEFT JOIN boarding_passes bp ON tf.flight_id = bp.flight_id
AND tf.ticket_no = bp.ticket_no
WHERE
1 = 1
and tf.ticket_no = '0005432661915';

```

Rows	Node Cost	Plans
4	0%	Append
1	13.9%	Aggregate
606149825272	86.1%	Strategy: Plain
579686	0%	Nested Loop Inner join
1045652	0%	Seq Scan on boarding_passes as b
1045652	0%	Materialize
1045652	0%	Seq Scan on ticket_flights as a
3	0%	Nested Loop on ((tf.ticket_no = bp.ticket_no) AND (tf.flight_id = bp.flight_id)) Left join
3	0%	Nested Loop Inner join
3	0%	Nested Loop Inner join
3	0%	Nested Loop Inner join
3	0%	Index Only Scan on ticket_flights as tf on (tf.ticket_no = '0005432661915':bpchar)
3	0%	Index Scan on flights as f on (f.flight_id = tf.flight_id)
1	0%	Index Scan on airports_data as ml on (ml.airport_code = f.departure_airport)
1	0%	Index Scan on airports_data as ml_1 on (ml_1.airport_code = f.arrival_airport)
3	0%	Materialize
3	0%	Bitmap Heap Scan on boarding_passes as bp
3	0%	Bitmap Index Scan on (bp.ticket_no = '0005432661915':bpchar)

- **Default Bind:** You can bind the part set as a variable in the query sentence to the default value.
- **Reset Bind:** You can release the bound default value and check the variable.
- **Formatting:** You can improve readability by applying indentation and formatting to a SQL query statement.
- **Plan:** To check plan information, enter **DB Name**, **User Name**, and **Password**, and then select 🔍.
 - **Text:** You can check plan information in text format.
 - **JSON:** By viewing the plan information in Json format, you can quickly find the points where bottlenecks occur.

ⓘ The function to view plan information in **JSON** format is supported only in DBX agent 1.6.12 or later.

- ✓ If you select a table or index while viewing the plan information in **JSON** format, the **Object detail** window appears where you can see the configurations for **columns** and **index** items in the table.



SQL details View SQL Statistics →

Query (hash: 1726024768) Default Bind Reset Bind Formatting

```

SELECT
/* WhaTap2D#4 */
datname,
pid,
extract(
epoch
from
current_timestamp - query_start
) as runtime,
extract(
epoch
from
harkend_start

```

Plan postgres whatap ⚙ Text JSON 🔍

Rows	Node Cost	Plans
99	46.8%	Plans - Hash Join Left join on (s.usersysid = u.oid) - Hash Join Left join on (s.datid = d.oid) - Function Scan as s - Filter : ((s.state IS NOT NULL) AND (s.state <> 'idle':text)) - Seq Scan pg_database as d - Hash - Seq Scan pg_auth as u
99	14.7%	
99	14.5%	
2	0%	
2	11.8%	
6	0%	
6	12.3%	

tap

SQL details View SQL Statistics →

Query (hash: 1726024768) Default Bind Reset Bind Formatting

```

SELECT
/* WhaTap2D#4 */
datname,
pid,
extract(
epoch
from
current_timestamp - query_start
) as runtime,
extract(
epoch
from
harkend_start

```

Plan postgres

Rows	Node Cost	Plans
99	46.8%	Plans - Hash Join Left join on (s.usersysid = u.oid) - Hash Join Left join on (s.datid = d.oid) - Function Scan as s - Filter : ((s.state IS NOT NULL) AND (s.state <> 'idle':text)) - Seq Scan pg_database as d - Hash - Seq Scan pg_auth as u
99	14.7%	
99	14.5%	
2	0%	
2	11.8%	
6	0%	
6	12.3%	

Object Detail

pg_database

type: table
idx_scan:
n_dead_tup:
last_autovacuum:
last_autoanalyze:

seq_scan:
n_live_tup:
last_vacuum:
last_analyze:

Index List

index_name	columns	type	is_unique	condition	comment
pg_database_datname_i	datname	BTREE	t		
pg_database_old_index	oid	BTREE	t		

This function is supported in DBX agent 1.6.15 or later. For more information about the database roles and related settings, see [the following](#).

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.
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

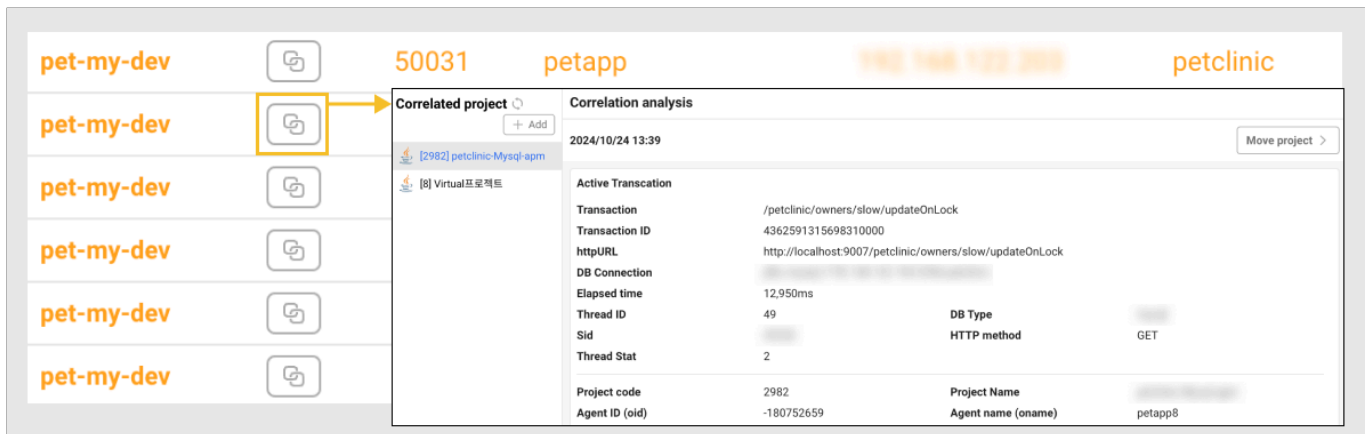
```
plan_db=(dbname)
```

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

```
planuid.bat user pwd
```

Checking the linked projects

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



Correlation analysis	
2024/10/24 13:39	
[2982] petclinic-Mysql-spm	
[8] Virtual프로젝트	
Active Transaction	
Transaction	/petclinic/owners/slow/updateOnLock
Transaction ID	4362591315698310000
httpURL	http://localhost:9007/petclinic/owners/slow/updateOnLock
DB Connection	
Elapsed time	12,950ms
Thread ID	49
Sid	
Thread Stat	2
Project code	2982
Agent ID (oid)	-180752659
Project Name	
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  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.

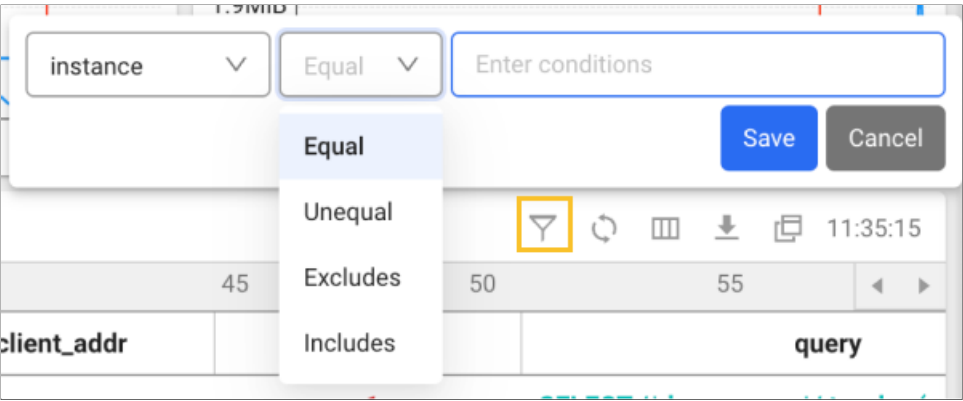
Changing the column order

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**.

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

- [Active session](#) [Lock tree](#) [Process information](#)

Item	Description
pid	Process ID
db	Database name
user	SQL execution account
application	Client application name
client_host	Host name of the connected client (reported by reverse DNS lookup of client_addr)
client_addr	Client IP address

Item	Description
<code>client_port</code>	Client port number (in case of using the UNIX socket -1)
<code>query_hash</code>	Hash value of query (internal WhaTap control value)
<code>query_start</code>	Time when the active query started (if the state is not active, it is the last query started time)
<code>state</code>	Current status of the backend process (active, idle, etc.)
<code>state_change</code>	Last time when the state was changed
<code>xact_start</code>	Time when the current transaction for the process started (<code>null</code> if no transaction is active)
<code>backend_start</code>	Time when the backend process started
<code>backend_xmin</code>	Minimum value of <code>xid</code> whose backend process is being processed
<code>backend_type</code>	Type of the backend process (e.g. <code>autovacuum worker</code> , <code>logical replication launcher</code> , <code>client backend</code> , <code>checkpointer</code>)
<code>backend_xid</code>	<code>xid</code> whose backend is being processed
<code>wait_event_type</code>	Type of the event whose backend process is in standby
<code>wait_event</code>	Waiting event name where the backend process is in standby

ⓘ WhaTap basically stores the client-related information.

Item	Description
<code>holder_relation</code>	Object (table or index) whose <code>holder</code> session has any lock.
<code>holder_type</code>	Type of the <code>holder</code> session (transaction ID, tuple, etc.)
<code>lock_mode</code>	<code>holder_mode</code> (exclusive lock, shared lock, etc.)

Item	Description
lock_request	waiter_mode (exclusive lock, shared lock, etc.)
state	Current status of each session (backend process)
waiter_relation	Object whose waiter session is waiting for lock acquisition.
waiter_type	Type of the waiter session (transaction ID, tuple, etc.)

The following items are the metrics collected if the XOS agent has been installed.

Item	Description
cputime	CPU usage time
cpuusage	CPU Utilization
elapse	Elapsed time of CPU usage
vsize	Virtual memory size (Kb)
rss	Resident Set Size (RSS) that is the number of physical pages associated with the process.
state	Process status
ioread	Actual time spent reading the block (milliseconds)
iowrite	Actual time spent writing the block (milliseconds)
pss	Process specific memory usage + Percentage of shared memory occupied by one process
uid	user id
cmd	Executing command
longcmd	Full path of cmd

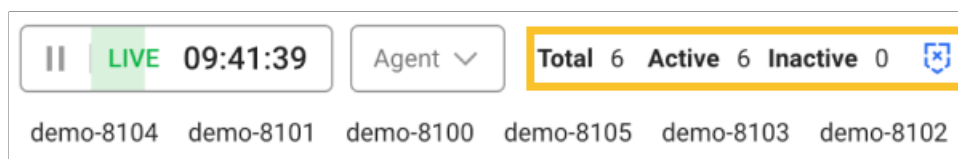
Monitoring Multiple Instances

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

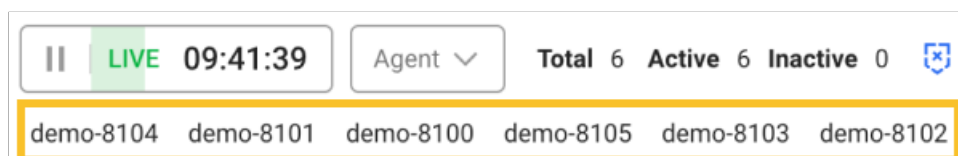
Checking the agent connection status



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

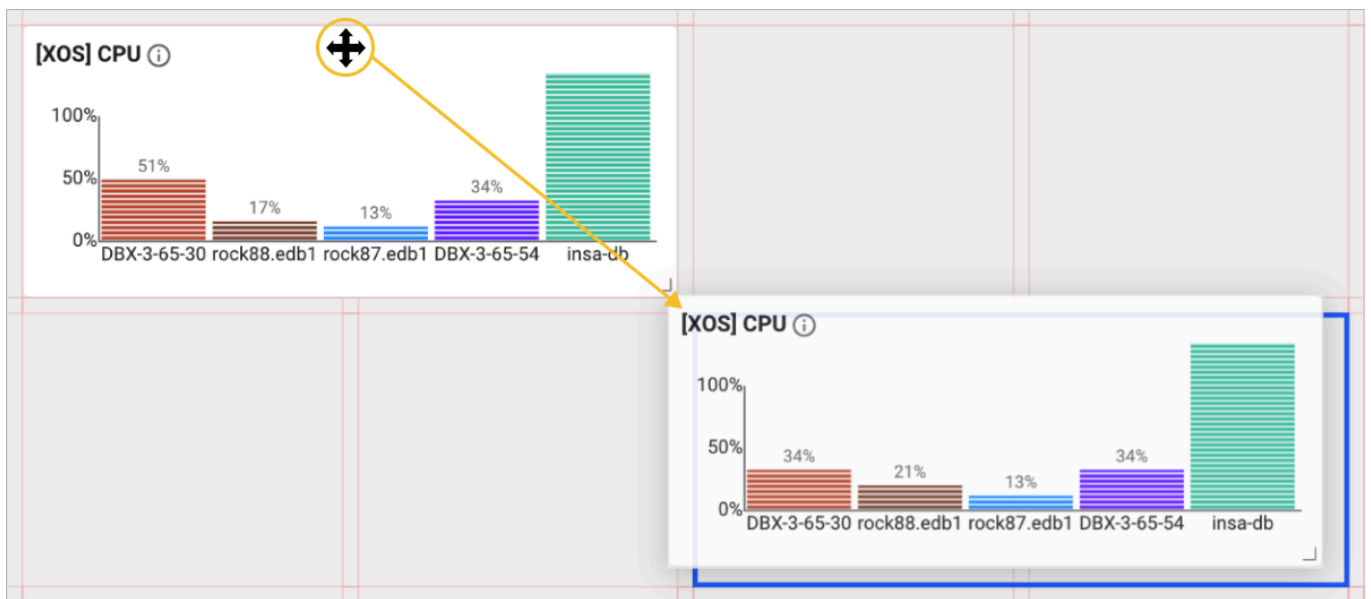
The widgets on the dashboard can be adjusted to the desired size and placed in the desired positions. You can also delete unnecessary widgets or add them again.

Resizing a widget



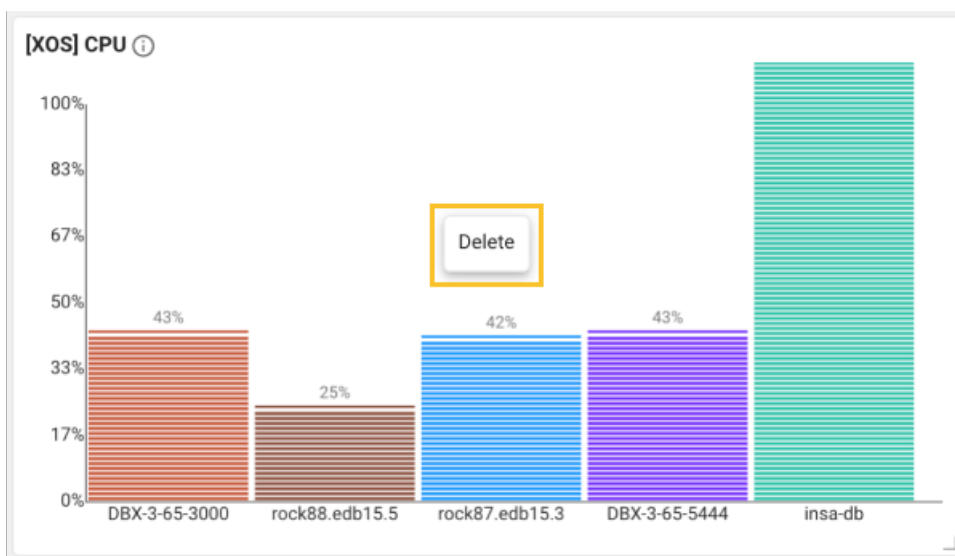
Click and hold the mouse on the  element at the lower right of the widget, and then drag it to the desired size. Grids with a uniform horizontal-to-vertical ratio appear and the size of the widget can be adjusted in each grid.

Moving a widget



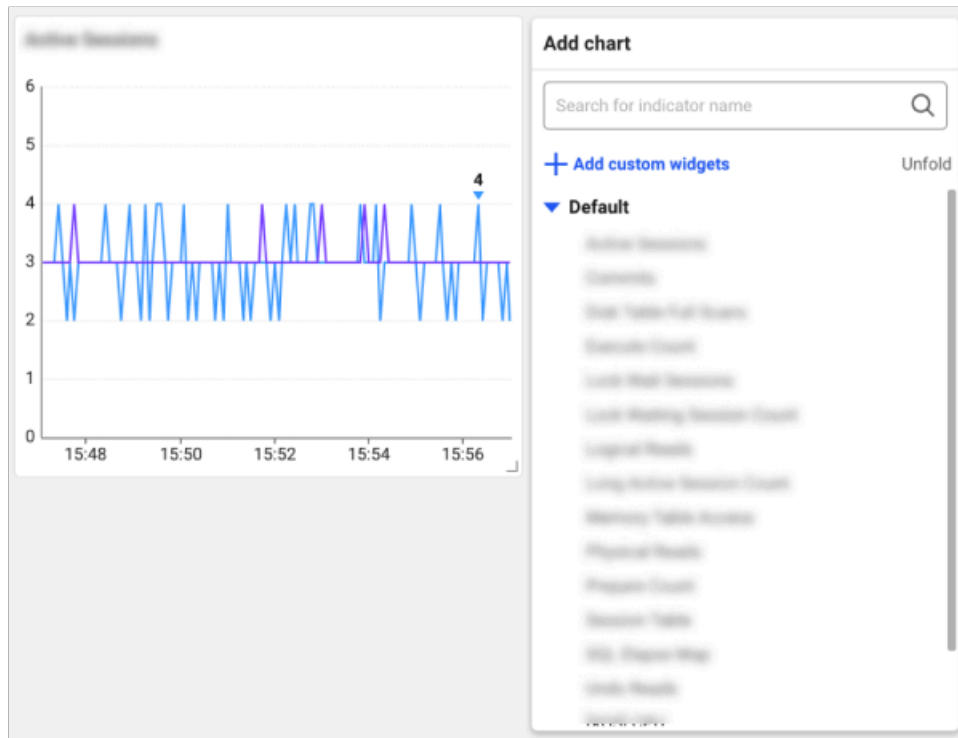
When you move the mouse cursor to the top of the widget, the cursor shape changes to **+**. At this time, you can move the widget by dragging with your left mouse button and move to the desired position.

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](#).

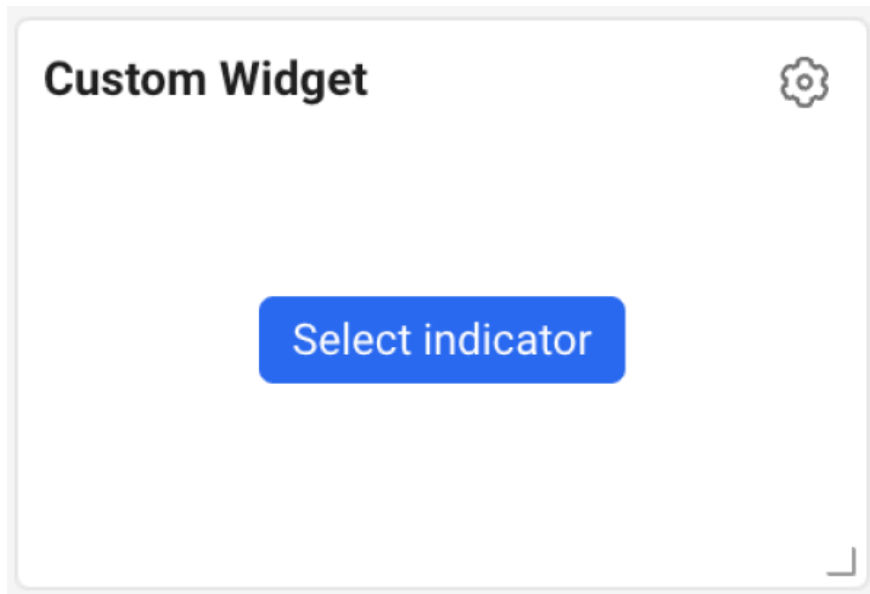
Custom widget

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

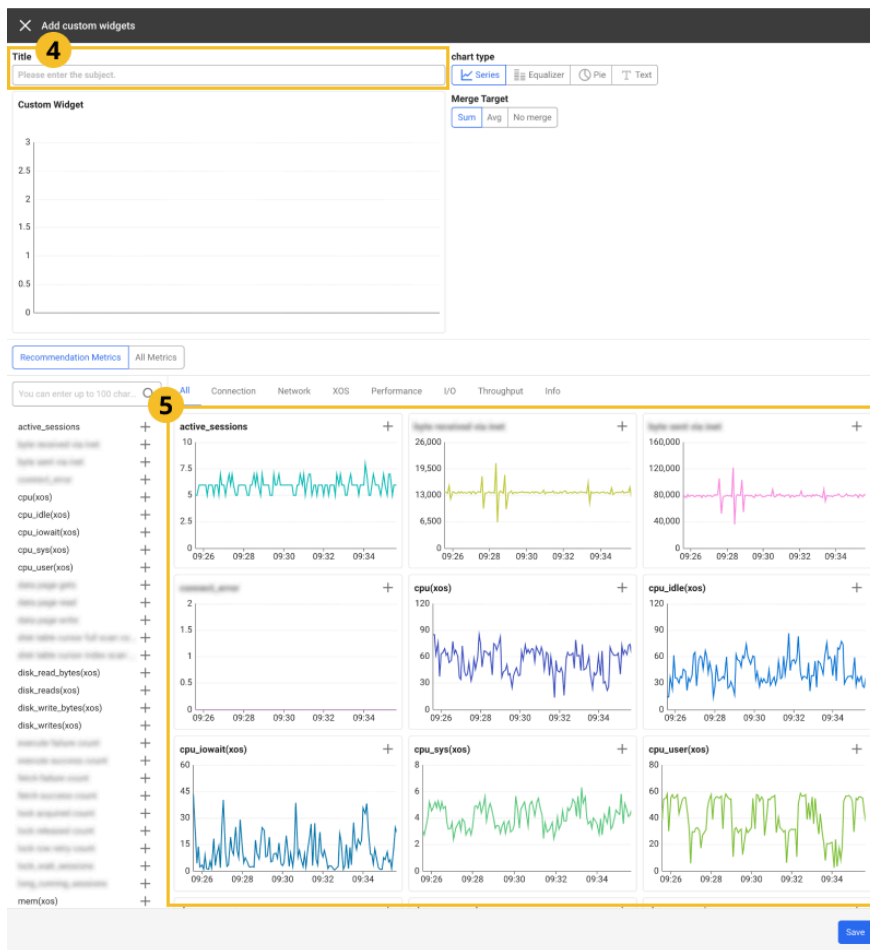
provided by default, use the custom widget feature.

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.

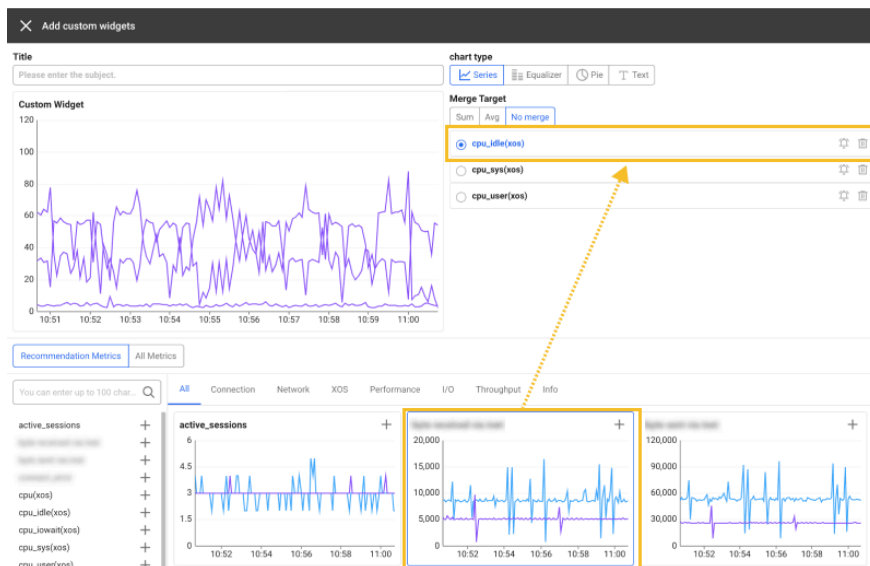
- **Sum**: It sums up the metric values collected from multiple agents and displays them as a single value. This option is useful for displaying the overall total by merging data from multiple agents.
- **Avg**: It averages the metric values collected from multiple agents and displays them as a single 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:** This preset consists of XOS metrics, AWS RDS CloudWatch metrics, and major DB metrics.
- **Default(xos):** This preset consists of XOS metrics and major DB metrics.
- **Default(rds):** This preset consists of AWS RDS CloudWatch metrics and major DB 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
```

AWS RDS CloudWatch widgets

- [RDS] CPUUtilization

You can see the CPU usages for each RDS DB instance in real time.

- [RDS] FreeableMemory

You can see the memory usages for each RDS DB instance in real time.

- [RDS] FreeStorageSpace

You can see the free storage space in the RDS database in real time.

! For information on how to configure the agent for inquiry of AWS CloudWatch metrics, see [the following](#).

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**

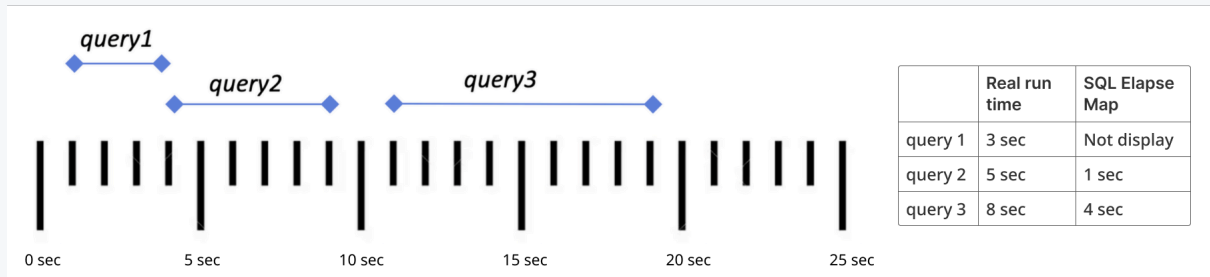
It provides information about the long-term latency of the sessions holding or waiting for locks in the database. By combining lock information and active session information, you can trace long-term latencies for queued sessions. It monitors the sessions that are holding locks or waiting for long time.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

- ❗ 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**.



- **DML Tuples**

Displays a real-time chart for the number of statements run for DML (Data Manipulation Language).

- **Slow Query**

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

- **Logical I/O**

The `blks_hit` metric is the total number of cases a disk block has been already found in the buffer cache and does not require read operation.

- **Physical I/O**

Number of cases data has been read from physical disk.

- **Uptime**

You can check how much time has passed since the agent (instance) started. The time appears in the `{dd}d {hh:mm:ss}` format.

- **Commits**

Number of commits occurred in the system or session.

- **Replication Delay**

Displays a real-time chart for the delayed times that are reflected in the replication DB in case of the replication structure.

- **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.

- **Vacuum Sessions**

The number of vacuum processes in progress can be checked. When the  button is selected on the upper right of the widget, a **Vacuum history** window appears. **Vacuum history** provides detailed information on the vacuum process executed.

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](#) 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 screenshot shows the 'Session detail' window for session ID 10087. The window is divided into several sections:

- Session Info:** Displays session details such as pid (10087), datname (postgres), query_start (2024-03-11 08:33:40), runtime (0), state (active), backend_xmin (2093748), client_addr (192.168.1.204), client_hostname (192.168.1.204), and query_param ('idle',26096).
- Wait Info:** Displays wait-related information including query_hash (348448832), wait_event, wait_event_type, and application_name (PostgreSQL JDBC Driver).
- Process Info:** Displays process-related information such as cputime (14), cpuusage (0.33%), elapse (10912), vsize (344528), rss (45160), state (active), ioread (23885), and iowrite (0).
- SQL Text:** Shows the SQL query being executed: `SELECT /* WhaTap2D#4 */ datname,pid,extrac... 'idle'`.
- Controls:** Includes a 'Kill Session' button and a 'Session history' link.

- The session information is viewed in real time by default. If you select **II** 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 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](#).
- **Process Info:** You can see the process information details of the selected session.

View SQL Details

To check the SQL query information of the executing session, select the **query** (or **sql_text**) column in the **Active sessions** section. The **SQL details** window appears. You can check the SQL query statements and plan information.

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

The screenshot shows a 'query' window with a 'START_REPLICATION' button and a 'SELECT /* ... */' query. Below this is the 'SQL details' window. The 'Query' tab shows a SQL query with a hash of 1098839306. The 'Plan' tab shows a table with columns 'Rows', 'Node Cost', and 'Plans'. The 'Plans' column shows a detailed execution plan for the query, including 'Append', 'Aggregate', 'Nested Loop Inner join', 'Seq Scan on boarding_passes as b', 'Materialize', 'Seq Scan on ticket_flights as a', 'Nested Loop on ((tf.ticket_no = bp.ticket_no) AND (tf.flight_id = bp.flight_id)) Left join', 'Nested Loop Inner join', 'Index Only Scan on ticket_flights as tf on (tf.ticket_no = '0005432661915':bpchar)', 'Index Scan on flights as f on (f.flight_id = tf.flight_id)', 'Index Scan on airports_data as ml on (ml.airport_code = f.departure_airport)', 'Index Scan on airports_data as ml_1 on (ml_1.airport_code = f.arrival_airport)', 'Materialize', 'Bitmap Heap Scan on boarding_passes as bp', and 'Bitmap Index Scan on (bp.ticket_no = '0005432661915':bpchar)'.

- **Default Bind:** You can bind the part set as a variable in the query sentence to the default value.
- **Reset Bind:** You can release the bound default value and check the variable.
- **Formatting:** You can improve readability by applying indentation and formatting to a SQL query statement.
- **Plan:** To check plan information, enter **DB Name**, **User Name**, and **Password**, and then select .
 - **Text:** You can check plan information in text format.
 - **JSON:** By viewing the plan information in Json format, you can quickly find the points where bottlenecks occur.

 The function to view plan information in **JSON** format is supported only in DBX agent 1.6.12 or later.

- ✓ If you select a table or index while viewing the plan information in **JSON** format, the **Object detail** window appears where you can see the configurations for **columns** and **index** items in the table.



The screenshot displays the SQL details interface. The top section shows the query plan for a query with hash 1726024768. The plan includes a Hash Join Left join on (s.usessysid = u.oid), followed by a Hash Join Left join on (s.datid = d.oid), a Function Scan as s, a Hash, a Seq Scan pg_database as d, another Hash, and a Seq Scan pg_auth as u. The bottom section shows the object detail for the pg_database table, including its type (table), statistics (n_live_tup, n_dead_tup, last_autovacuum, last_autoanalyze), and an index list with columns, types, and uniqueness.

This function is supported in DBX agent 1.6.15 or later. For more information about the database roles and related settings, see [the following](#).

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.
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](https://whatap.com)) file and then enter the DB name.

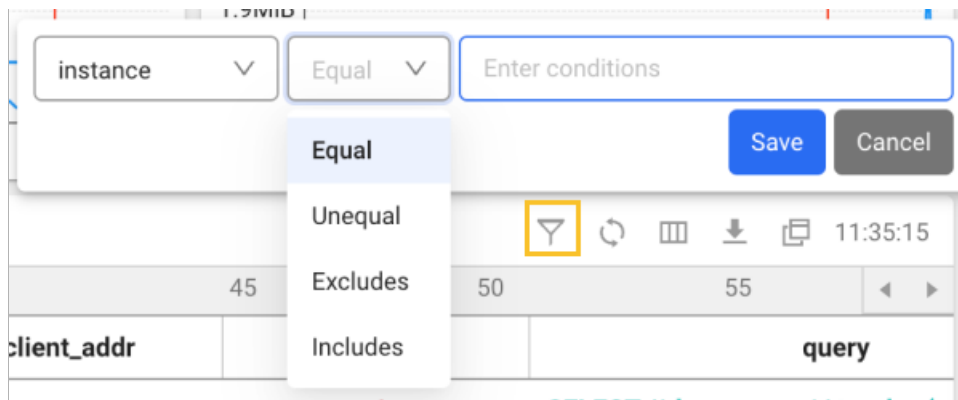
`whatap.conf`

```
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  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**.

Column information guide

- Active session Lock tree Process information

Item	Description
 pid	Process ID

Item	Description
db	Database name
user	SQL execution account
application	Client application name
client_host	Host name of the connected client (reported by reverse DNS lookup of client_addr)
client_addr	Client IP address
client_port	Client port number (in case of using the UNIX socket -1)
query_hash	Hash value of query (internal WhaTap control value)
query_start	Time when the active query started (if the state is not active, it is the last query started time)
state	Current status of the backend process (active, idle, etc.)
state_change	Last time when the state was changed
xact_start	Time when the current transaction for the process started (null if no transaction is active)
backend_start	Time when the backend process started
backend_xmin	Minimum value of xid whose backend process is being processed
backend_type	Type of the backend process (e.g. autovacuum worker , logical replication launcher , client backend , checkpointer)
backend_xid	xid whose backend is being processed
wait_event_type	Type of the event whose backend process is in standby
wait_event	Waiting event name where the backend process is in standby

! WhaTap basically stores the client-related information.

Item	Description
holder_relation	Object (table or index) whose holder session has any lock.
holder_type	Type of the holder session (transaction ID, tuple, etc.)
lock_mode	holder_mode (exclusive lock, shared lock, etc.)
lock_request	waiter_mode (exclusive lock, shared lock, etc.)
state	Current status of each session (backend process)
waiter_relation	Object whose waiter session is waiting for lock acquisition.
waiter_type	Type of the waiter session (transaction ID, tuple, etc.)

The following items are the metrics collected if the XOS agent has been installed.

Item	Description
cputime	CPU usage time
cpuusage	CPU Utilization
elapsed	Elapsed time of CPU usage
vsize	Virtual memory size (Kb)
rss	Resident Set Size (RSS) that is the number of physical pages associated with the process.
state	Process status
ioread	Actual time spent reading the block (milliseconds)
iowrite	Actual time spent writing the block (milliseconds)

Item	Description
pss	Process specific memory usage + Percentage of shared memory occupied by one process
uid	user id
cmd	Executing command
longcmd	Full path of cmd

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.

Slow Query

On the slow query screen, you can monitor and view slow SQL that exceeds the threshold time set in DB parameters. PostgreSQL records slow queries to a log file, but to check them, you need to connect to the DB server via terminal and read the text log directly. This screen collects slow query logs in real time and provides them as visual charts and lists.

- In the **Slow Query** chart, you can visually identify when slow queries occurred and their execution time.
- In **Slow Query List**, you can check detailed information for individual queries and view execution plans (Plan).
- Both real-time (Live) mode and historical time queries are supported.

ⓘ An additional agent is required to collect slow queries. In on-premises environments, install the **XOS agent**, and in cloud (AWS RDS, etc.) environments, install the **DBXC agent**.

✔ The slow query chart is similar to the SQL Elapse Map chart on the dashboard, but with the following differences:

Table | Differences between Slow Query chart and SQL Elapse Map chart

Category	Slow Query	SQL Elapse Map
Data Source	Slow query log recorded by DB	Active sessions collected by the agent at 5-second intervals
Execution Time Accuracy	Accurate — actual execution time recorded by DB	Up to 5 seconds of error possible due to 5-second interval collection
Collection Target	Only queries exceeding the threshold time (<code>log_min_duration_statement</code>)	All active sessions running at the time of collection

Prerequisites

To display data on this screen, [DB configuration](#), [DBX Agent configuration](#), and either **XOS Agent** or **DBXC Agent** configuration depending on your environment are required.

DB configuration

Set the slow query threshold time in [postgresql.conf](#). PostgreSQL restart or running `pg_reload_conf()` is required after changes.

postgresql.conf

```
log_min_duration_statement = 500
```

Parameter	Description
<code>log_min_duration_statement</code>	Records SQL that exceeds the set value (milliseconds) in the log. For example, if set to <code>500</code> , SQL taking 500ms or more is recorded.

DBX Agent configuration

Add the following setting to [whatap.conf](#).

whatap.conf

```
slow_query_log=ON
```

Option	Type	Default	Description
<code>slow_query_log</code>	String	OFF	Enable slow query log collection

OS agent configuration

An agent matching the DB operating environment is required to collect slow query logs.

Environment	Required Agent	Key Configuration	Setup Guide
On-premises (Linux, Windows)	XOS agent	None (if auto-detection fails, specify log file path directly in <code>slow_query=</code>)	XOS Agent Configuration
Cloud (AWS RDS, NCP, etc.)	DBXC agent	<code>slow_query: true</code> (per-instance setting in <code>config.yaml</code>)	DBXC Agent Configuration

Slow Query

Basic options

In the top filter area of the slow query screen, you can select the time period and target DB instance to query.

- **Time:** Select real-time (Live) mode or a past time. Live mode automatically refreshes every 10 seconds.
- **Instance:** Select the target DB instance to query. You can select all or individual instances.

Slow Query chart

Displays the time when slow queries occurred as a scatter chart. The Y-axis is execution time (seconds) and the X-axis is time.

- Each dot represents one slow query; dots higher up indicate queries that took longer.
- **Mouse drag** on the chart area displays only the queries from that interval in the list below.
- The **▲ / ▼** buttons in the upper right adjust the Y-axis range.

Slow Query List

Displays the list of slow queries in the selected interval from the chart.

Column	Description
Instance	Name of the instance where the query was executed
DB	Database name
User	Executing user name
Log Time	Time when the slow query log was recorded
Type	Log type (e.g., statement)
Coltype	Collection type
Duration (s)	Query execution time (seconds)
SQL	Executed SQL statement

SQL details

Click the query in the SQL column to open the details window.

Data interpretation guide

Key checkpoints

- **Queries with long Duration:** Queries with long individual execution time → SQL tuning, index check
- **When the same SQL appears repeatedly:** A specific query is consistently slow → Fundamental tuning required

- **Concentrated in a specific time period:** Check for external factors such as DB load, lock contention, batch jobs during that time

Practical usage scenarios

Scenario 1. Real-time monitoring of slow queries

1. Observe the Slow Query chart in **Live** mode
2. When new dots appear, drag and select that area
3. Check queries in the Slow Query List → Click SQL to view details and execution plan

Scenario 2. Analyzing slow queries in a specific time period

1. Select the problematic time period in the time filter
2. Drag the interval where queries are concentrated in the chart
3. Check queries with the longest Duration first to analyze the cause

Scenario 3. Adjusting the slow query threshold

1. If too many slow queries are being collected, raise the `log_min_duration_statement` value to loosen the threshold
2. If no slow queries are being collected, lower the `log_min_duration_statement` value or check the `slow_query_log=ON` setting

Analyzing with AI

Click the **WhaTap AI** button on the screen to open the WhaTap AI chat window, where the queried screen data is summarized and the analysis result is shown as a conversation. After reading the result, you can continue with follow-up questions in the same conversation.

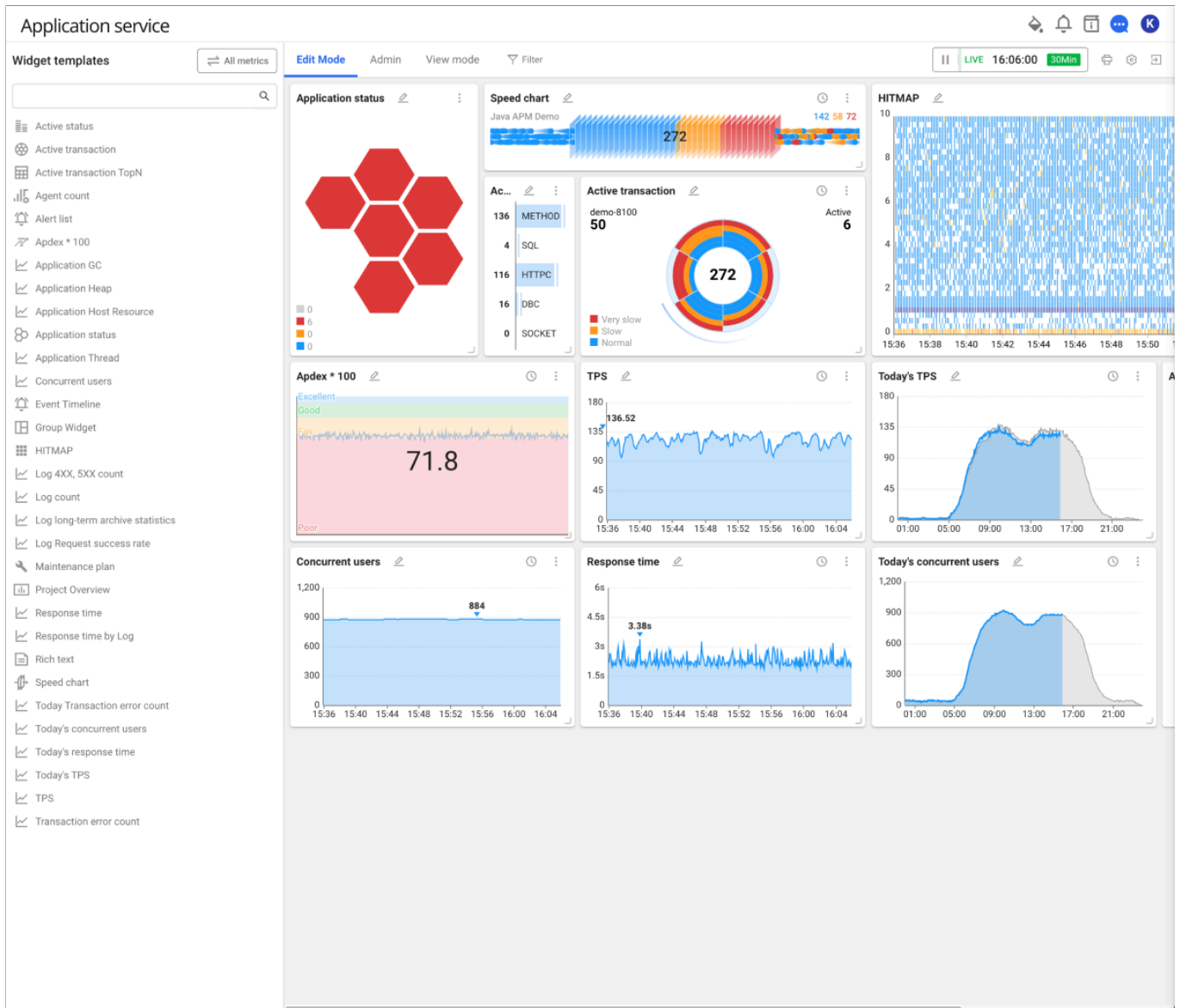
- The data used for the analysis is the top 20 items with the longest elapsed time among the items selected on the scatter plot.

- The analysis target is the data queried on the screen. If you change the query conditions and analyze again, the analysis is based on the changed data.
- The conversation history is recorded in the `AI Analysis - {screen name}` format.

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.



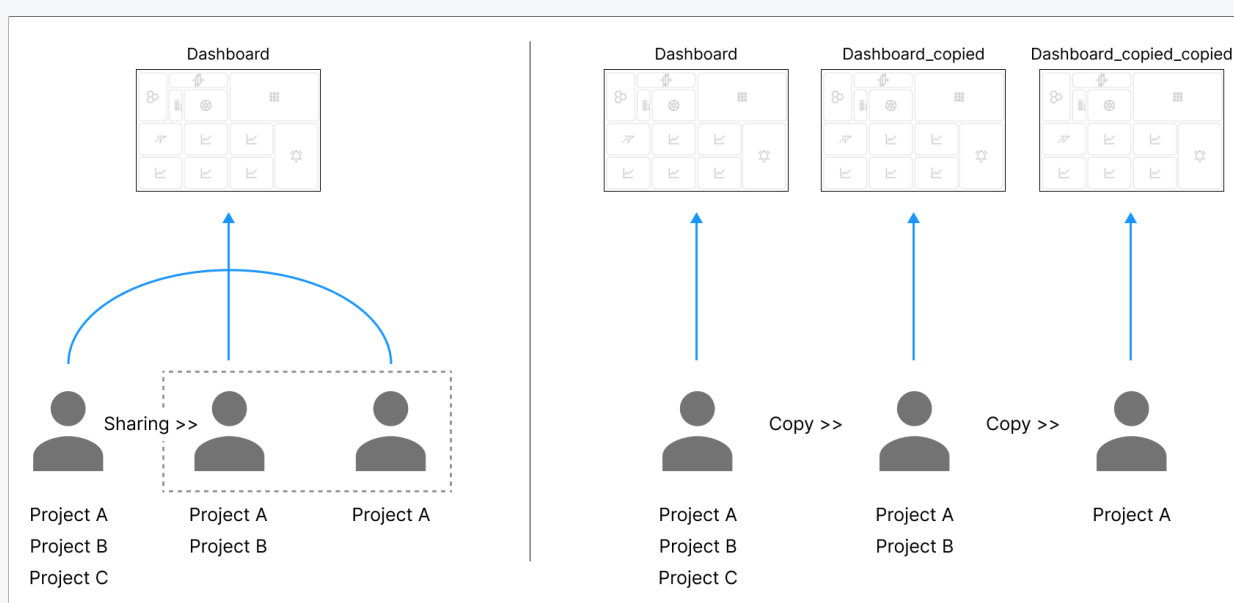
- If a user who received a shared dashboard deletes it, it is deleted only from that user's **Dashboard list**. The original dashboard remains.

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**

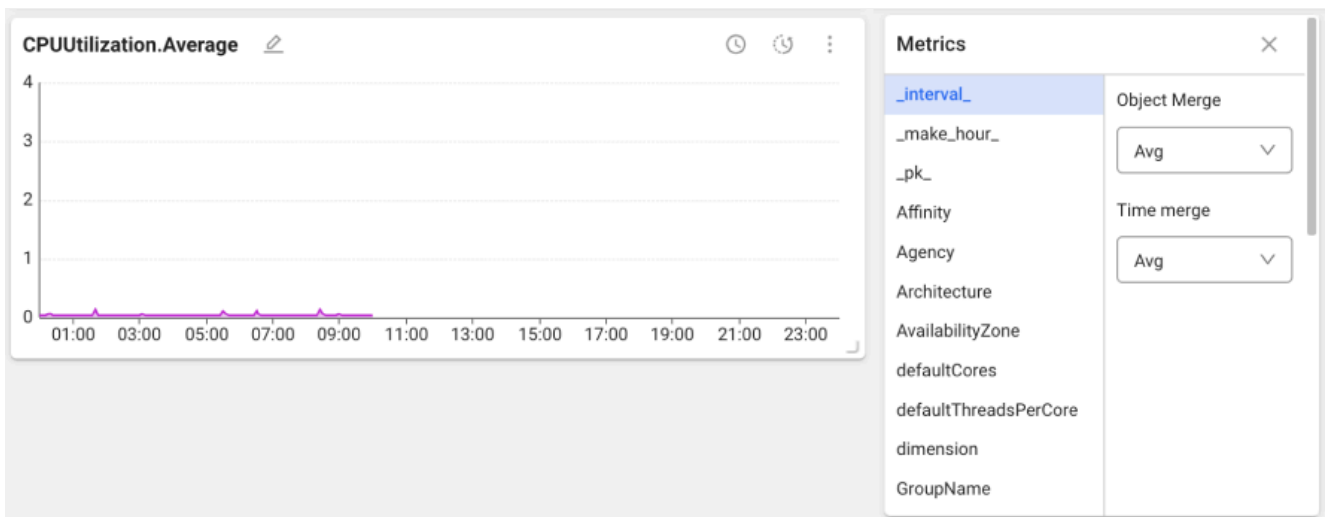
It is used to group the chart data based on the tag (e.g. agent name) selected on the left. At this time, if the tags match among the data with different fields (e.g. CPU), the corresponding data is merged.

- **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**.

Comparison Between Count Trends

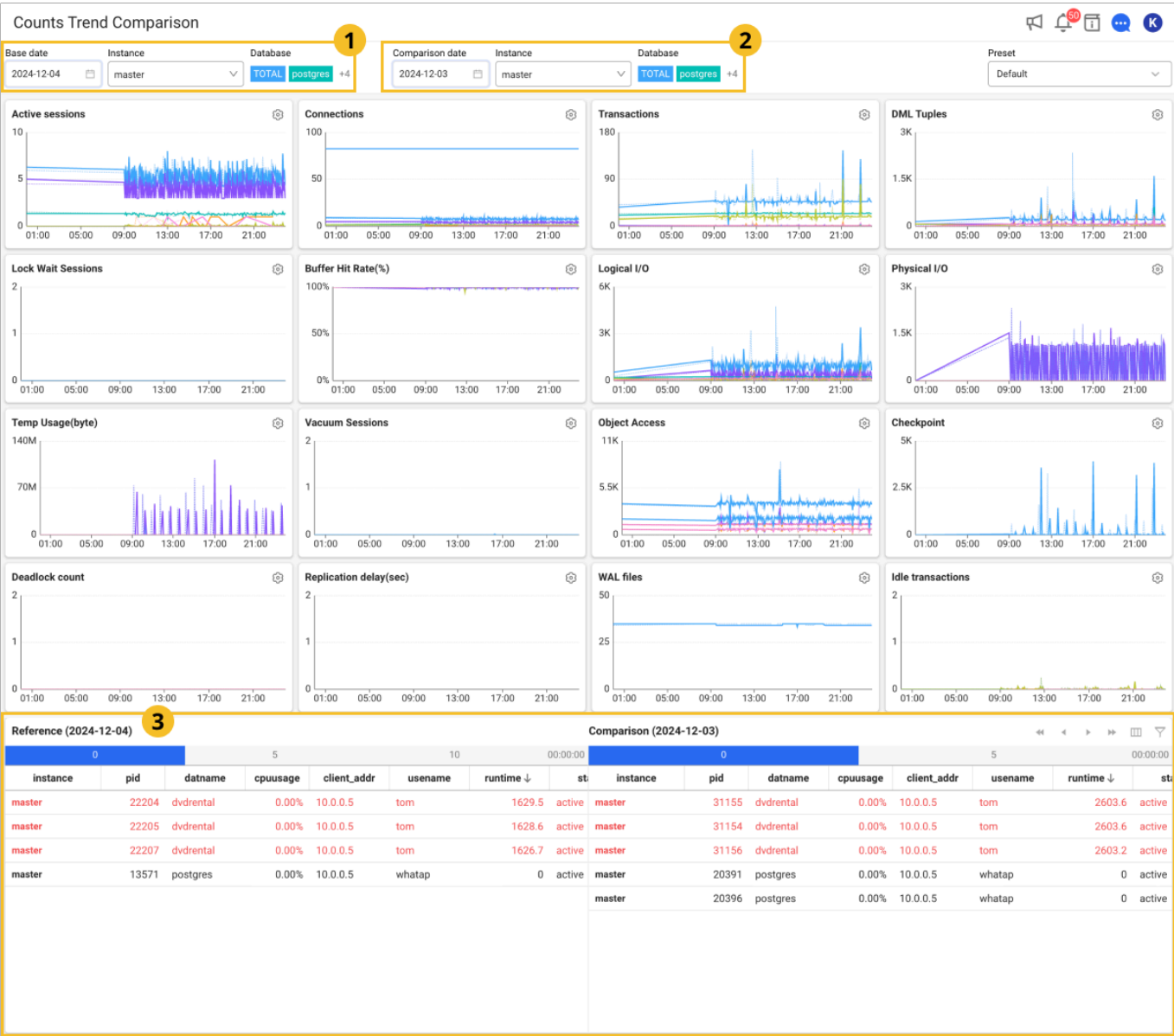
Home > Select Project > **Analysis** > **Counts Trend Comparison**

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



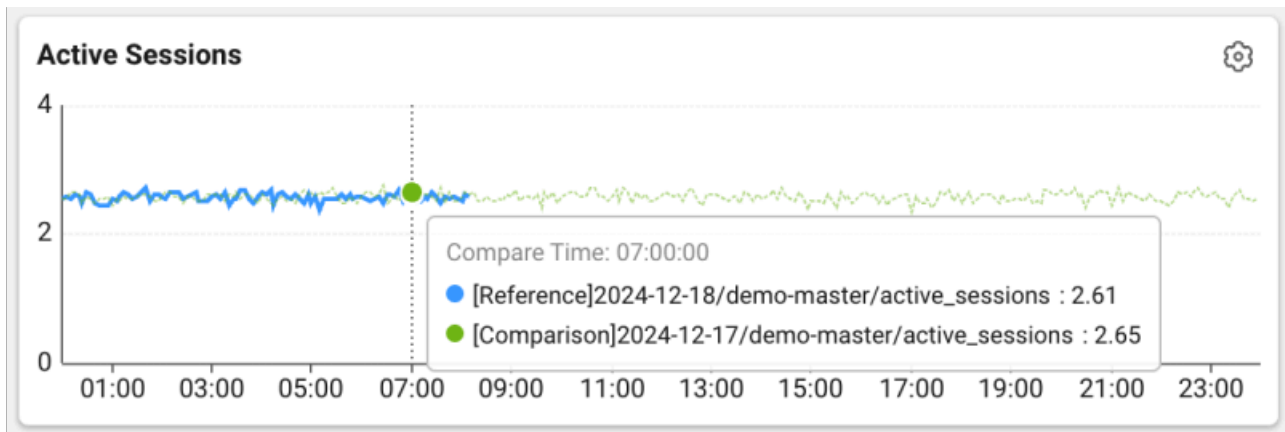
- If you select 1 Base date, 2 Comparison date, and Instance, the data is automatically applied to all widgets based on the selected items.
- To see the meaning of the metric displayed in the widget, select ⓘ next to the name or see [the following](#).

❗ 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.



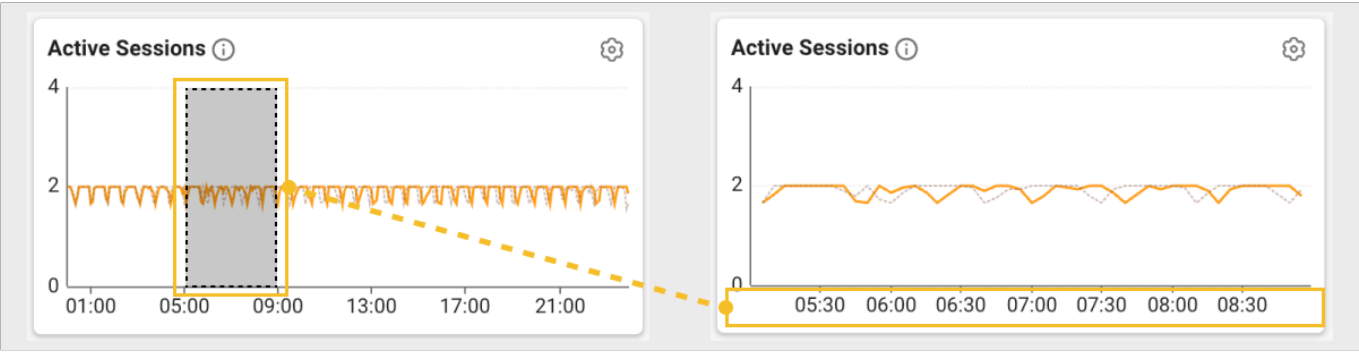
- The color of **Database** at the top of the screen is displayed in the same color on the graph chart.

Comparing the active session data

Reference (2024-12-04)								Comparison (2024-12-03)							
0		5		10		00:00:00		0		5		00:00:00			
instance	pid	datname	cpuusage	client_addr	username	runtime ↓	st	instance	pid	datname	cpuusage	client_addr	username	runtime ↓	st
master	22204	dvdrental	0.00%	10.0.0.5	tom	1629.5	active	master	31155	dvdrental	0.00%	10.0.0.5	tom	2603.6	active
master	22205	dvdrental	0.00%	10.0.0.5	tom	1628.6	active	master	31154	dvdrental	0.00%	10.0.0.5	tom	2603.6	active
master	22207	dvdrental	0.00%	10.0.0.5	tom	1626.7	active	master	31156	dvdrental	0.00%	10.0.0.5	tom	2603.2	active
master	13571	postgres	0.00%	10.0.0.5	whatap	0	active	master	20391	postgres	0.00%	10.0.0.5	whatap	0	active
								master	20396	postgres	0.00%	10.0.0.5	whatap	0	active

- 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.

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

- [Active session](#) Lock tree Process information

Item	Description
pid	Process ID
db	Database name
user	SQL execution account

Item	Description
<code>application</code>	Client application name
<code>client_host</code>	Host name of the connected client (reported by reverse DNS lookup of <code>client_addr</code>)
<code>client_addr</code>	Client IP address
<code>client_port</code>	Client port number (in case of using the UNIX socket -1)
<code>query_hash</code>	Hash value of query (internal WhaTap control value)
<code>query_start</code>	Time when the active query started (if the state is not active, it is the last query started time)
<code>state</code>	Current status of the backend process (active, idle, etc.)
<code>state_change</code>	Last time when the state was changed
<code>xact_start</code>	Time when the current transaction for the process started (<code>null</code> if no transaction is active)
<code>backend_start</code>	Time when the backend process started
<code>backend_xmin</code>	Minimum value of <code>xid</code> whose backend process is being processed
<code>backend_type</code>	Type of the backend process (e.g. <code>autovacuum worker</code> , <code>logical replication launcher</code> , <code>client backend</code> , <code>checkpointer</code>)
<code>backend_xid</code>	<code>xid</code> whose backend is being processed
<code>wait_event_type</code>	<code>Type</code> of the event whose backend process is in standby
<code>wait_event</code>	Waiting event name where the backend process is in standby

❗ WhaTap basically stores the client-related information.

Item	Description
holder_relation	Object (table or index) whose holder session has any lock.
holder_type	Type of the holder session (transaction ID, tuple, etc.)
lock_mode	holder_mode (exclusive lock, shared lock, etc.)
lock_request	waiter_mode (exclusive lock, shared lock, etc.)
state	Current status of each session (backend process)
waiter_relation	Object whose waiter session is waiting for lock acquisition.
waiter_type	Type of the waiter session (transaction ID, tuple, etc.)

The following items are the metrics collected if the XOS agent has been installed.

Item	Description
cputime	CPU usage time
cpuusage	CPU Utilization
elapse	Elapsed time of CPU usage
vsize	Virtual memory size (Kb)
rss	Resident Set Size (RSS) that is the number of physical pages associated with the process.
state	Process status
ioread	Actual time spent reading the block (milliseconds)
iowrite	Actual time spent writing the block (milliseconds)
pss	Process specific memory usage + Percentage of shared memory occupied by one process

Item	Description
uid	user id
cmd	Executing command
longcmd	Full path of cmd

Lock Tree

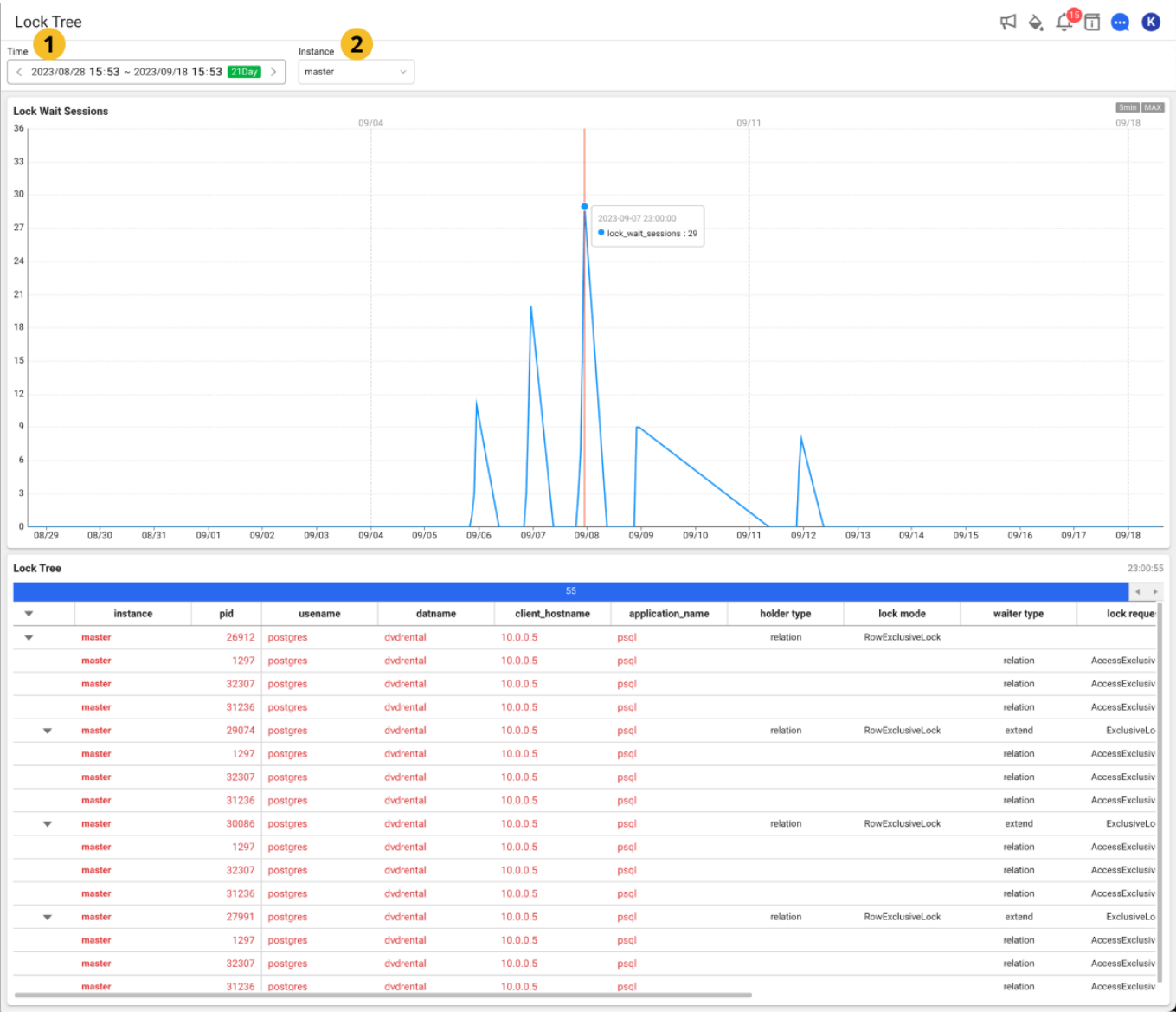
Home > Select Project > **Analysis** > **Lock Tree**

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



button in seconds, you can view the lock information in 5-second increments.

! Through the tags on the upper right of the [Lock Wait Sessions](#) chart, you can see the units to view.

- As you continuously drag (drill down) a specific time zone on the graph chart, you can see the charts and lock trees for detailed time zones.
- You can see the data in 5-second increments by dragging within 3 hours on the chart. However, the data in 5-second increments can only be viewed within the most recent month. The data earlier than one month can only be viewed as 5-minute summary data. For example, you can drag (drill down) lock tree data from 40 days ago for the 5-minute summary data, but you cannot see 5-second data.

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.

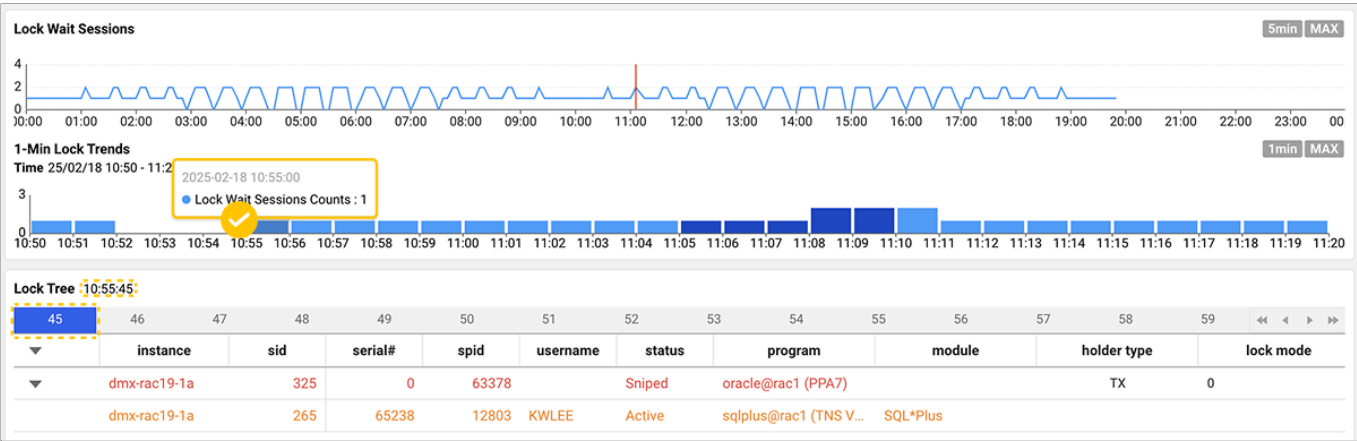
! [Total Locks](#) indicates the total number of locks generated in the time zone.

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.



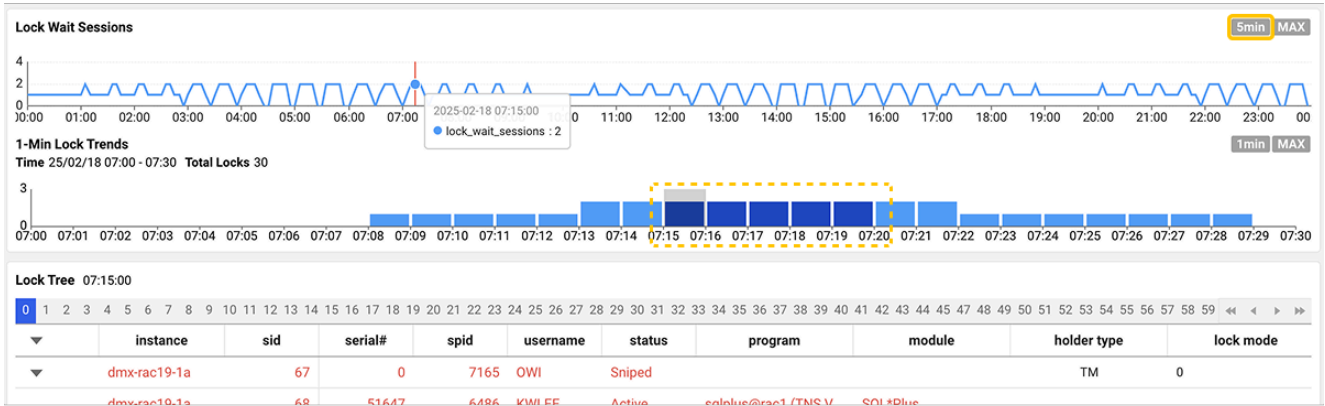
If you click the bar of a different time zone in the 1-Min Lock Trends chart with the Lock Wait Sessions unchanged, the time range is synced to the Lock Tree table by using the seconds with the highest number of locks for the minute. This allows you to see the information reflected in the Lock Tree table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



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

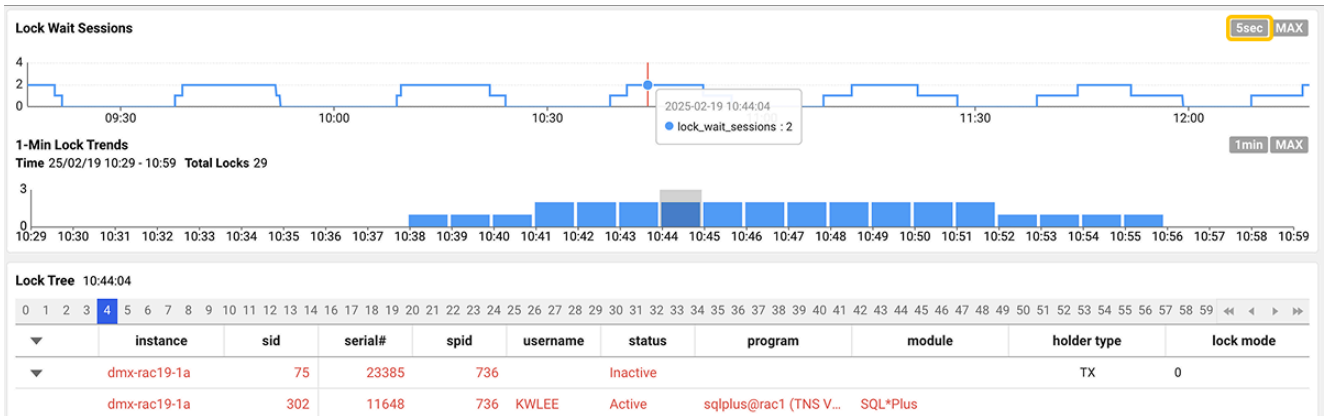
The bar color of the 5-minute zone from the time of inquiry appears more prominent.

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.



Column information guide

- [Active session](#) [Lock tree](#) [Process information](#)

Item	Description
pid	Process ID
db	Database name
user	SQL execution account

Item	Description
<code>application</code>	Client application name
<code>client_host</code>	Host name of the connected client (reported by reverse DNS lookup of <code>client_addr</code>)
<code>client_addr</code>	Client IP address
<code>client_port</code>	Client port number (in case of using the UNIX socket -1)
<code>query_hash</code>	Hash value of query (internal WhaTap control value)
<code>query_start</code>	Time when the active query started (if the state is not active, it is the last query started time)
<code>state</code>	Current status of the backend process (active, idle, etc.)
<code>state_change</code>	Last time when the state was changed
<code>xact_start</code>	Time when the current transaction for the process started (<code>null</code> if no transaction is active)
<code>backend_start</code>	Time when the backend process started
<code>backend_xmin</code>	Minimum value of <code>xid</code> whose backend process is being processed
<code>backend_type</code>	Type of the backend process (e.g. <code>autovacuum worker</code> , <code>logical replication launcher</code> , <code>client backend</code> , <code>checkpointer</code>)
<code>backend_xid</code>	<code>xid</code> whose backend is being processed
<code>wait_event_type</code>	<code>Type</code> of the event whose backend process is in standby
<code>wait_event</code>	Waiting event name where the backend process is in standby

❗ WhaTap basically stores the client-related information.

Item	Description
holder_relation	Object (table or index) whose holder session has any lock.
holder_type	Type of the holder session (transaction ID, tuple, etc.)
lock_mode	holder_mode (exclusive lock, shared lock, etc.)
lock_request	waiter_mode (exclusive lock, shared lock, etc.)
state	Current status of each session (backend process)
waiter_relation	Object whose waiter session is waiting for lock acquisition.
waiter_type	Type of the waiter session (transaction ID, tuple, etc.)

The following items are the metrics collected if the XOS agent has been installed.

Item	Description
cputime	CPU usage time
cpuusage	CPU Utilization
elapse	Elapsed time of CPU usage
vsize	Virtual memory size (Kb)
rss	Resident Set Size (RSS) that is the number of physical pages associated with the process.
state	Process status
ioread	Actual time spent reading the block (milliseconds)
iowrite	Actual time spent writing the block (milliseconds)
pss	Process specific memory usage + Percentage of shared memory occupied by one process

Item	Description
uid	user id
cmd	Executing command
longcmd	Full path of cmd

Session History

This screen allows you to view session information collected during the query period in chronological order. You can analyze which sessions executed which SQL at specific times, and what the status and resource usage of each session was.

- Data is provided based on active session snapshots collected at 5-second intervals.
- The **Session Trend** chart allows you to visually track changes in session count over time.
- Use the field panel and filters to query only sessions that meet specific conditions.

Querying Session History

Set query conditions in the top filter, filter sessions using the field panel, then check session changes and status over time through the session trend chart and list.

Basic Options

You can set time, target instance, and filters in the top filter area.

- **Time:** Select the query time range; set to 10-minute intervals by default, and you can move with left/right arrows
- **Instance:** Select the DB instance to query
- **Filter:** Add conditions per field to filter data; AND logic is applied for multiple conditions
- **View Mode:** Switch between table view and tree view
- **Download:** Export in CSV format

Field Panel

You can easily add filter conditions from the field panel on the left side of the screen.

1. Click a field name to display a condition input popup.
2. Set the operator and value, then click the **Apply** button to add the condition to the top filter.

3. You can search for field names using the search bar at the top. When a filter condition is applied, only sessions that meet the conditions are displayed in the chart and table.
 - For example, if you set a specific keyword as a contain condition in the `query` field, only sessions executing SQL that includes that keyword are queried. When multiple conditions are applied, **AND** logic is used.

Session Trend Chart

Displays the number of active sessions and lock wait sessions during the query period as a bar chart. The X-axis is time and the Y-axis is the number of sessions. You can query up to **1 hour** at a time.

- Among all active sessions, **lock wait sessions are displayed in red.**
- You can visually identify the total number of active sessions and the proportion of lock wait sessions at specific time periods.
- **Drag** the chart area to drill down into that interval for detailed analysis.
- **Click** a specific time on the chart to display only the active sessions for that time period in the table below.

Session List Table

Displays collected session information in chronological order. You can navigate pages using the pagination at the bottom, and you can set the number of items displayed per page.

Column Guide

The **supported environment** for each column indicates the conditions under which data for that column can be queried.

- **PostgreSQL:** Collected from the `pg_stat_activity` view. The columns available may differ depending on the PostgreSQL version.
- **XOS:** Can only be queried when the XOS agent is additionally installed.
- **Common:** Always available regardless of the DB version or agent installation status.

Basic Information

Column	Description	Supported Environment
time	Time when session information was collected	Common
instance	DB instance name	Common
pid	Backend process ID	PostgreSQL
datname	Name of the connected database	PostgreSQL
username	Connected user name	PostgreSQL
client_addr	Client IP address	PostgreSQL
client_hostname	Client host name	PostgreSQL
client_port	Client port number	PostgreSQL
application_name	Client application name	PostgreSQL

Execution Information

Column	Description	Supported Environment
state	Current session status (active, idle, idle in transaction, etc.)	PostgreSQL
runtime	Execution time of the current query (seconds)	PostgreSQL
xact_runtime	Execution time of the current transaction (seconds)	PostgreSQL
wait_event	Name of the currently waiting event	PostgreSQL
wait_event_type	Type of the wait event (Lock, LWLock, IO, IPC, etc.)	PostgreSQL 9.6 or later

Column	Description	Supported Environment
state_change	Time when the status last changed	PostgreSQL
query	SQL statement being executed	PostgreSQL
query_param	Bind parameter values of the SQL	Common
query_hash	SQL hash value	Common

Time Information

Column	Description	Supported Environment
xact_start	Start time of the current transaction	PostgreSQL
query_start	Start time of the current query	PostgreSQL
backend_start	Start time of the backend process	PostgreSQL

Backend Information

Column	Description	Supported Environment
backend_type	Backend process type (client backend, autovacuum worker, etc.)	PostgreSQL 10 or later
backend_xid	Current transaction ID of the backend	PostgreSQL 9.4 or later
backend_xmin	xmin horizon of the backend	PostgreSQL 9.4 or later
query_id	Query ID	PostgreSQL 14 or later

Resource Usage

Column	Description	Supported Environment
<code>cpuusage</code>	CPU usage	XOS
<code>cputime</code>	CPU usage time	XOS
<code>rss</code>	Resident Set Size (physical memory usage)	XOS
<code>pss</code>	Proportional Set Size (proportional memory usage)	XOS
<code>vsize</code>	Virtual memory size	XOS
<code>ioread</code>	I/O read amount	XOS
<code>iowrite</code>	I/O write amount	XOS

Data Interpretation Guide

Key Checkpoints

Column	Meaning	Interpretation
<code>state</code>	Current session status	active: Executing SQL, idle in transaction: Idle within a transaction → May cause lock issues if maintained for a long time
<code>runtime</code>	Execution time of the current query	Higher values indicate long-running queries → Tuning target
<code>xact_runtime</code>	Execution time of the current transaction	Higher values indicate long-held transactions → Possible cause of lock contention or vacuum delays

Column	Meaning	Interpretation
wait_event	Currently waiting event	Lock type: Lock contention occurring, IO type: Disk I/O bottleneck
cpuusage	CPU usage	High values indicate CPU-intensive query → Query optimization needed

Identifying Abnormal Sessions

- Sessions with abnormally high `runtime` : Long-running queries → Review execution plan, index tuning
- Sessions with `state` as idle in transaction and long `xact_runtime` : Transaction is open but idle → Causes lock waits for other sessions and vacuum delays
- Multiple sessions with `wait_event` of **Lock type**: Lock contention on a specific table or row → Review transaction patterns
- Sessions with `wait_event_type` as **IO**: Disk I/O bottleneck → Query optimization, review memory settings such as `shared_buffers`

Practical Use Scenarios

Scenario 1. Analyzing a Sudden Session Spike at a Specific Time

1. Check the **interval where the session count spikes** in the session trend chart
2. Check `state` and `wait_event` in the session list for that time period
3. Analyze whether the same `query` repeats or if requests are concentrated from a specific `client_addr`
4. Use filters to classify sessions by `state` or `wait_event_type`

Scenario 2. Tracking Long-Running Sessions

1. Add a `runtime` filter in the field panel to query long-running sessions
2. Check the `query` of that session to identify which SQL has been running for a long time
3. Check the `wait_event` to determine if the query is executing or waiting for a lock

4. If necessary, check the execution plan (Plan) of that SQL

Scenario 3. Checking idle in transaction Sessions

1. Add "idle in transaction" condition to the `state` filter in the field panel
2. Check sessions with long `xact_runtime` to understand how long the transaction has been open
3. Identify the source application using the session's `application_name` and `client_addr`
4. Note that long-held transactions can cause lock waits for other sessions and vacuum delays

Scenario 4. Analyzing Lock Contention Causes

1. Check the interval with a high proportion of red (lock wait) in the session trend chart
2. Add "Lock" condition to the `wait_event_type` filter in the field panel
3. Check the `query` and `datname` of waiting sessions to identify the contention target
4. Check if sessions with long `xact_runtime` are holding locks

Additional Features

Using Filters

1. Click the filter add button or click a field name in the field panel.
2. Select a filter key.
 - **String:** `=` (contains), `!=` (does not contain)
 - **Number:** `==` (equals), `>=` (greater than or equal to), `<=` (less than or equal to)
3. Enter a value and apply.
4. Apply **AND logic** for multiple conditions.

Table Column Settings

You can add/remove columns to display and change the order by dragging through the column selection feature in the table header.

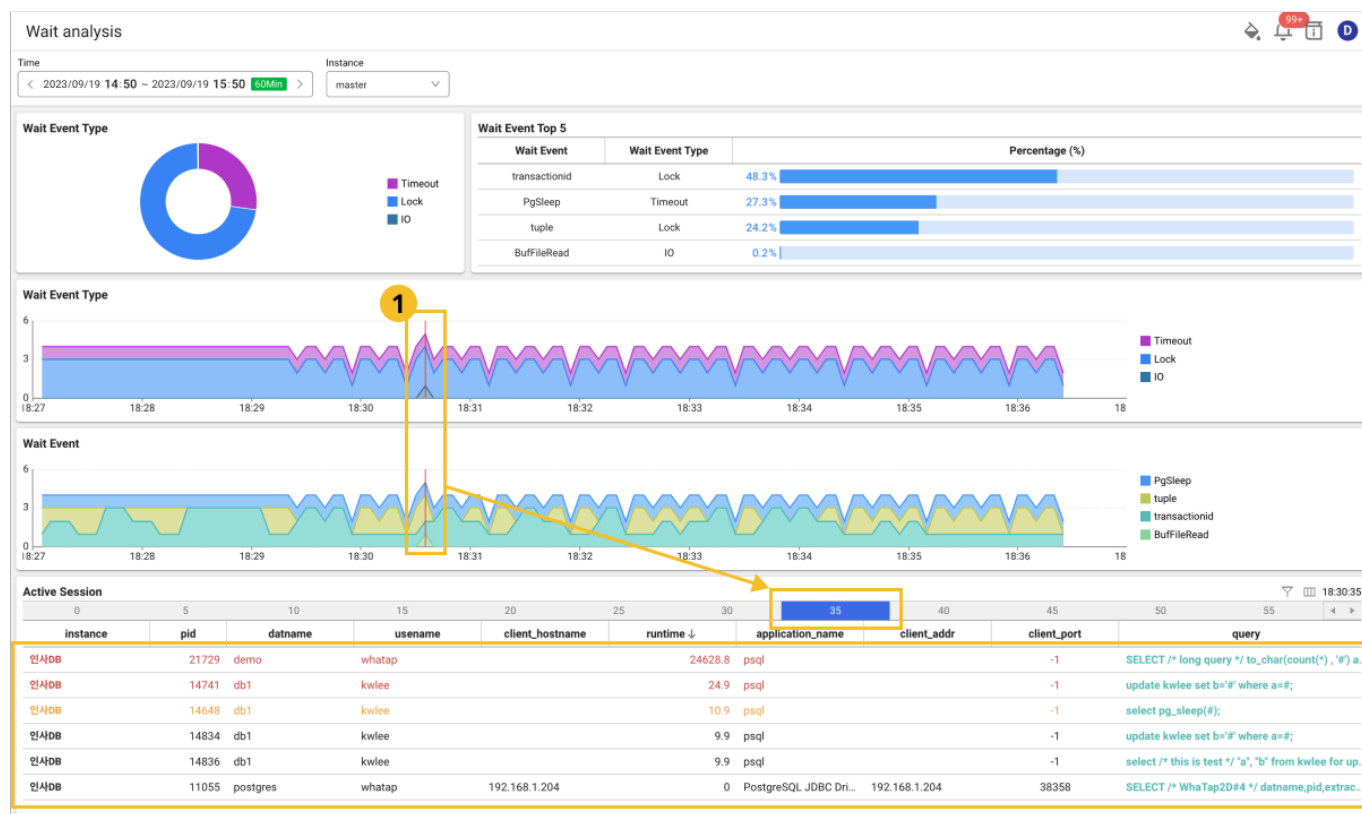
Wait Analysis

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

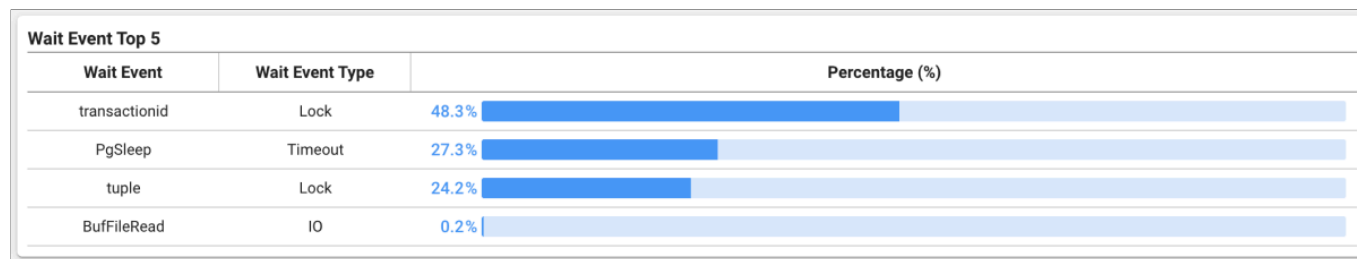
You can check in which wait event the active session is waiting and which wait events occur frequently. Active sessions collected every 5 seconds can be analyzed from the wait perspective.

- ! This function is supported in DBX agent 1.6.10 or later.
- The collected wait data includes `wait_event` and `wait_event_type` in the `pg_stat_activity` view. For more information about the metric, see [the following](#).

Basic screen guide



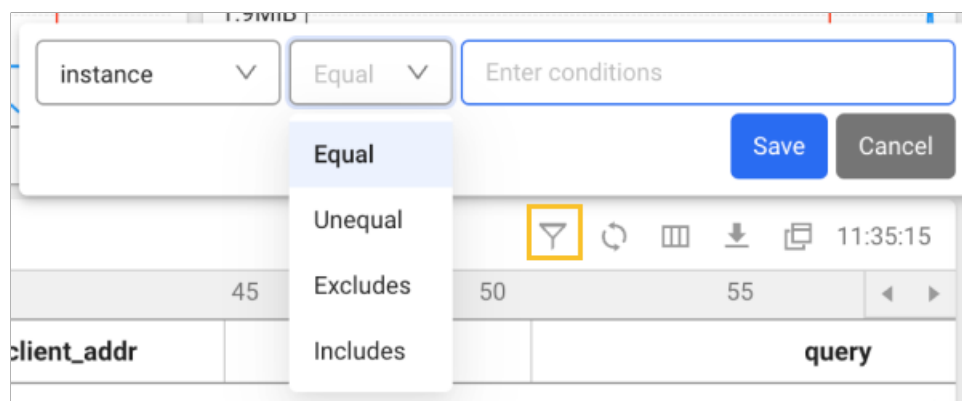
Select the **Time** and **Instance** to view. You can see the wait data corresponding to the set time and instance. If there are active sessions that frequently wait (delay) during the query period, it can be used to analyze what types of wait events occurred.



In [Wait Event Top 5](#), you can see which wait events occurred most frequently. Using the above image as an example, the most frequently occurred type of wait event is **Lock**. If you select the ① detailed time in the graph chart, you can view the active sessions in the time and check the SQL statements in which the wait events occurred.

- ① • 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** windows, see [the following](#).
- When you select the [query](#) column in the [Active Session](#) list, the **SQL details** window appears. For more information about the **SQL details** windows, see [the following](#).

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**.

Vacuum Analysis

Vacuum Analysis is a screen that **visually reconstructs** the Bloating, Dead Tuple, Age, and other information provided by Top Objects from a **Vacuum perspective**. You can get an at-a-glance view of the Vacuum status of an instance through Summary cards, Top 5 charts, and Vacuum execution history.

- With **Vacuum Health Summary**, you can quickly understand the overall Vacuum health status of the instance.
- In **Table Analysis**, you can identify tables that need attention based on Dead Tuple and Bloat Size criteria.
- With **Vacuum Execution History**, you can check Vacuum execution patterns and history in time series.

! What is Vacuum?

Because PostgreSQL uses MVCC (Multi-Version Concurrency Control), existing rows are not immediately deleted when UPDATE or DELETE is performed, but remain as **Dead Tuples**. Vacuum is a maintenance operation that cleans up these Dead Tuples so disk space can be reused.

If Vacuum is not executed in a timely manner, the following problems can occur:

- **Increased table Bloat:** Unnecessary space increases, leading to higher disk usage and query performance degradation
- **Transaction ID Wraparound:** When Transaction Age reaches the limit, a safety mechanism kicks in that automatically stops DB writes

Prerequisites

DB configuration

Grant permissions to the monitoring account so that the agent can query schema information.

```
-- Grant schema access permission
GRANT USAGE ON SCHEMA {schema_name} TO {DB_User};

-- Grant query permission for existing tables
```

```
GRANT SELECT ON ALL TABLES IN SCHEMA {schema_name} TO {DB_User};
```

```
-- Automatically grant query permission for tables created in the future
```

```
ALTER DEFAULT PRIVILEGES FOR USER {owner_name} IN SCHEMA {schema_name} GRANT SELECT ON TABLES TO {DB_User};
```

Statement	Description
GRANT USAGE ON SCHEMA	Grants permission to access the schema.
GRANT SELECT ON ALL TABLES IN SCHEMA	Grants query permission for existing tables in the schema.
ALTER DEFAULT PRIVILEGES ... GRANT SELECT	Automatically grants query permission for new tables created in the schema in the future.

Agent configuration

Check the following settings in whatap.conf. Since the default value of `pg_object` is `true`, **collection occurs without any additional configuration.**

```
pg_object=true
```

Option	Type	Default	Description
pg_object	boolean	true	Configures whether to collect Top Object information. Default is <code>true</code> .
pg_object_hour	int	5	Configures the collection time for Top Object information. Default is <code>5</code> , collection begins at 5 AM.

Data collection sources

The data on the Vacuum Analysis screen is collected from PostgreSQL system catalogs and statistics views.

Section	Main Source	Notes
Dead Tuples	<code>pg_class</code> , <code>pg_namespace</code>	Queries tuple count using <code>pg_stat_get_live_tuples()</code> and <code>pg_stat_get_dead_tuples()</code> functions.
Top Bloat Size	<code>pg_class</code> , <code>pg_stats</code> , <code>pg_stat_user_tables</code>	Estimates Bloat size based on column statistics (<code>null_frac</code> , <code>avg_width</code>).
Transaction Age	<code>pg_catalog.pg_database</code> , <code>pg_catalog.pg_settings</code>	Calculates Transaction Age by database using <code>age(datfrozenxid)</code> and compares with the <code>autovacuum_freeze_max_age</code> setting.
Vacuum Execution History	<code>pg_stat_progress_vacuum</code> , <code>pg_stat_activity</code>	Queries the progress, wait events, and query information of running Vacuum.

Analyzing Vacuum

You can analyze the table status and whether maintenance is needed based on Vacuum-related metrics. The key metrics are as follows.

Basic options

In the top filter area, you can set the time and target instance.

- **Time:** Select the date to query.
- **Instance:** Select the target DB instance.

Vacuum Health Summary

Provides 3 summary cards with key Vacuum-related metrics for the instance. Dead Tuples and Top Bloat Size are collected **once a day** (based on the `pg_object_hour` setting time), and the collection time is displayed in parentheses next to the card title. The **Table Analysis** data at the bottom is also collected at the same time.

Card	Description
Dead Tuples	Displays total Dead Tuple count and distribution by database.
Top Bloat Size	Displays the largest table Bloat size (MB) and distribution by database.
Transaction Age	Displays the Age of the oldest transaction and the percentage (%) against the total limit. The percentage by database can also be checked.

Dead Tuples

If Dead Tuple counts are high, Vacuum may not be executing in a timely manner. Clicking a database name filters the **Table Analysis** Dead Tuple Top 5 at the bottom by that database.

! When Dead Tuples are high

- Execute `VACUUM table_name;`
- Dead Tuples cause query performance degradation and increased disk usage.
- Check whether Autovacuum is enabled in settings and whether the threshold is too high.

Top Bloat Size

Bloat refers to a state where a table is excessively large relative to actual data due to Dead Tuples, etc. Large Bloat wastes unnecessary disk space and degrades sequential scan performance. Clicking a database name filters the **Table Analysis** Bloat Size Top 5 at the bottom by that database.

! When Bloat Size is high

- Execute `VACUUM FULL table_name;`
- VACUUM FULL causes table locking, so it is recommended to perform it outside business hours.
- Severe Bloat causes query performance degradation, index inefficiency, and disk waste.
- Also check Autovacuum settings, UPDATE/DELETE patterns, and whether HOT Update is possible.



- Using `pg_repack` allows space reclamation without table locking.

Transaction Age

PostgreSQL's Transaction ID can use approximately 2 billion IDs. When Transaction Age approaches this limit, **XID Wraparound** occurs and the DB may be forcibly shut down to prevent data corruption.

- Agent version **v2.59.01** or later is required, and data is collected at **5-second intervals**.



When Transaction Age is high

- Execute `VACUUM FREEZE database_name;`
- If Transaction ID exceeds 2 billion, XID wraparound occurs and the DB may be forcibly shut down to prevent data corruption.
- Be sure to identify and resolve the cause:
 - Long-running transactions
 - Unprocessed replication slots
 - Disabled autovacuum, etc.

Threshold settings

Clicking the **Threshold Settings** button in the upper right of the screen allows you to set thresholds for Dead Tuples and Bloat Size. When a threshold is set, the currently configured value is displayed at the top, and cards exceeding the threshold have their **background color changed** so you can quickly recognize abnormal states. To clear the settings, click the **Clear Settings** button in the threshold settings window.

Metric	Setting	Description
Dead Tuples	Warning $\geq$	Warning-level Dead Tuple count
	Critical $\geq$	Critical-level Dead Tuple count
Bloat Size (GB)	Warning $\geq$	Warning-level Bloat size (GB)

Metric	Setting	Description
	Critical $\geq$	Critical-level Bloat size (GB)

- When **Warning** is exceeded: Card background changes to **orange**.
- When **Critical** is exceeded: Card background changes to **red**.

Table Analysis

Provides Top 5 charts for analyzing Vacuum-related metrics by table.

Dead Tuple Top 5

Displays the top 5 tables with the most Dead Tuples as a horizontal bar chart. Table names are displayed in `database.schema.table` format.

- Tables with many Dead Tuples are in a state that requires Vacuum.
- Detailed data can be checked using filter and table view toggle buttons.

Bloat Size Top 5

Displays the top 5 tables with the largest Bloat Size as a horizontal bar chart. The unit is bytes.

- For tables with large Bloat Size, consider reclaiming space with `VACUUM FULL` or `pg_repack`.
- Detailed data can be checked using filter and table view toggle buttons.

Vacuum Execution History

Displays the Vacuum execution history for the selected period as a time-series chart.

- You can identify patterns of whether Vacuum is being executed periodically or concentrated in specific time periods.
- If there are long intervals without Vacuum execution, Autovacuum settings should be checked.

Vacuum & Analyze History

Clicking a specific time period in the chart displays the **Vacuum & Analyze History** table at the bottom, where you can check details of Vacuum and Analyze operations performed in that time period (1-hour units).

Column	Description
Time	Vacuum/Analyze execution time
Database	Target database name
Phase	Vacuum execution phase (e.g., scanning heap, vacuuming indexes, etc.)
PID	Process ID that performed Vacuum
Query	Executed Vacuum/Analyze query
Heap Scanned	Number of Heap blocks scanned
Heap Total	Total number of Heap blocks
Heap Vacuumed	Number of Heap blocks processed by Vacuum
Index Vacuum	Number of index Vacuum executions
Max Dead Tuples	Maximum number of Dead Tuples that can be processed at once
Dead Tuples	Number of Dead Tuples found
Runtime	Vacuum execution time
Scanned %	Scan ratio relative to total Heap
Vacuumed %	Vacuum processing ratio relative to total Heap
Wait Event	Wait event name

Column	Description
Wait Event Type	Wait event type

Configuration

Displays Vacuum-related parameters configured for the current DB instance for reference. Only Vacuum-related items among PostgreSQL's various DB parameters are selected and shown, so you can check Autovacuum behavior or I/O cost settings without connecting to the DB separately.

AUTOVACUUM SETTINGS

Parameter	Description
autovacuum	Whether the Autovacuum daemon is enabled
autovacuum_max_workers	Maximum number of Autovacuum worker processes that can run simultaneously
autovacuum_naptime	Autovacuum execution interval (unit: seconds)
autovacuum_vacuum_threshold	Minimum number of Dead Tuples required to trigger Vacuum
autovacuum_vacuum_scale_factor	Ratio of Dead Tuples to table size to trigger Vacuum
autovacuum_vacuum_cost_delay	I/O cost delay time for Autovacuum (unit: milliseconds)
autovacuum_vacuum_cost_limit	I/O cost limit that Autovacuum can accumulate before pausing

Autovacuum executes Vacuum on a table when Dead Tuples of

$\text{autovacuum_vacuum_threshold} + \text{autovacuum_vacuum_scale_factor} \times \text{table row count}$ have occurred.

I/O IMPACT SETTINGS

Parameter	Description
<code>vacuum_cost_page_hit</code>	Cost when a page is found in shared buffer
<code>vacuum_cost_page_miss</code>	Cost when a page needs to be read from disk
<code>vacuum_cost_page_dirty</code>	Cost when a page is modified (dirtied)

Data interpretation guide

Key metrics interpretation

Metric	Normal	Warning	Critical
Dead Tuples	Small amount relative to table	Continuously increasing trend	Large accumulation, Vacuum not running
Bloat Size	Below 20% of table size	20–50% of table size	Above 50% of table size
Transaction Age	Below 5%	Above 50%	Above 80% → Immediate action required

Situations requiring action

- **Dead Tuples continuously increasing**

Autovacuum cannot keep up with the workload → Lower `autovacuum_vacuum_scale_factor` , increase `autovacuum_max_workers`

- **Tables with large Bloat Size**

Normal Vacuum does not return space to the OS → Consider `VACUUM FULL` or `pg_repack` (Note:

`VACUUM FULL` causes table locking)

- **High Transaction Age ratio**

Vacuum cannot clean up old transactions → Check long-running transactions, consider running manual

`VACUUM FREEZE`

- **Gaps in Vacuum Execution History**

Autovacuum is disabled or workers are insufficient and cannot run → Check Autovacuum settings

Analysis using Top Objects

Vacuum Analysis is a screen that visually reconstructs the **Bloating**, **Dead Tuple**, and **Age** information provided by [Top Objects](#) from a Vacuum perspective. Both screens are based on the same data (collected at `pg_object_hour` time), but provide different perspectives depending on the analysis purpose.

Comparison Item	Vacuum Analysis	Top Objects
Analysis Perspective	Vacuum maintenance-focused	Object (table/index)-focused
Data Scope	Bloating, Dead Tuple, Age + Vacuum execution history	6 types: Bloating, Scan, DML, Analyze Time, Age, Dead Tuple
Representation	Summary cards + Top 5 charts + time-series history	Sortable table list (Top N)
Unique Features	Vacuum Health Summary, Vacuum execution history, threshold alerts, DB parameter view	Scan/DML analysis, Object Detail, index analysis

Usage guide

- ✓ • For the purpose of **understanding overall Vacuum status** → Quickly check the current situation with Summary cards and Top 5 charts on the **Vacuum Analysis** screen.
- For **detailed analysis of a specific table** → Analyze in detail with various tabs and sort criteria on the



Top Objects screen.

- When **identifying tables that need Vacuum and then analyzing the cause**, using both screens together is effective.

Top Object

Home > Select Project > **Analysis** > **Top Object**

It provides objects with the bloating size, scan count, DML execution count, and objects with many dead tuples for each object (table, index) of individual instance, and provides information to identify objects that are used a lot and perform Vacuum or Analyze at appropriate times.

Top Object

Bloating

Scan

DML

Analyze Time

Age

Dead Tuple

Table

Index

Date

Instance

Lookup criteria

Number of hits

< 2023-11-07 >

demo

bloat_ratio

50

🔍

📄

datname	schemaname	tablename	est_rows	table_size	bloat_size	bloat_ratio ↓	time
db1	public	whatap	42	24KiB	16KiB	67%	2023-11-07 16:25:40
db1	public	kwlee	3	16KiB	8KiB	50%	2023-11-07 16:25:40
demo	bookings	t1	4000000	253.3MiB	85.1MiB	34%	2023-11-07 16:25:40
demo	bookings	t3	4000000	253.3MiB	85.1MiB	34%	2023-11-07 16:25:40
demo	bookings	t2	4000000	253.3MiB	85.1MiB	34%	2023-11-07 16:25:40
demo	bookings	ticket_flights	1045650	68MiB	8.1MiB	12%	2023-11-07 16:25:40
demo	bookings	tickets	366733	48MiB	1.7MiB	4%	2023-11-07 16:25:40
demo	bookings	flights	33121	3MiB	24KiB	1%	2023-11-07 16:25:40
demo	bookings	bookings	262788	13MiB	8KiB	0%	2023-11-07 16:25:40
demo	bookings	seats	1339	64KiB	0	0%	2023-11-07 16:25:40
demo	bookings	airports_data	104	24KiB	0	0%	2023-11-07 16:25:40
db2	public	db2_kwlee	2	8KiB	0	0%	2023-11-07 16:25:40
demo	bookings	boarding_passes	579686	33.3MiB	48KiB	0%	2023-11-07 16:25:40
demo	bookings	aircrafts_data	9	8KiB	0	0%	2023-11-07 16:25:40

- Set **Date**, **Instance**, **Lookup criteria**, and **Number of hits** to view on the screen and then select . The results for the selected conditions appear in the table.
- If you select the **tablename** or **indexname** column in the inquiry list, the **Object detail** window appears where you can see the configuration for **column** and **index**.

Top Object

Bloating

Scan

DML

Analyze Time

Age

Dead Tuple

Table

Index

Date

Instance

Lookup criteria

Number of hits

<

2023-12-06

>

DBX-2-60-5444

▼

bloat_ratio

▼

50

datname	schemaname	tablename	
whatap	public	whatap	3000010
whatap	public	whatap3	10
whatap	public	whatap2	11
whatap	whatap	whatap	10
whatap	Test1	Test	3
whatap	whatap2	whatap	1
whatap	public	whatap5	2
whatap	public	whatap4	10
whatap	public	Whatap	1

Object Detail

whatap

type: table

seq_scan: 0

idx_scan: 0

n_live_tup: 0

n_dead_tup: 0

last_autovacuum:

last_analyze:

last_autoanalyze:

Column List

No	column_name	data_type	is_nullable	column_default	comment
1	a	integer	YES		whatap column a
2	b	character varying(20)	YES		

Index List

index_name	columns	type	is_unique	condition	comment
whatap_idx	a	BTREE	f		
whatap_b_idx	b	BTREE	f		

❗ **Object detail** is provided in the DBX agent 1.6.15 or later. For more information about the database roles and related settings, see [the following](#).

- ❗
- The Top Object data is collected once a day. For more information about the related agent settings, see [the following](#).
 - To view data, the DB monitoring account requires the following roles:

```
grant select on all tables in schema {schema_name} to whatap;
```

- This function is supported in DBX agent 1.6.13 or later.

Bloating

Bloating is a phenomenon in which the object size increases due to an increase of unused tuples.

Tables and indexes with the values greater than the collection reference set in `table[index]_bloat_ratio[bytes]` in the agent configuration ([whatap.conf](#)), appear in the list. If there is no corresponding object, data may not be viewed.

✓ Tuning tips

For tables with higher `Bloat_ratio`, consider performing the Vacuum operation. For more information, see the following document.

- [Measures to deal with the PostgreSQL table bloating](#)
- [Dealing with significant Postgres database bloat — what are your options?](#)

If **Bloating** is viewed, the following columns are displayed.

Type	Column name	Description
Table	<code>datname</code>	Database name
	<code>schemaname</code>	Schema name
	<code>tablename</code>	Table Name
	<code>est_rows</code>	Value predicted by the number of Dead tuple + Live tuple
	<code>table_size</code>	Table size
	<code>bloat_size</code>	Expected table size inflated by Dead tuple
	<code>bloat_ratio</code>	Inflated size ratio.
Index	<code>datname</code>	Database name
	<code>schemaname</code>	Schema name
	<code>tablename</code>	Table Name
	<code>indexname</code>	Index name
	<code>table_size</code>	Table size
	<code>index_size</code>	Index size

Type	Column name	Description
	bloat_size	Expected index size inflated by Dead tuple
	bloat_ratio	Inflated size ratio.
	index_scans	Number of index scans when using indexes

Scan

`Seq_scan` means the full scan without using indexes, and `idx_scan` means the number of index uses.

✓ Tuning tips

Consider creating indexes for tables with higher `Seq_scan` and deleting indexes with lower `idx_scan` .
For more information, see [the following](#).

If **Scan** is viewed, the following columns are displayed.

Type	Column name	Description
Table	datname	Database name
	schemaname	Schema name
	tablename	Table Name
	seq_scan	Number of sequential scans (full scans) of the table
	seq_tup_read	Number of live rows from sequential scans
	idx_scan	Number of index scans for the table
	idx_tup_fetch	Number of live rows read by index scan

Type	Column name	Description
Index	<code>datname</code>	Database name
	<code>schemaname</code>	Schema name
	<code>tablename</code>	Table Name
	<code>indexname</code>	Index name
	<code>idx_scan</code>	Number of index scans
	<code>idx_tup_fetch</code>	Number of valid table rows extracted from index scans using indexes
	<code>idx_tup_read</code>	Number of index items returned by index scan

DML

Tables with higher `dml_count` values are frequently used tables.

✓ Tuning tips

Identify the main tables in your system and then use them as references when changing tables or designing the architecture.

If **DML** is viewed, the following columns are displayed.

Column name	Description
<code>datname</code>	Database name
<code>schemaname</code>	Schema name
<code>tablename</code>	Table Name

Column name	Description
dml_count	n_tup_ins + n_tup_upd + n_tup_del
n_tup_ins	Number of inserted lines
n_tup_upd	Number of updated lines
n_tup_del	Number of deleted lines
n_tup_hot_upd	Number of hot updated lines (e.g. no separate index update required)

Analyze Time

This list includes objects for which Analyze and Vacuum operations have become outdated.

✓ Tuning tips

Older tables may have inaccurate statistical information. Review the operations. For more information, see [the following](#).

If **Analyze Time** is viewed, the following columns are displayed.

Column name	Description
datname	Database name
schemaname	Schema name
tablename	Table Name
last_analyze	Last time of manual table analysis
last_autoanalyze	Last time of table analysis by the autovacuum daemon

Column name	Description
<code>last_autovacuum</code>	Last time of table cleanup (vacuum) by the autovacuum daemon
<code>analyze_count</code>	Number of manual analyses
<code>last_vacuum</code>	Last time of table vacuumed manually (VACUUM FULL not applicable)
<code>autoanalyze_count</code>	Number of analyses by the autovacuum daemon
<code>autovacuum_count</code>	Number of cleanups (vacuum) by the autovacuum daemon
<code>vacuum_count</code>	Number of manual cleanups (vacuum) (VACUUM FULL excluded)
<code>n_mod_since_analyze</code>	Number of lines changed since last analysis (analyze)

Age

In PostgreSQL, XID (transaction id) is used recursively so that the XID may wrap around at a certain point. To prevent this, `xid_age` (Current XID - XID at the time of creation) must be managed so that it does not keep increasing. If `autovacuum_freeze_max_age` exceeds, Anti-Wraparound Vacuum is automatically performed and `xid_age` is managed below `autovacuum_freeze_max_age`.

This list displays vacuum targets that meet the following criteria: If there is no target, data may not be viewed.

- Table with dead tuples more than vacuum threshold (`autovacuum_threshold` + `autovacuum_scale_factor` * `number-of-tuples`).
- Table with the age (`relfrozenxid`) greater than `autovacuum_freeze_max_age`

✓ Tuning tips

If `xid_age` continues to increase, XID cleaning may not be performed under the set Auto Vacuum condition. Adjust the auto vacuum parameters or review the vacuum execution manually.

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

If **Age** is viewed, the following columns are displayed.

Column name	Description
<code>datname</code>	Database name
<code>schemaname</code>	Schema name
<code>tablename</code>	Table Name
<code>xid_age</code>	Max age of the table, <code>xid</code> (transaction ID) of the oldest record
<code>per_to_wraparound</code>	Metric that displays the remaining margin up to the <code>xid</code> (transaction ID) wrapping as a percentage. Formula: $\text{per_to_wraparound}(\%) = \text{xid_age} / \text{Autovacuum_freeze_max_age} * 100$
<code>table_size</code>	Table size
<code>autovacuum_vacuum_tuples</code>	If the number of dead tuples exceeds this value, autovacuum is performed.
<code>dead_tuples</code>	Number of dead tuples
<code>autovacuum_freeze_max_age</code>	The default value is set to 200 million.

Dead Tuple

Dead tuple is the data that has already been deleted using Delete or Update. Review the Vacuum operation to change to free space.

✓ Tuning tips

To reduce dead tuples, review the Vacuum operation. For more information, see [the following](#).

If **Dead Tube** is viewed, the following columns are displayed.

Column name	Description
<code>datname</code>	Database name
<code>schemaname</code>	Schema name
<code>tablename</code>	Table Name
<code>dead_tuple</code>	Number of unused tuples due to Delete or Update
<code>dead_tuple_ratio</code>	Proportion of unused tuples
<code>live_tuple</code>	Number of unused tuples
<code>live_tuple_ratio</code>	Proportion of used tuples
<code>total_relation_size</code>	Total size of relation including the index and TOAST data
<code>total_tuple</code>	Number of all tuples

Agent setting

The following is the agent setting to view Top Object information. Check the options required in whatap.conf.

- **pg_object** Boolean

Default `true`

Set whether or not to collect data from the Top object. To collect data, change it to `true`.

- **pg_object_hour** Int

Default `5`

Set the time to collect data from the Top object. The default value is `5` and collection starts at 5 AM.

- **table_bloat_ratio** Percentage

Default `50`

Data is collected when the table's bloating ratio is greater than the set value. The default value is 50%.

- **table_bloat_bytes** Byte

Default 10485760

Data is collected when the table's bloating bytes is greater than the set value. The default value is 10 MB.

- **index_bloat_ratio** Percentage

Default 50

Data is collected when the index's bloating ratio is greater than the set value. The default value is 50%.

- **index_bloat_bytes** Byte

Default 10485760

Data is collected when the index's bloating bytes is greater than the set value. The default value is 10 MB.

- **autovacuum_list_limit** Int

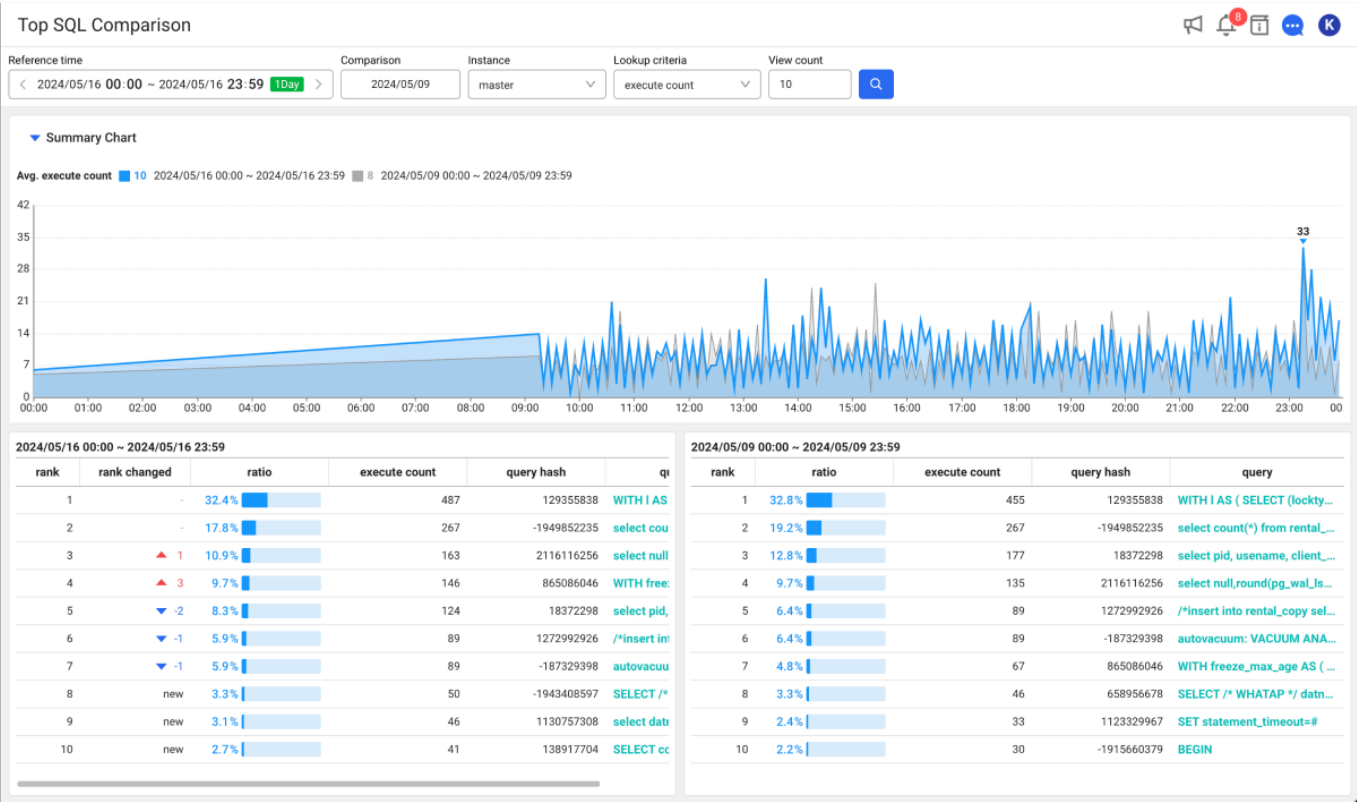
Default 50

Table data is collected according to the set value from the oldest age (`relfrozenxid`). The default value is 50.

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.



- **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**.

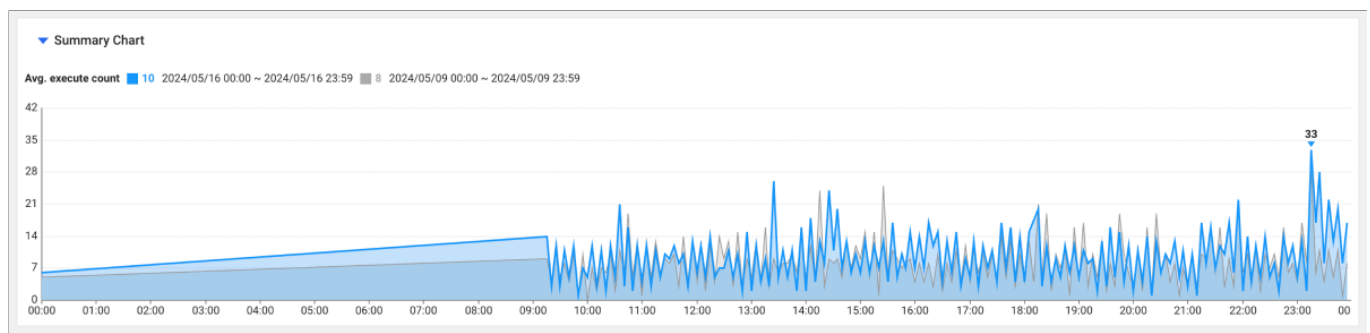
Date	Hour
Comparison 2024/05/16 Instance master Date Hour 1 Days Ago 2024/05/16 7 Days Ago 2024/05/10 30 Days Ago 2024/04/17 Custom 2024-05-16 Cancel Confirm	Comparison 2024/05/16 Instance master Date Hour (n)hours ago 1 Cancel Confirm

- **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.

View SQL Details

To see the detailed SQL information, select the **query** column in the SQL ranking table. The **SQL details** window appears. You can check the SQL query statements and plan information.

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

The screenshot displays the SQL details window for a query. At the top, a 'query' box contains the text 'START_REPLICATION' and 'SELECT /* ...'. Below this, the 'SQL details' window is open, showing the query text on the left, a table of statistics in the middle, and the execution plan on the right.

Query (hash: 1098839306)

```
SELECT
/* long query */
to_char(count(*), 'DD.MM.YYYY') as
when,
'n' as departure,
'c' as arrival,
's' as status,
'a' as seat_no
FROM
bookings.ticket_flights a,
bookings.boarding_passes b
union all
SELECT
to_char(f.scheduled_departure, 'DD.MM.YYYY') AS
when,
f.departure_city || ' ' || f.departure_airport || ' ' AS departure,
f.arrival_city || ' ' || f.arrival_airport || ' ' AS arrival,
f.status,
bp.seat_no
FROM
ticket_flights tf
JOIN flights_v f ON tf.flight_id = f.flight_id
LEFT JOIN boarding_passes bp ON tf.flight_id = bp.flight_id
AND tf.ticket_no = bp.ticket_no
WHERE
1 = 1
and tf.ticket_no = '0005432661915';
```

Plan demo whatap Text JSON

Rows	Node Cost	Plans
4	0%	Append
1	13.9%	Aggregate
606149825272	86.1%	Nested Loop Inner join
579686	0%	Seq Scan on boarding_passes as b
1045652	0%	Materialize
1045652	0%	Seq Scan on ticket_flights as a
3	0%	Nested Loop on ((tf.ticket_no = bp.ticket_no) AND (tf.flight_id = bp.flight_id)) Left join
3	0%	Nested Loop Inner join
3	0%	Nested Loop Inner join
3	0%	Nested Loop Inner join
3	0%	Index Only Scan on ticket_flights as tf on (tf.ticket_no = '0005432661915':bpchar)
1	0%	Index Scan on flights as f on (f.flight_id = tf.flight_id)
1	0%	Index Scan on airports_data as ml on (ml.airport_code = f.departure_airport)
1	0%	Index Scan on airports_data as ml_1 on (ml_1.airport_code = f.arrival_airport)
3	0%	Materialize
3	0%	Bitmap Heap Scan on boarding_passes as bp
3	0%	Bitmap Index Scan on (bp.ticket_no = '0005432661915':bpchar)

- **Default Bind:** You can bind the part set as a variable in the query sentence to the default value.
- **Reset Bind:** You can release the bound default value and check the variable.
- **Formatting:** You can improve readability by applying indentation and formatting to a SQL query statement.
- **Plan:** To check plan information, enter **DB Name**, **User Name**, and **Password**, and then select .
 - **Text:** You can check plan information in text format.
 - **JSON:** By viewing the plan information in Json format, you can quickly find the points where bottlenecks occur.

The function to view plan information in **JSON** format is supported only in DBX agent 1.6.12 or later.

- ✓ If you select a table or index while viewing the plan information in **JSON** format, the **Object detail** window appears where you can see the configurations for **columns** and **index** items in the table.



The screenshot displays the SQL details interface. The top section shows the query plan for a query with hash 1726024768. The plan includes a Hash Join Left join on (s.usersysid = u.oid), followed by a Hash Join Left join on (s.datid = d.oid), a Function Scan as s, a Hash, a Seq Scan pg_database as d, another Hash, and a Seq Scan pg_auth as u. The bottom section shows the object detail for pg_database, including its type (table), indexes (pg_database_datname_idx, pg_database_oid_index), and statistics (seq_scan, idx_scan, n_live_tup, n_dead_tup, last_autovacuum, last_autoanalyze).

This function is supported in DBX agent 1.6.15 or later. For more information about the database roles and related settings, see [the following](#).

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.
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](https://whatap.io/docs/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
```

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

master

View Changes

	2024/06/14 09:24:25 Today	2024/06/13 09:00:37
Key	Value	Value
uptime	76 days 09:09:29.998437	75 days 08:45:41.923636
start_time	2024-03-29 15:14:55.323539+00	2024-03-29 15:14:55.323539+00
zero_damaged_pages	off	off
xmloption	content	content
xmlbinary	base64	base64
work_mem	4096	4096 kB
wal_writer_flush_after	128	128 * 8kB
wal_writer_delay	200	200 ms
wal_sync_method	fdatasync	fdatasync

- 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

Comparison date

Instance

2024/06/14

2024/06/13

master

View Changes

	2024/06/14 09:28:12 Today	2024/06/13 09:00:37
Key	Value	Value
uptime	76 days 09:13:17.137711	75 days 08:45:41.923636

✔ Comparing database parameters for today

Reference date	Comparison date	Instance	
2024/06/14	2024/06/14	master	View Changes

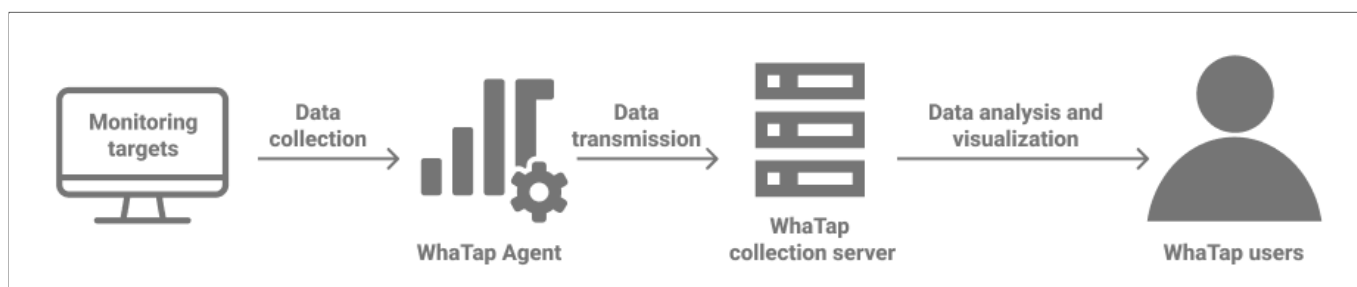
	2024/06/14 09:21:36 Today	2024/06/14 09:00:41 Today
Key	Value	Value
uptime	76 days 09:06:41.149465	76 days 08:45:46.45443
start_time	2024-03-29 15:14:55.323539+00	2024-03-29 15:14:55.323539+00
zero_damaged_pages	off	off
xmloption	content	content

If you set both **Reference date** and **Comparison date** to the date for today, you can view and compare the real-time data and the last saved data. The storage time of the last data may differ depending on the database server environment.

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

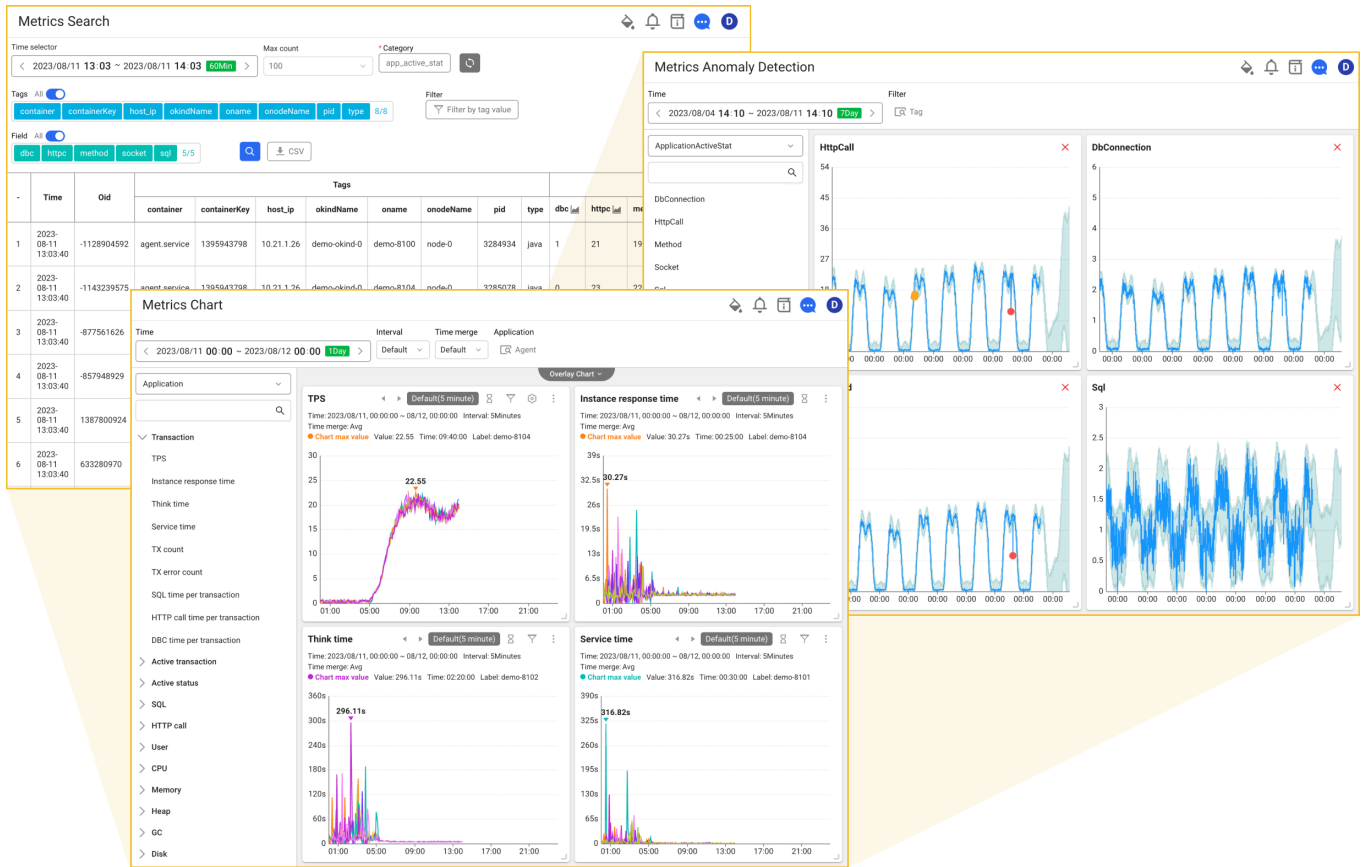
To view collected data, select a project on the Home screen, and then select [Sitemap](#) > [Metrics Search](#).

For example, select a category named `db_postgresql_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.



PostgreSQL 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 other metrics, 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_postgresql_counter

Tags

The following lists the tags that are collected in the `db_postgresql_counter` category.

Tag	Description	Remarks
<code>agentIp</code>	IP address per agent.	-
<code>dbIsMulti</code>	Whether to support the multitenant or not.	-
<code>dbName</code>	Database name.	-
<code>dbType</code>	Database type.	-
<code>dbVersion</code>	Version of the database in use.	-
<code>oname</code>	Agent name	Unique value

Field

The following lists the fields that are collected in the `db_postgresql_counter` category.

Metrics	Description	Category
<code>active_sessions</code>	Number of active sessions.	Connection
<code>CPUUtilization</code>	Percentage of CPU resources being used by the database instances.	Performance
<code>FreeStorageSpace</code>	Available storage space for the instance.	I/O
<code>FreeableMemory</code>	Amount of memory that can be used immediately by the system.	I/O

Metrics	Description	Category
NetworkReceiveThroughput	Speed of network data received by the instance	Troughput
NetworkTransmitThroughput	Speed of network data transmitted by the instance	Troughput
blk_read_time	Time taken to read a data block in the backend (in milliseconds).	I/O
blk_write_time	Time taken to write a data block in the backend (milliseconds).	I/O
blks_hit	Total count where a disk block was found in the buffer cache and did not require a read.	I/O
blks_hit_ratio	Percentage of queries retrieved from the cache out of the total queries. Calculation method: $\text{sum}(\text{blks_hit}) / \text{sum}(\text{blks_hit} + \text{blks_read}) * 100$	I/O
blks_read	Number of data blocks read directly from disk.	I/O
buffers_checkpoint	Number of buffers flushed during checkpoint.	Info
check_point	Number of cases a database checkpoint occurs.	Troughput
checkpoint_write_time	Time (milliseconds) taken to write data during checkpoint.	Troughput
checkpoints_timed	Number of checkpoints occurred according to the <code>checkpoint_timeout</code> setting.	Troughput
commit_count	Number of transaction commits.	Troughput
confl_bufferpin	Number of session conflicts due to buffer pin contention.	Troughput
confl_deadlock	Number of deadlock conflicts.	Troughput
confl_lock	Number of session conflicts due to lock contention.	Troughput
confl_snapshot	Number of session conflicts due to snapshot contention.	Troughput

Metrics	Description	Category
<code>confl_tablespace</code>	Number of session conflicts due to tablespace contention.	Troughput
<code>conflicts</code>	Number of queries that were aborted because they conflicted with database recovery operations.	Troughput
<code>connect_error</code>	Number of database connection failures.	Connection
<code>deadlock_count</code>	Total number of deadlocks occurred.	Performance
<code>dml_count</code>	Total number of DML (Data Manipulation Language) operations.	Troughput
<code>fatal_count</code>	Number of fatal errors.	Info
<code>idle_trx</code>	Number of idle transactions.	Connection
<code>idle_trx_time</code>	Cumulative time of idle transactions.	Connection
<code>idx_scan</code>	With this method, table records are accessed using the keys stored in index leaf blocks. This metric has the values queried every minute.	Performance
<code>index_hit_ratio</code>	Percentage of index queries retrieved from the cache.	Performance
<code>lock_wait_sessions</code>	Number of pending sessions.	Performance
<code>long_running_sessions</code>	Number of long running sessions.	Performance
<code>max_connections</code>	Maximum number of connections allowed. Value set in <code>pg_settings</code> . If this value is insufficient, the 'FATAL ERROR: too many clients already.' Errors may occur.	Info
<code>max_runtime</code>	Execution time of the longest running query.	Performance
<code>node_state</code>	This database consists of replication relationships. <code>1</code> : Master, <code>2</code> : Slave.	Info

Metrics	Description	Category
<code>numbackends</code>	Number of backend processes connected to the database.	Info
<code>oldest_cur_xid</code>	The oldest active transaction ID.	Info
<code>per_to_emergency_autovac</code>	Number of emergency automatic vacuum occurrences.	Performance
<code>per_to_wraparound</code>	Number of vacuum operations to prevent transaction ID wraparound.	Performance
<code>replication_count</code>	Number of active replication databases (Slave).	Performance
<code>replication_delay</code>	Replication delay time. Delay time reflected in the replication database for the replication structure, this is a metric measured in the slave DB.	Performance
<code>replication_lag</code>	Extent to which replicated data lags behind the original data.	Performance
<code>rollback_count</code>	Number of transaction rollbacks.	Performance
<code>seq_scan</code>	Method to read records by performing a full scan on the table. Select this when there is no index or when the range to be read is wide even if an index exists. This metric has the values queried every minute.	Performance
<code>table_hit_ratio</code>	Percentage of queries retrieved from the cache.	Performance
<code>temp_bytes</code>	Total amount of bytes written to temporary files by queries in the database.	Performance
<code>temp_files</code>	Number of temporary files created by queries in the database.	Performance
<code>total_runtime</code>	Total query execution time.	Performance
<code>total_sessions</code>	Total number of connected sessions.	Connection
<code>tps</code>	Number of processed transactions per second.	Performance

Metrics	Description	Category
tup_deleted	Number of tuples deleted by queries in the database.	Troughput
tup_fetched	Number of tuples retrieved from the database by queries.	Troughput
tup_inserted	Number of tuples inserted by queries in the database.	Troughput
tup_returned	Number of tuples returned by queries in the database.	Troughput
tup_updated	Number of tuples updated by queries in the database.	Troughput
uptime	Database uptime.	Info
vacuum_running_count	Number of vacuum operations running.	Performance
wait_sessions	Number of waiting sessions.	Performance
wal_count	Number of Write-Ahead Logging (WAL) records written. Spikes in the WAL file counts are usually caused by the large amount of changes in a short period or vacuum-like maintenance operations that generate temporary tables and objects. They should slowly return to normal levels. This usually results in lots of disk I/O and CPU activity, slowing the application query's execution until things get back to normal. This metric has the values queried every minute.	Performance
warning_count	Number of warnings occurred.	Info

XOS-related metrics

The following describes the metrics collected from the XOS agent.

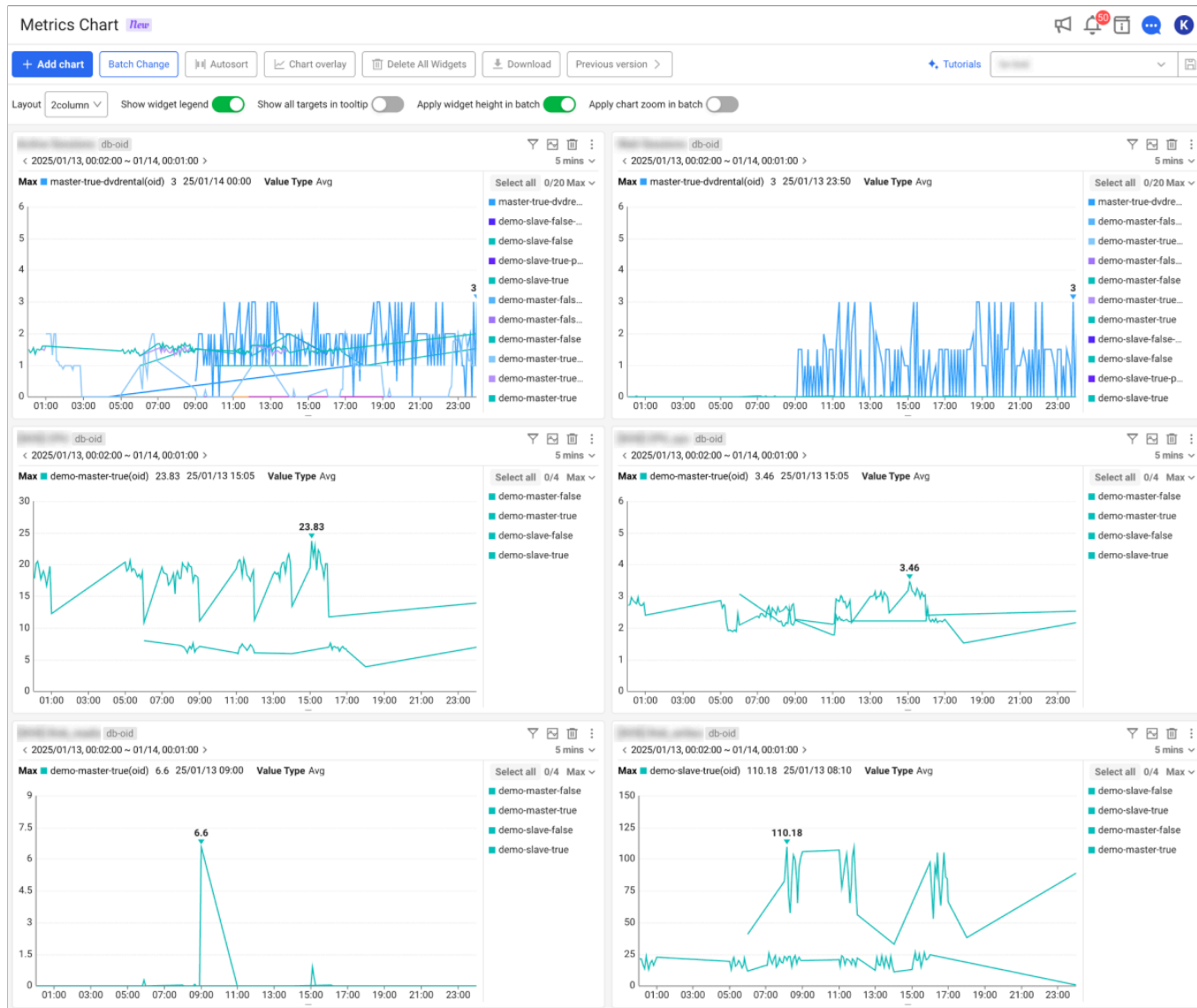
Metric	Unit	Description
<code>cpu(xos)</code>	pct	CPU utilization <code>cpu_user</code> + <code>cpu_sys</code> + <code>cpu_iowait</code>
<code>cpu_idle(xos)</code>	pct	CPU idle time
<code>cpu_iowait(xos)</code>	pct	CPU's I/O waiting time
<code>cpu_sys(xos)</code>	pct	CPU's kernel code execution time
<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
<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

Metric	Unit	Description
<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

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

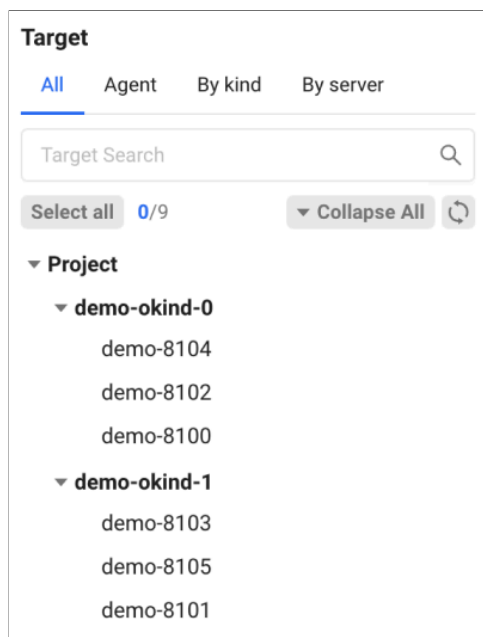
To add a metric as a widget to the dashboard, select **Add chart** on the upper left.

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.

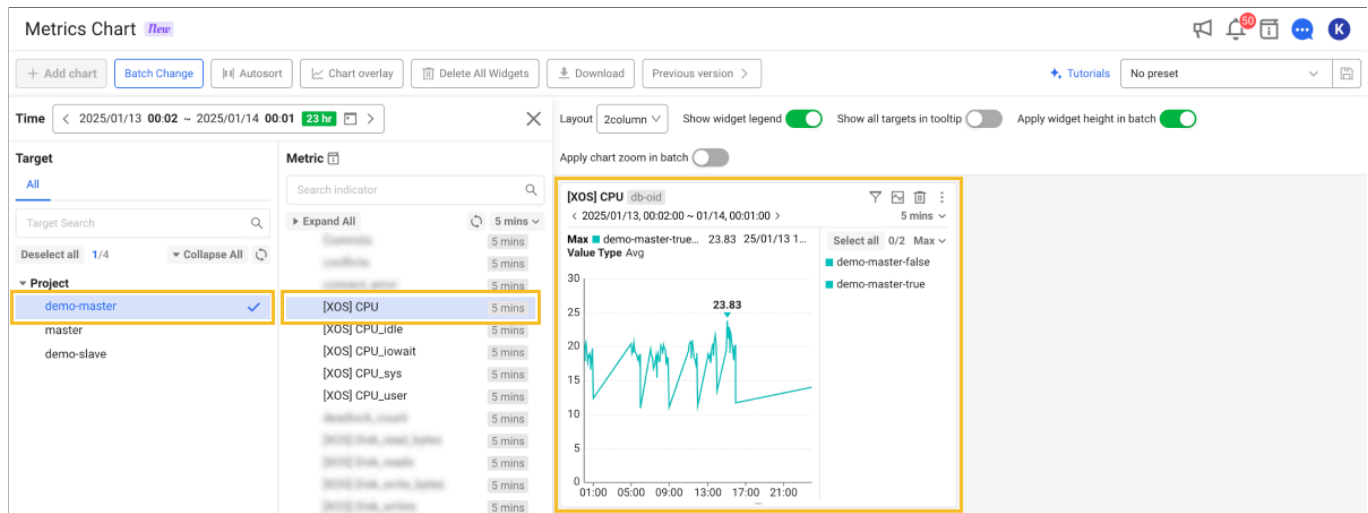


- **Agent:** You can select an individual agent or all agents.
- **By kind:** Agents can be monitored in each group classified by `whatap.okind` in the agent configuration.
- **By server:** Agents can be monitored in each group classified by `whatap.onode` in the agent configuration.

- ⓘ • 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

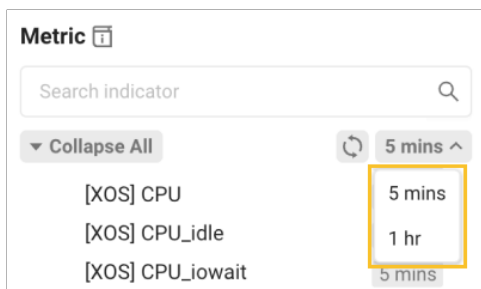
Select a target agent to monitor. Then the list that can be selected on the **Metric** panel appears. Select a desired metric in the list.



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**.

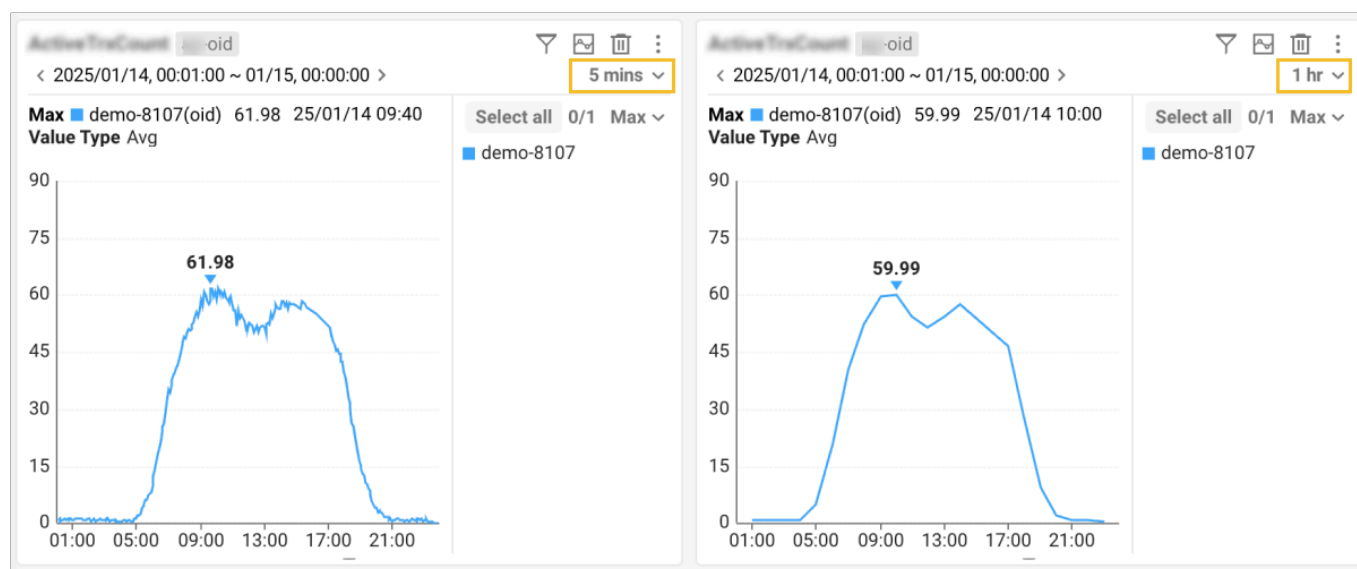
Setting the interval



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.

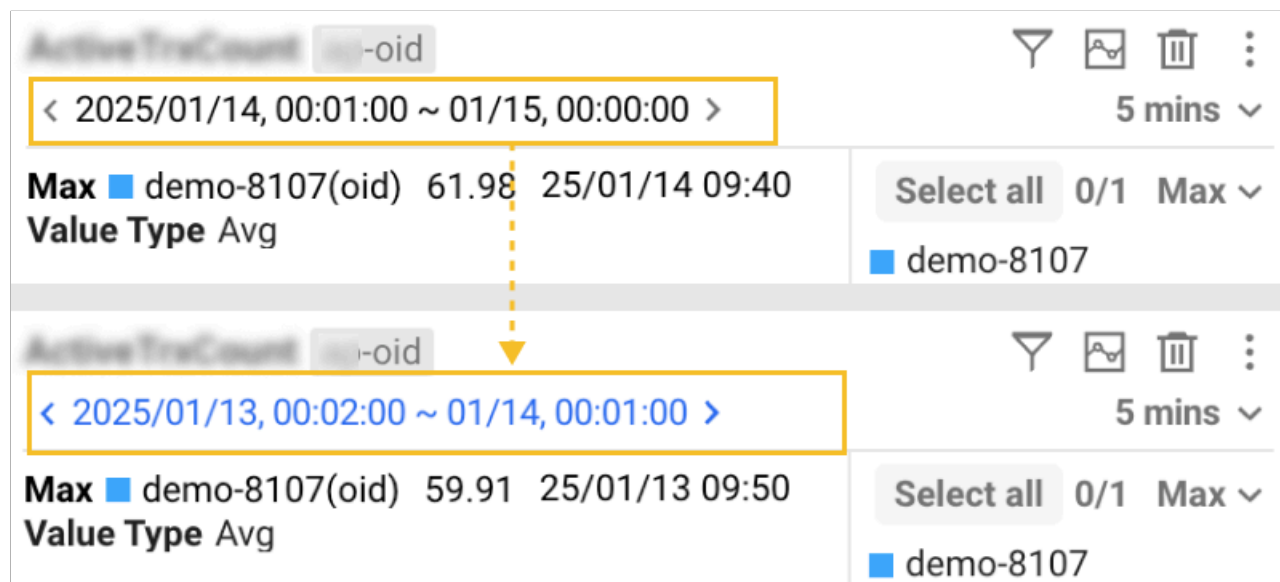
For example, the **5 min** interval displays average values for 5-minute intervals, and the **1 hour** interval displays average values for every one hour.



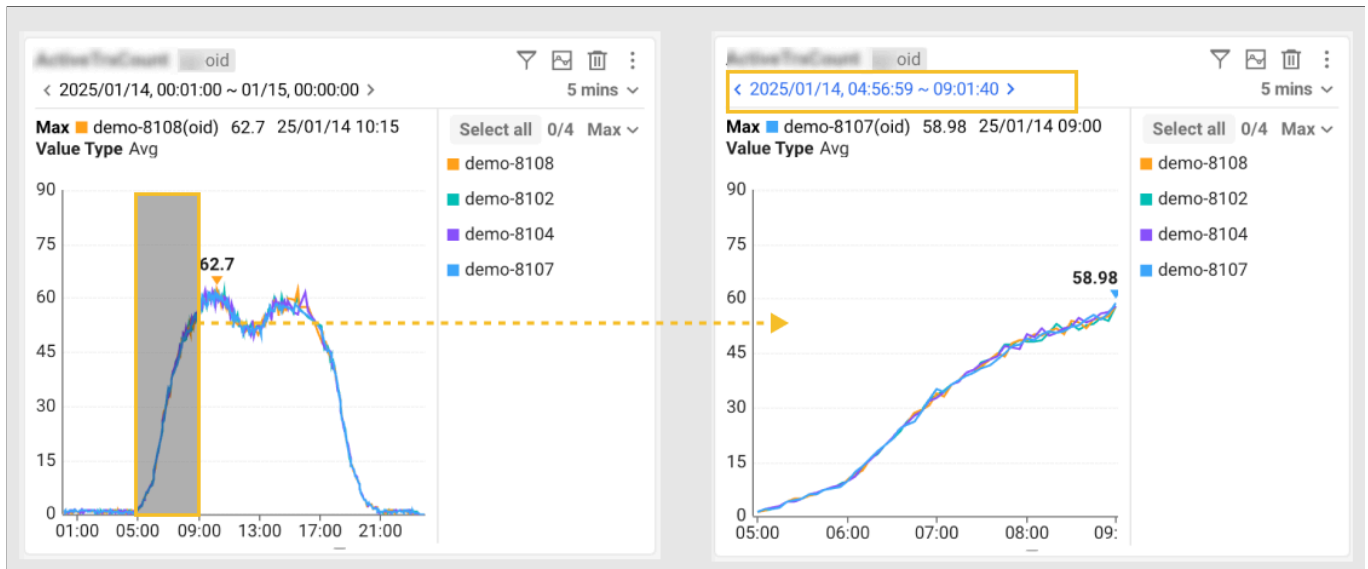
- ✓ By adjusting the interval option, you can analyze the data from different perspectives, from detailed data over a short time to overall trend over a long time.

Changing the query time

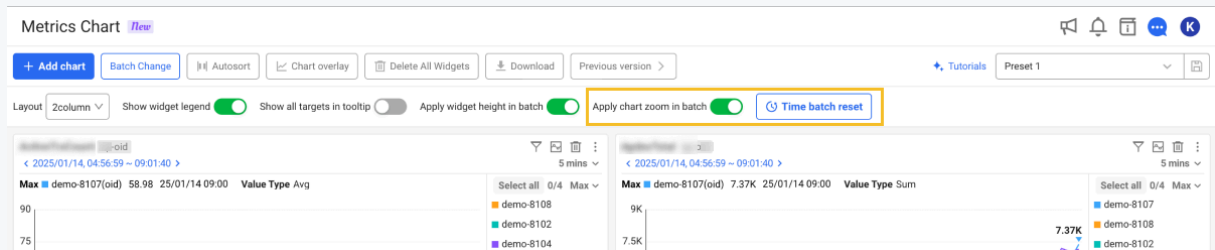
To change the chart query time range for an individual widget, select **<** or **>** at the top of the widget. The time change range is the same as the time reference set in the **Time** option. For example, if the **Time** option is set to **1 day** and you select **<** or **>**, you can change the 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.

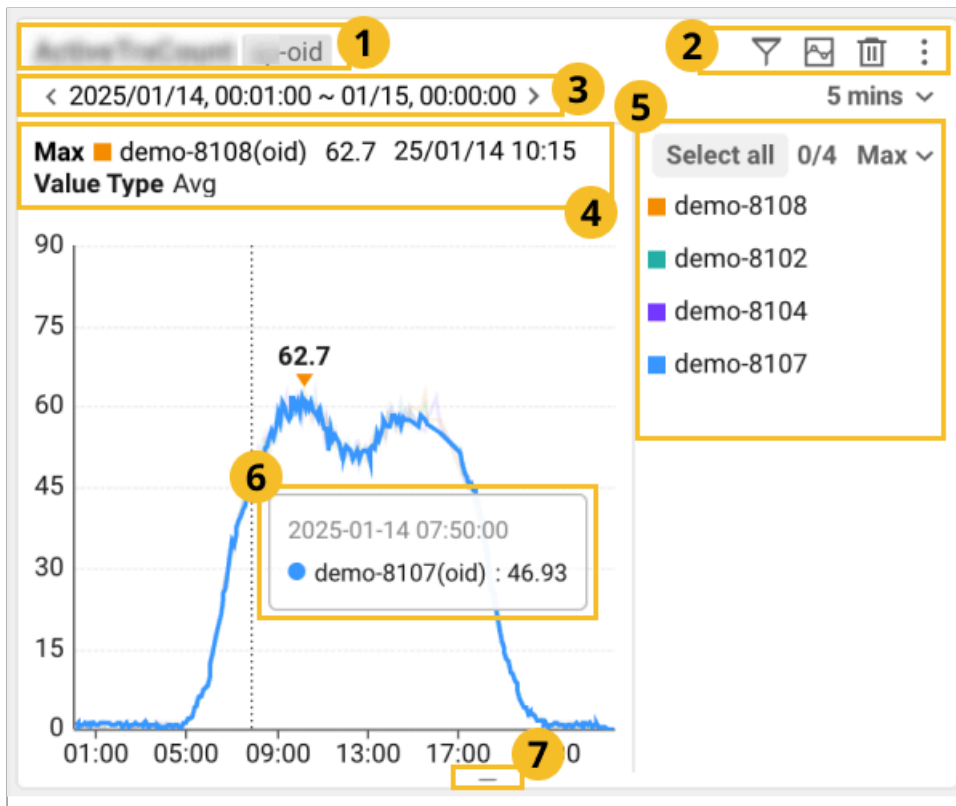


- To change the query time range and then change it back to a recent time, select .
- If the **Apply chart zoom in batch** option is enabled at the top of the screen, the  **Time batch reset** button appears. By selecting this button, you can reset the time for all widgets.

- ! 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.

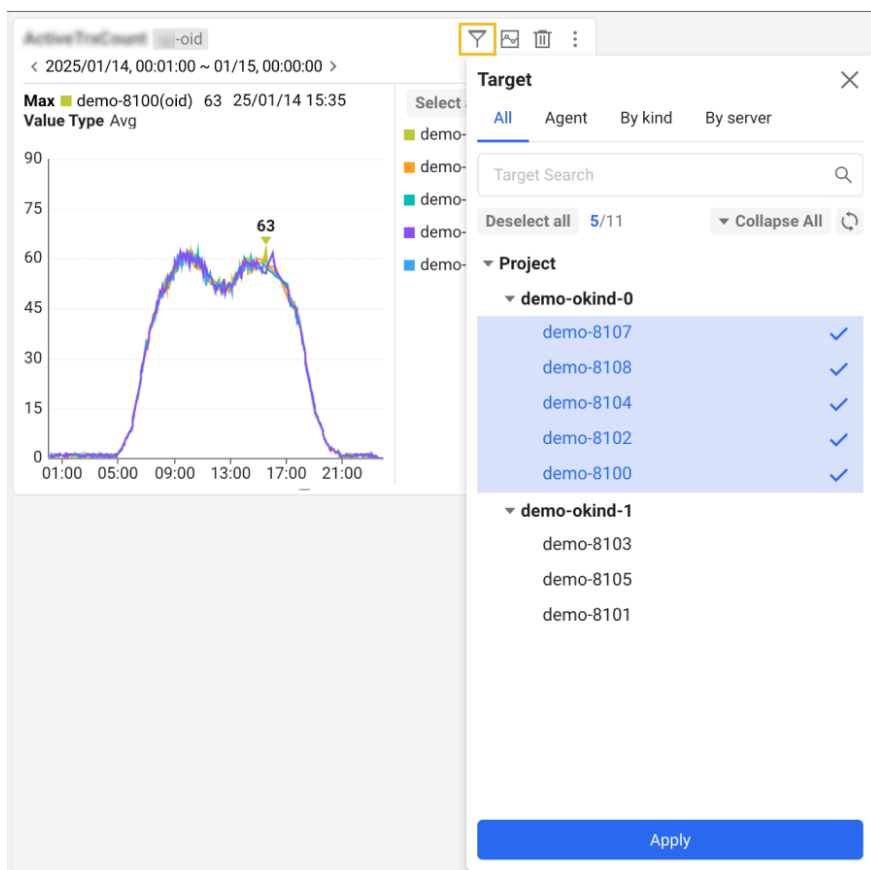
Using the chart widget options

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.



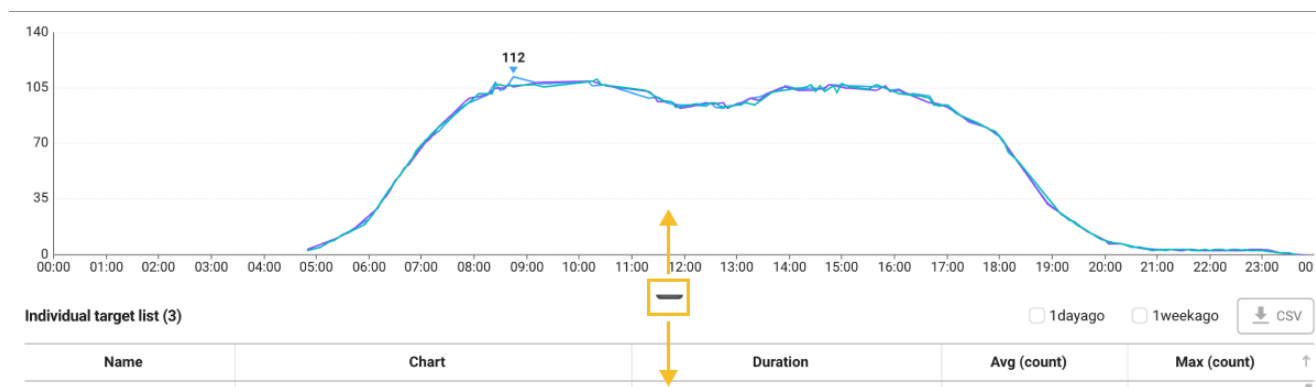
2. Select the desired targets according to the instructions on the screen.
3. Select **Apply**.

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.

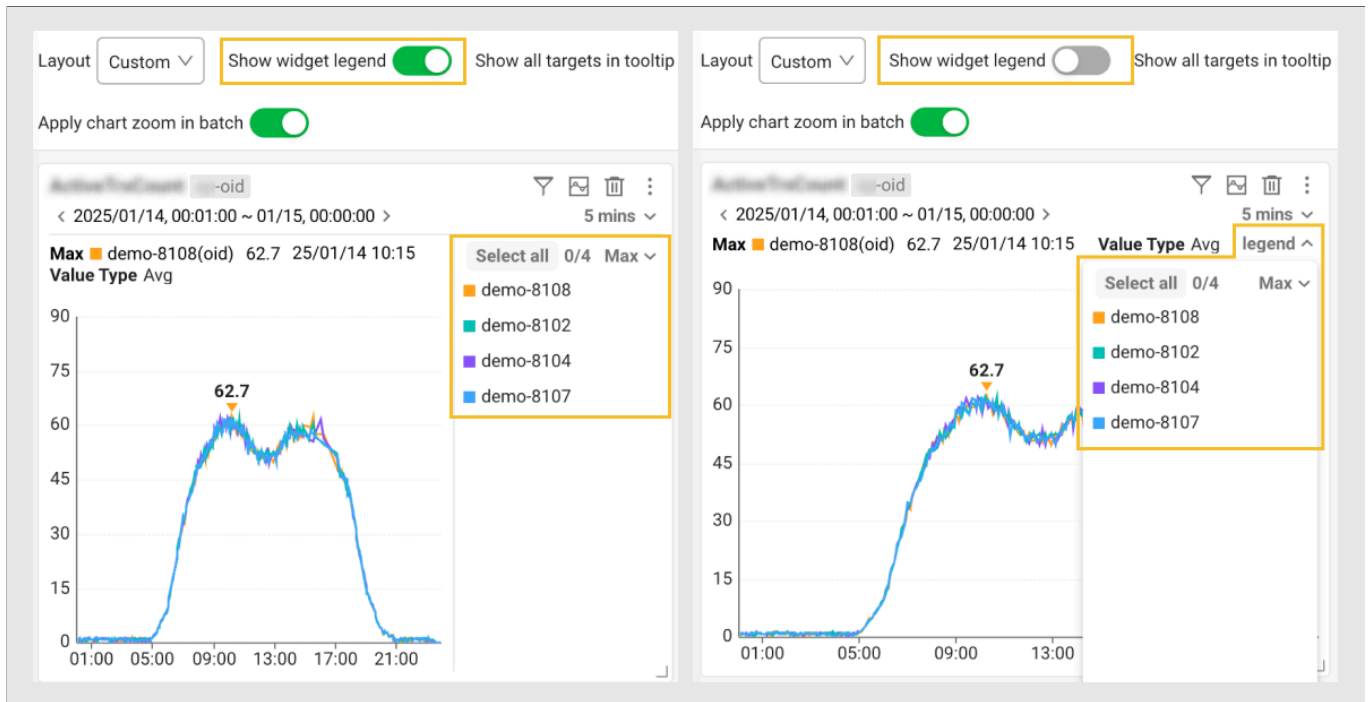


Removing the chart widget

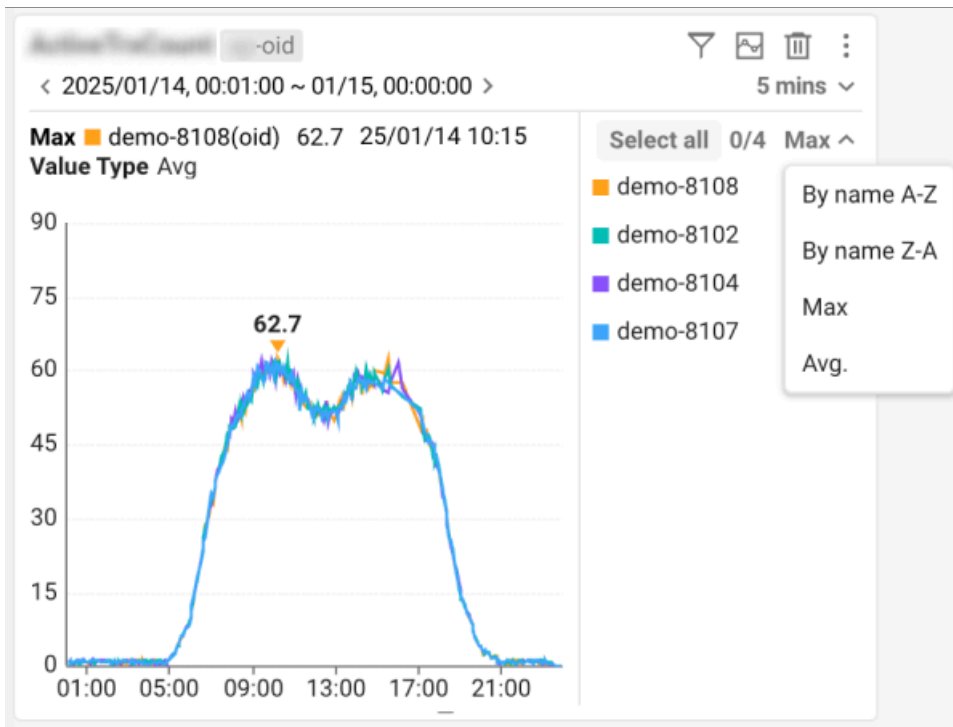
To remove any chart widgets placed on the screen, select .

Using the legend

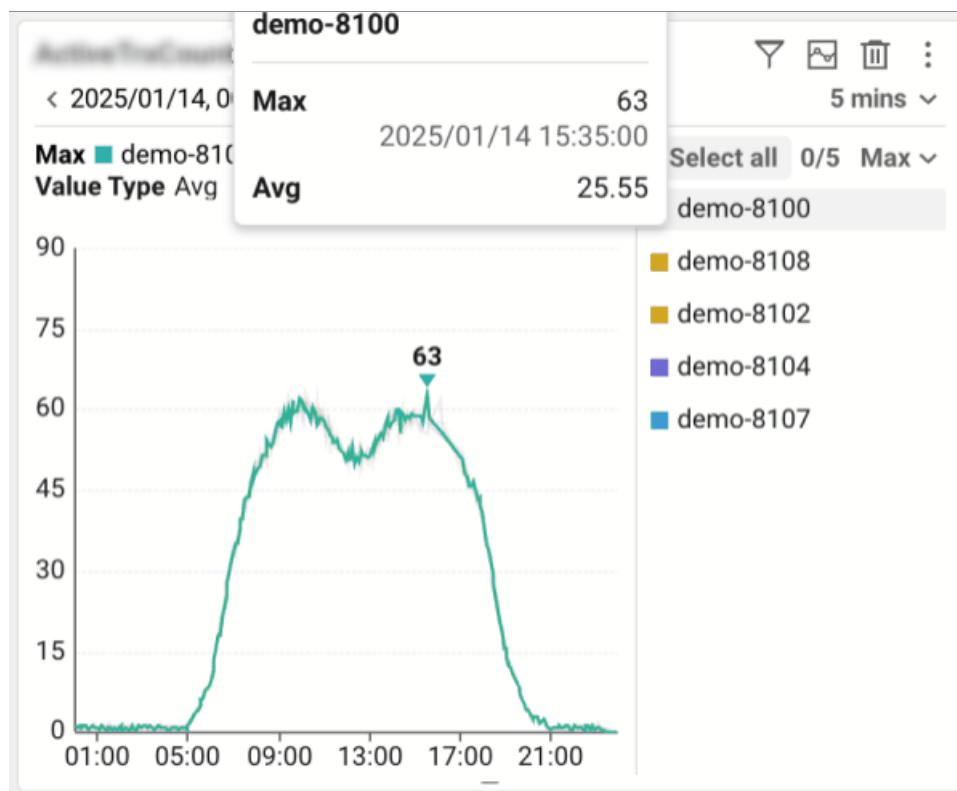
You can check the monitoring targets displayed on the chart through the legend and then sort or filter them by desired criteria. You can show or hide the legend on the right of the widget's chart by turning on or off the **Show widget legend** option at the top of the screen. With the option turned off, you can check and select the monitoring targets by clicking **legend**.



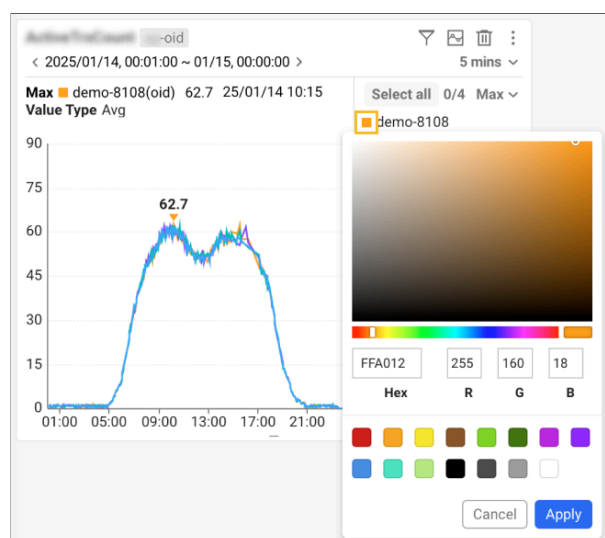
- You can select or unselect monitoring targets by selecting **Select all/Deselect all**.
- In the legend list, you can sort monitored objects by name, maximum value, or average value.



- 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



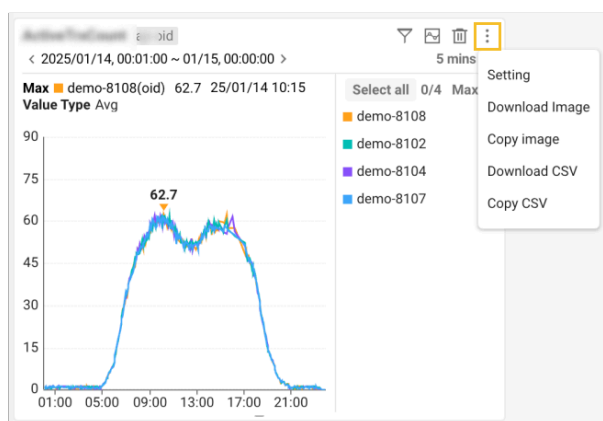
You can change the graph color corresponding to a monitoring target. In the legend list, select the square box

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.

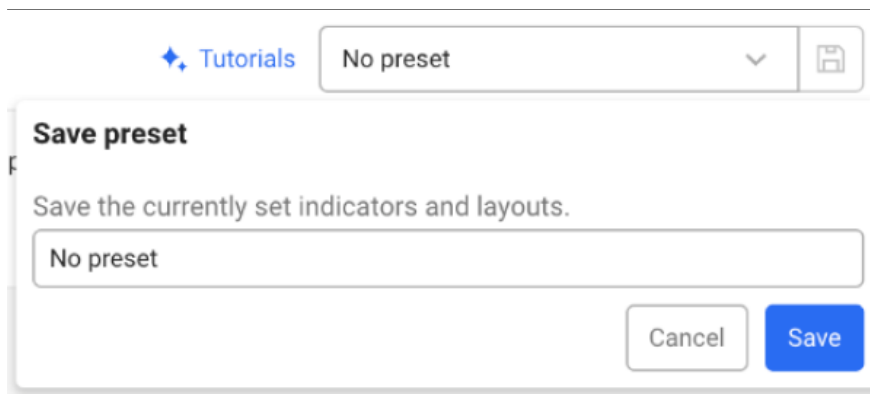
Setting presets

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](#).

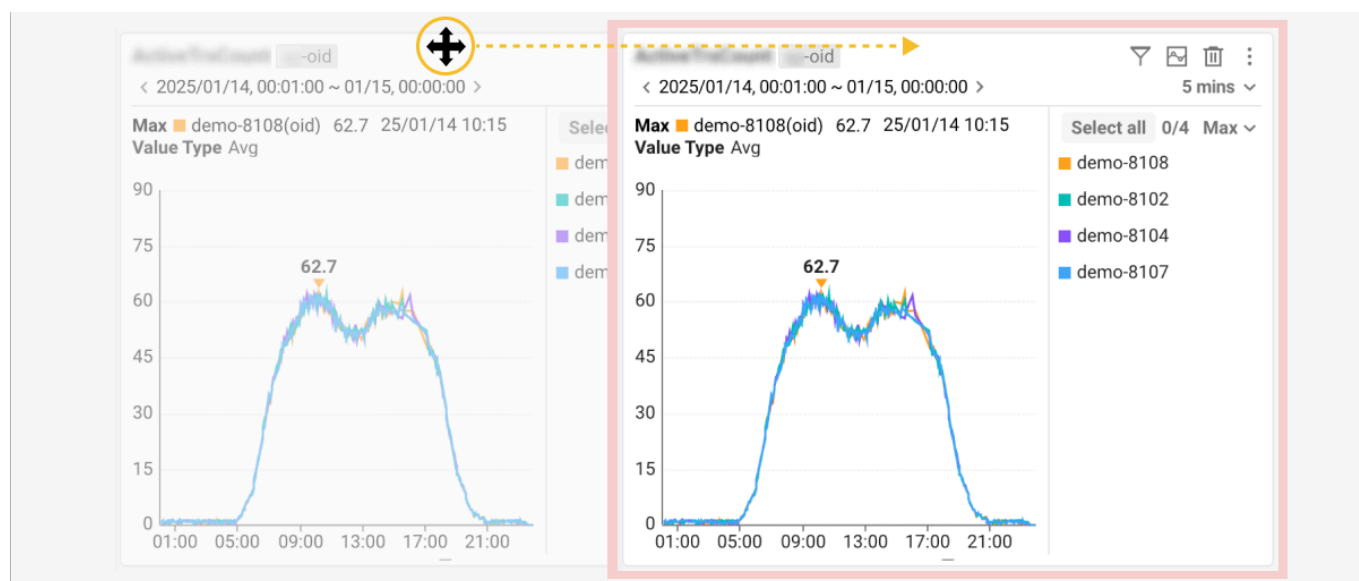
Deleting presets

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.

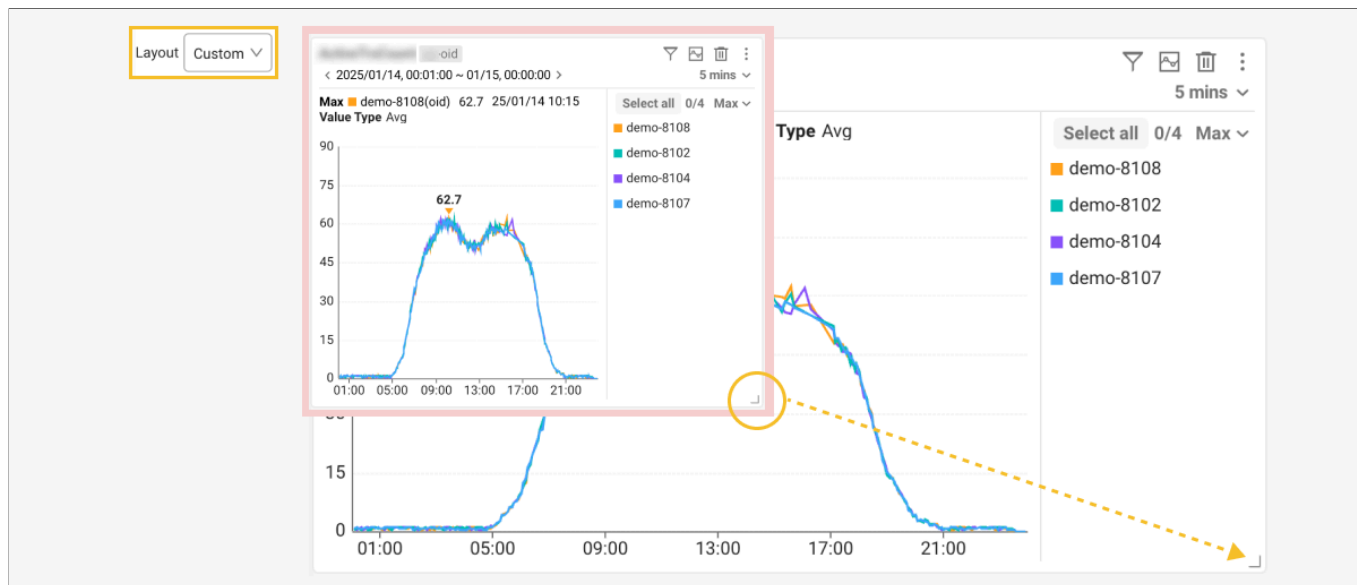
Moving a widget



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.

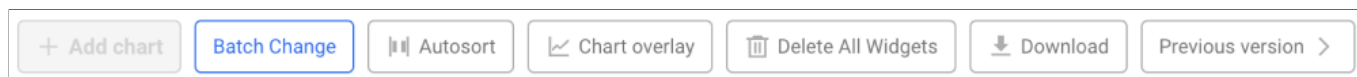


Click and hold the mouse on the  element at the lower right of the widget, and then drag it to the desired size.

ⓘ 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.

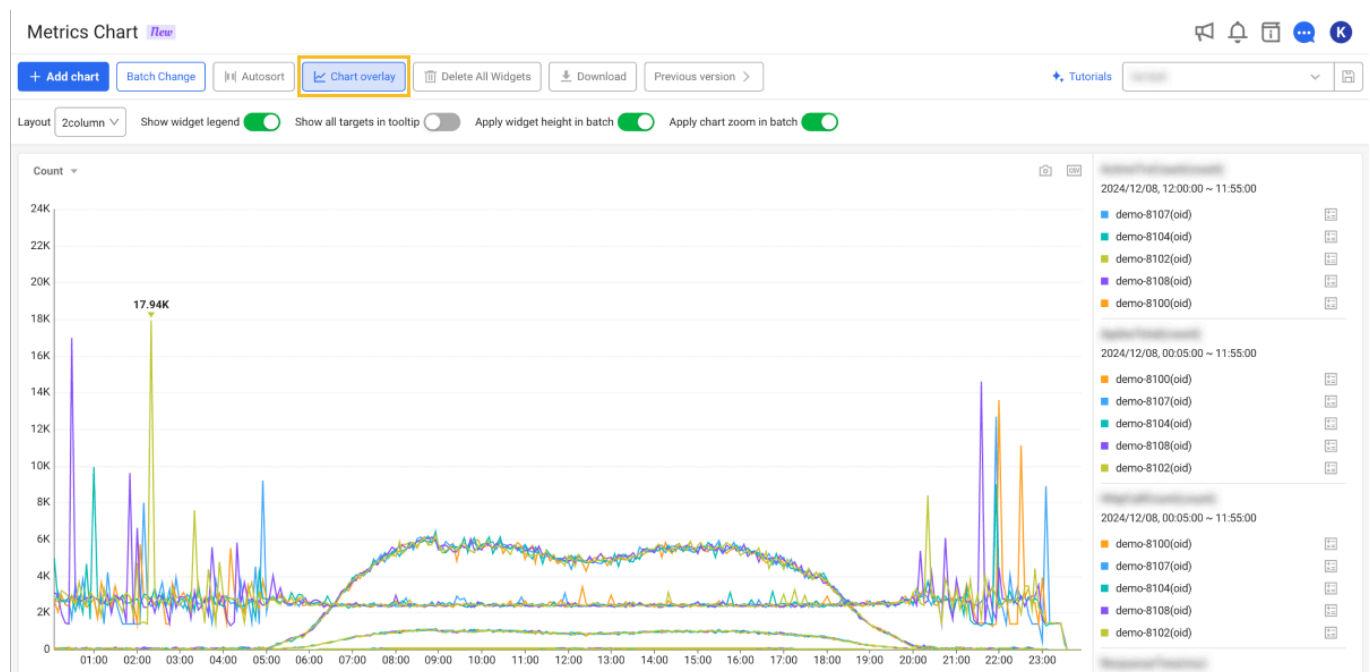
- **Apply widget:** Select a chart widget to apply the options to change. You can apply changed options to the selected widget at once.

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.

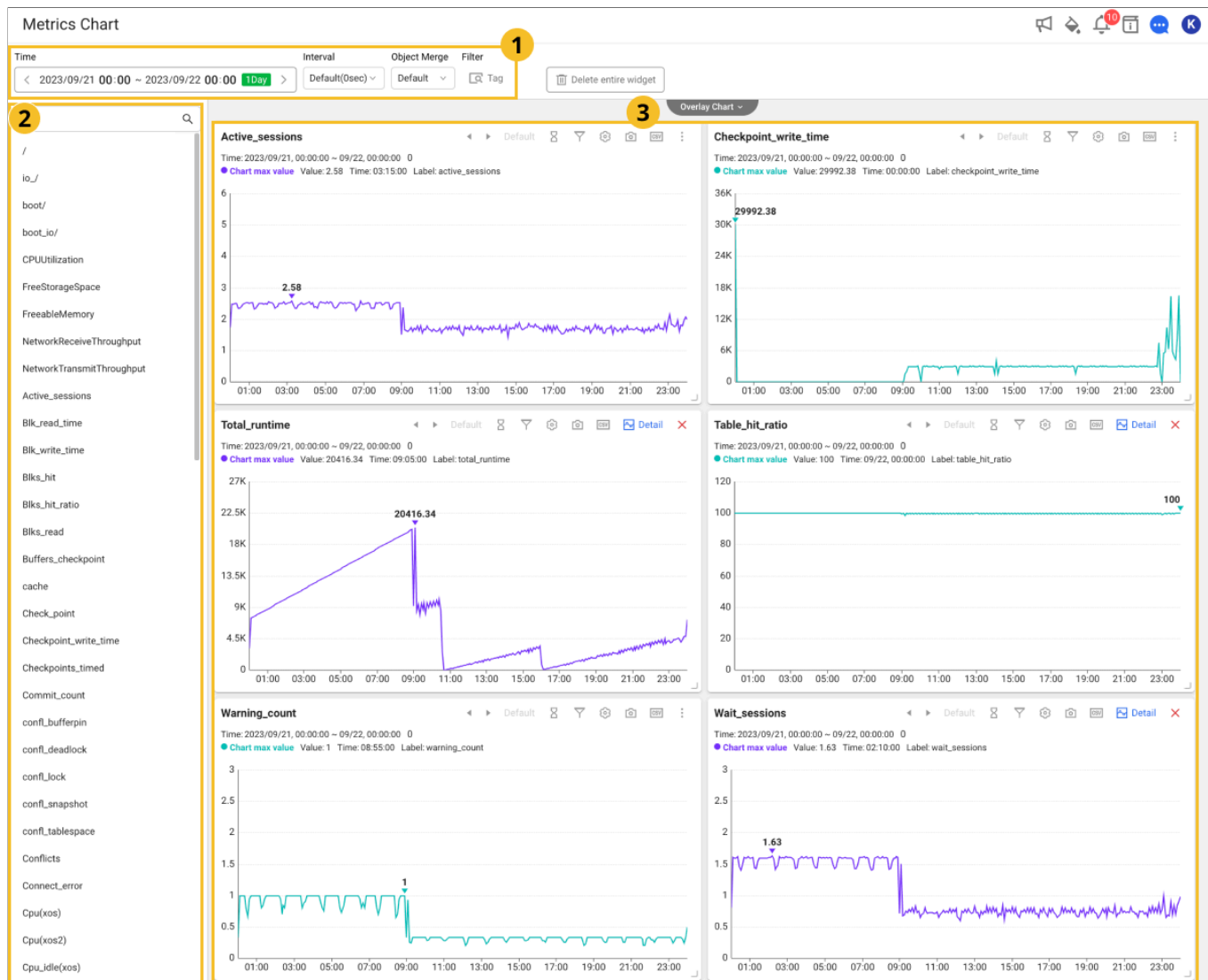
Using the tutorial

When you select **Tutorials** on the upper right of the screen, you can check the changes after the [Service 2.8.0](#) update.

Metrics 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. Search the metrics or select a metric from the list. Based on the data within the time range specified in the top menu of the **1** area, you can see the data via the chart widgets in the **3** area.

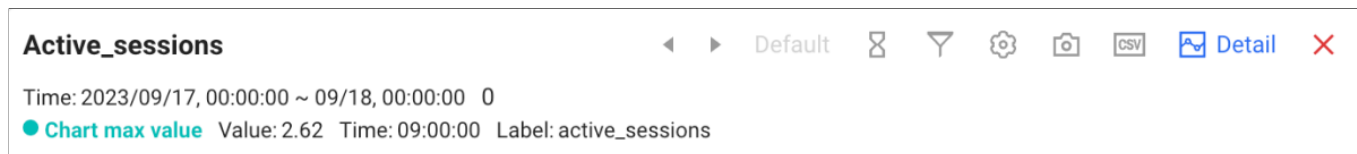
Data Merge

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**.

1 Metrics Search

Time selector: < 2023/08/17 07:57 ~ 2023/08/17 08:57 60Min > Max count: 100 Category: app_counter

2 Tags: All container containerKey host_ip okindName oname onodeName pid type 8/8 Filter: Filter by tag value

Field: All _clock_delta active_tx_0 active_tx_3 active_tx_8 active_tx_count apdex_satisfied apdex_tolerated apdex_total arrival_rate httpc_count httpc_error httpc_time meterlog resp90 resp95 resp_time sql_count sql_error sql_fetch_count sql_fetch_time sql_time tps tx_count tx_doc_time tx_error tx_http_time tx_sql_time tx_time tx_time_sql_sum 5/29

3

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

1 Time and category selection

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

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

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

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

2 Tag and field selection

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

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

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

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

3 Metrics table

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

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

Metric Anomaly Detection

Home > Select Project > **Analysis** > **Metrics 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.

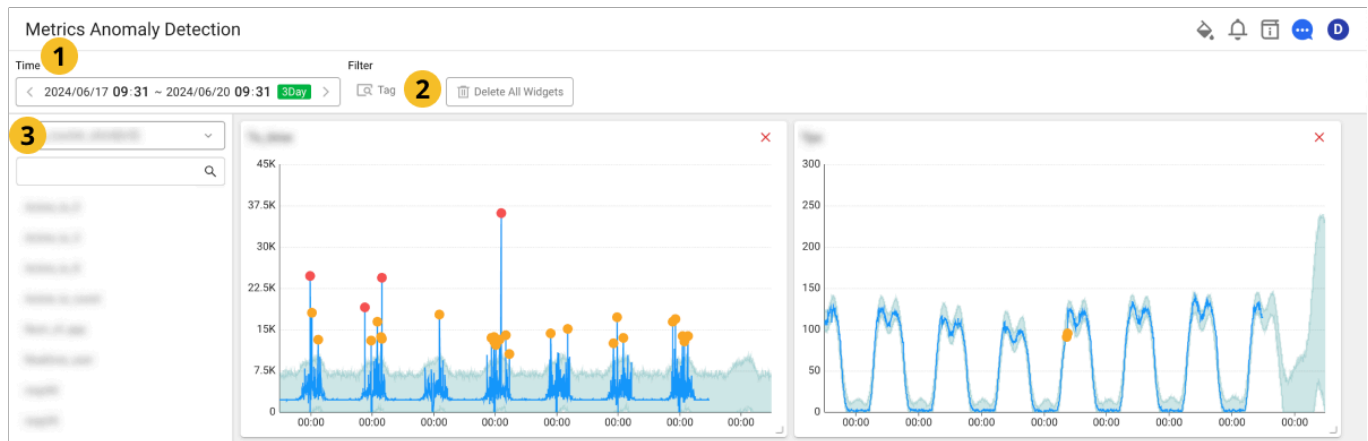
- ❗ The widgets placed on the screen are not saved and reset when you go to another menu.
- It is similar to **Analysis** > **Metrics Chart** except for the pattern and anomaly display.
- The alert, **Anomaly detection** is based on the **Metric anomaly detection**. For more information about the **Anomaly 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.

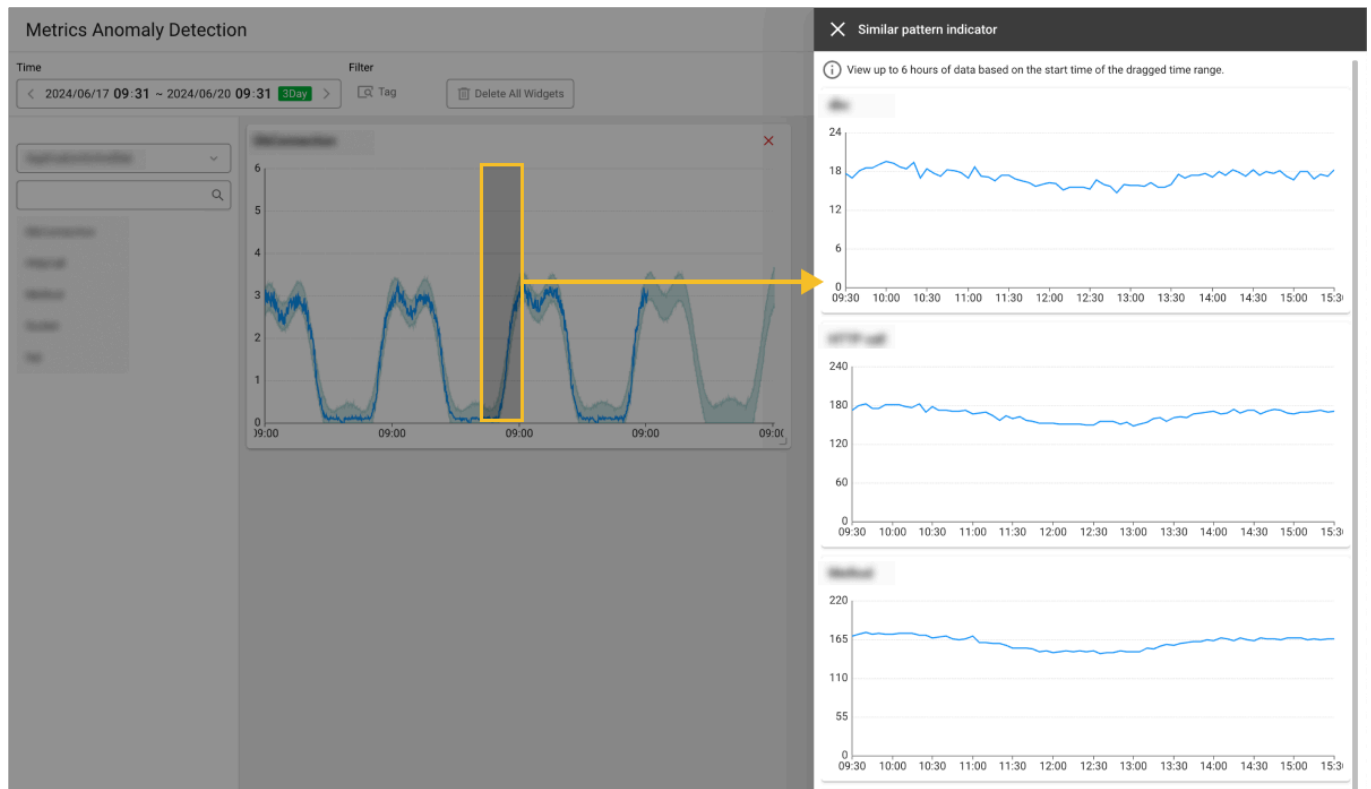
Place the selected metric to the right of the screen.

- 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.

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.

Top SQL

The **Top SQL** screen provides a list of Top SQL for the query period. This is SQL statistics self-calculated by WhaTap based on **SQL in Active Sessions** collected by the agent, including metrics such as execution count, execution time, and wait time.

- Queries can be made in the range of **minimum 1 minute to maximum 30 days**, making it suitable for long-term analysis.
- Trend charts allow you to visually identify SQL occurrence patterns in a specific time period.
- Statistics are calculated based on SQL excluding literal values.

✔ Differences from SQL Statistics (DB)

Category	Top SQL	SQL Statistics (DB)
Data Source	Active Sessions collected by the agent	pg_stat_statements view in DB
Statistics Calculation	Self-calculated by WhaTap	Collects statistics provided directly by DB
Separate DB Configuration	Not required	pg_stat_statements extension installation required
Key Metrics	Execution count, execution time (avg/max/sum), wait time	Execution count, execution time, row processing, block I/O, etc.
Query Range	1 minute to 30 days	Hourly/daily units
Trend Chart	Provided	Not provided

Prerequisites

Agent configuration is required to use the Top SQL analysis feature.

Agent configuration

5-minute interval statistics are provided by default without any additional configuration. For more precise 1-minute interval analysis, add the following setting to [whatap.conf](#).

```
sqlstat_send_interval=1
```

Option	Type	Default	Description
<code>sqlstat_send_interval</code>	int	5	Interval (minutes) for generating and sending SQL statistics data. Can be set to 1 or 5.

Analyzing Top SQL

The Top SQL screen is configured to analyze SQL performance from various perspectives by selecting an **analysis criteria tab** and **sort order**. Users can explore step by step from group analysis from an application perspective to individual SQL performance analysis using drill-down/drill-up.

Top SQL analysis proceeds in the following order:

1. Select the analysis criteria in the **analysis criteria tab**.
 - Analysis criteria types: ALL, DB, APPLICATION, USER, HOST
2. Select the top selection metric in **sort order**.
 - Top selection metric types: Elapsed Time, Wait Elapsed Time, etc.
3. Identify groups with high load in the chart.
4. Select a specific group and check individual Top SQL in the SQL List.
5. If further analysis is needed, navigate to the SQL detail analysis screen.

Basic options

In the top filter area, you can set the time, target instance, filters, and more.

- **Time:** Select the query time range (1 minute to 30 days)
- **Instance:** Select the target DB instance
- **Filter:** Condition-based search by field (default: exclude 'WhaTap' from `SQL Text`)
- **Sort order:** Select the criterion metric (Elapsed Time, Wait Elapsed Time, Avg Elapsed Time, Max Elapsed Time, Execute Count) for Top SQL selection, sorted in descending order by the selected metric
- **Query count:** Set the number of items to display in the table
- **Top N:** Set the number of top items to display in the trend chart
- **Column selection:** Add/remove columns to display and change order by drag
- **Download:** Export in CSV format

Field panel

In the field panel on the left side of the screen, you can easily add filter conditions. Click a field name to enter a condition, and that condition is added to the filter.

Field	Type	Description
SQL Text	String	SQL statement
Elapsed Time	Number	Total execution time (seconds)
Wait Elapsed Time	Number	Total wait time (seconds)
Avg Elapsed Time	Number	Average execution time (seconds)
Max Elapsed Time	Number	Maximum execution time (seconds)
Execute Count	Number	Execution count
WSQL Hash	String	SQL hash value

Field	Type	Description
User	String	Executing user
Datname	String	Database name
Application Name	String	Application name
Client Hostname	String	Client hostname

Analysis criteria tab

In the analysis criteria tab, select how to analyze Top SQL. Various analysis criteria such as ALL, DB, APPLICATION, USER, and HOST are provided. Depending on the selected criteria, SQL execution information can be analyzed in group units or SQL units.

ALL

Selecting ALL allows you to directly view Top SQL by SQL unit. The chart is also displayed by SQL unit, and is used to directly check SQL with high performance impact across the entire system.

DB/APPLICATION/USER/HOST

DB / APPLICATION / USER / HOST are frequently used analysis criteria from an application perspective. Selecting these tabs aggregates SQL execution metrics by the corresponding criteria and displays them in the chart.

Tab	Description
DB	Analyze SQL execution load by database unit.
APPLICATION	Analyze SQL load by application name.
USER	Analyze SQL load by user account.
HOST	Analyze SQL load by client host.

Selecting a specific group in the chart displays the **SQL list belonging to that group** (SQL List) at the bottom. This allows you to drill down from group → SQL to analyze individual Top SQL in detail.

Trend chart

Displays the trend of Top N SQL based on the selected metric tab as a time-series chart. You can see at a glance which SQL has high metric values and in which time periods. Each SQL can be checked in the legend at the bottom of the chart, and clicking the legend or chart highlights the corresponding SQL line.

The data interval of the chart is automatically adjusted according to the query time range.

Query Range	Chart Interval
3 hours or less	1 minute or 5 minutes (determined by <code>sqlstat_send_interval</code> value)
1 day or less	5 minutes
30 days or less	1 hour

Per-query chart

Clicking the ► button for a table item displays the **execution count** and **average execution time** charts for that individual query. The chart interval is the same as above.

- Hover the mouse to check values by time.
- Clicking the bar graph in the chart opens the **Instance Monitoring** screen for that time in a new window. You can check the overall instance and active sessions during the time when that SQL was executed.

Data interpretation guide

Key metrics

Column	Meaning	Interpretation Point
sql	SQL statement (excluding literal values)	Check what query it is
instance	Name of the instance where executed	Check which DB it was executed on
execute count	Execution count	The higher, the more frequently called SQL → Greater optimization effect
elapsed sum (sec)	Total accumulated execution time (seconds)	The higher, the greater the overall load SQL
wait elapsed sum (sec)	Total accumulated wait time (seconds)	The higher, the longer the wait SQL → Check for lock, I/O bottleneck, etc.
elapsed avg (sec)	Average execution time (seconds) = $\text{elapsed sum} / \text{execute count}$	The higher, the slower individual executions → Primary tuning target
elapsed max (sec)	Maximum execution time (seconds)	Detect intermittent performance degradation → Understand worst-case scenario

Identifying SQL to tune

- SQL with high **elapsed sum** and high **execute count**
Frequently executed with high overall load → Top priority tuning target
- SQL with high **elapsed avg**
The individual query itself is slow → SQL tuning, index check
- SQL with low **elapsed avg** but high **execute count**

Individual is fast but called too frequently → Caching, call logic improvement

- **SQL with high** `wait elapse sum`

Long wait time making overall execution slow → Check for lock contention, I/O bottleneck, etc.

- **SQL with large difference between** `elapse max` **and** `elapse avg`

Usually fast but suddenly slows down under specific conditions → Check for lock contention, data skew, etc.

Practical usage scenarios

Scenario 1. Finding SQL with high overall load

1. Sort in **descending order** by `elapse sum (sec)`
2. Top SQL has the greatest impact on overall DB load
3. Check `execute count` and `elapse avg` together to make a judgment
 - **If `elapse avg` is high:** The individual query itself is slow → SQL tuning, index check
 - **If `elapse avg` is low but `execute count` is high:** Called too frequently → Caching, call logic improvement

Scenario 2. Finding SQL with long wait time

1. Sort in **descending order** by `wait elapse sum (sec)`
2. Identify SQL with long wait times
3. Analyze causes such as lock contention, I/O bottleneck, connection wait

Scenario 3. Finding slow SQL

1. Sort in **descending order** by `elapse avg (sec)`
2. Identify SQL with slow individual executions
3. Check the **Plan** (execution plan) for the SQL to determine tuning points

Scenario 4. Detecting intermittent performance degradation

1. Check SQL with a large difference between `elapsed max (sec)` and `elapsed avg (sec)` → SQL that is usually fast but suddenly slows down under specific conditions
2. Open the per-query trend chart with the ► button to check the **time when it slowed down**
3. Click the bar graph for that time to analyze the cause in **Instance Monitoring**

Scenario 5. Analyzing anomalies in a specific time period

1. Check **abnormally high intervals** in the trend chart
2. Click on that time period to check instance monitoring metrics
3. Analyze which SQL was executed intensively during that time period

Additional features

You can check SQL details and set condition-based filters.

SQL details

Click the SQL column to open the details window. Refer to the following for descriptions of the SQL details window.

Adding filters

1. Click the add filter button.
2. Select the filter key.
 - **String:** `=` (contains), `!=` (does not contain)
 - **Number:** `==` (equals), `>=` (greater than or equal to), `<=` (less than or equal to)
3. Enter a value and apply.
4. Apply **AND logic** for multiple conditions.

PG SQL Statistics

ⓘ PG SQL Statistics will be renamed to **SQL Statistics (DB)** in the future.

This is **DB's own SQL statistics** provided by PostgreSQL's `pg_stat_statements` view. You can check delta values of execution count, execution time, number of rows processed per SQL, block I/O, etc. on the screen without querying the DB directly.

- The agent calculates and collects delta values at 1-hour intervals.
- Collects up to `statements_row_limit` based on the `statements_order_by` agent option (default `total_time`). (Default: 5,000 records)
- **DBX agent version 1.6.10 or later** is required.

ⓘ Data collection begins **approximately 1 hour after** configuration is complete.

Prerequisites

Both **DB configuration** and **Agent configuration** are required to display data on the screen.

DB configuration

1. Install the `pg_stat_statements` extension

`pg_stat_statements` is included in PostgreSQL's contrib module. Install the package if it is not already installed.

```
# Red Hat / CentOS
yum install postgresql-contrib

# Debian / Ubuntu
apt-get install postgresql-contrib
```

2. Configure postgresql.conf

Add `pg_stat_statements` to `shared_preload_libraries` . PostgreSQL restart is required after changes.

```
shared_preload_libraries = 'pg_stat_statements'
```

3. Create the extension module

Create the extension in the target monitoring database.

```
CREATE EXTENSION pg_stat_statements;
```

4. Verify installation

```
SELECT * FROM pg_stat_statements LIMIT 1;
```

Agent configuration

In PostgreSQL, the default value of the `statements` option is `true` , so **SQL statistics are collected by default without any additional configuration**. To disable collection, set `statements=false` .

Option	Type	Default	Description
<code>statements</code>	boolean	<code>true</code>	Configure whether to collect SQL statistics.

Advanced options

Option	Type	Default	Description
<code>statements_interval</code>	int	1	SQL statistics collection interval (in hours). If set to <code>statements_interval=1</code> , collection occurs every 1 hour.
<code>statements_interval_min</code>	int	0	SQL statistics collection interval (in minutes). If set to <code>statements_interval_min=10</code> , collection occurs



Option	Type	Default	Description
			every 10 minutes. This option takes priority over <code>statements_interval</code> when set.
<code>statements_min_row</code>	int	10000	Threshold for collecting SQL statistics data. See collection criteria below.
<code>statements_row_limit</code>	int	5000	Maximum number of records to collect. Collects up to the top count based on <code>statements_order_by</code> .
<code>statements_order_by</code>	String	"total_time"	Specifies the sort criteria when collecting SQL statistics. (e.g., <code>statements_order_by=total_time,temp_blks_read</code>)
<code>statements_schema</code>	String	""	If the <code>pg_stat_statements</code> extension is not installed on the DB specified in the <code>db</code> option of whatap.conf , specify the DB name where the extension is installed.

Collection criteria (`statements_min_row`)

Collects the SQL data if the `rows` value in `pg_stat_statements` (total number of rows retrieved or affected by the query) **exceeds** `statements_min_row`.

⚠ If the amount of SQL collected is too much or too little depending on the DB workload, you can control the collection volume by adjusting the `statements_min_row` and `statements_row_limit` values.

Analyzing SQL Statistics (DB)

Basic options

In the top filter area, you can set the query period, target instance, filter conditions, and more.

- **Time:** Select the date and time to query
- **Instance:** Select the target DB instance
- **Filter:** Filter data by condition (AND logic applied for multiple conditions)
- **Sort order:** Select the column to sort data by (e.g., `rows` , `total_time` , `calls` , etc.)
- **Query count:** Set the number of items to display in the table
- **Search in results:** Search by query keyword within the retrieved results
-  **(Column selection):** Add/remove columns to display, change order by drag
-  **(Download):** Export in CSV format

Column guide

Basic information

Column	Description
<code>query</code>	Normalized SQL statement (literal values replaced with <code>\$1</code> , <code>\$2</code> , etc.)
<code>queryid</code>	Hash code identifying the same normalized query
<code>username</code>	Name of the user who executed the query
<code>db</code>	Name of the database where the query was executed
<code>instance</code>	DB instance name where the SQL was executed

Execution and time

Column	Description
<code>calls</code>	SQL execution count
<code>total_time</code>	Total execution time (unit: milliseconds)
<code>rows</code>	Total number of rows retrieved or affected by the query

Block I/O (shared buffer)

Column	Description
<code>shared_blks_hit</code>	Number of blocks read from shared buffer cache (cache hit)
<code>shared_blks_read</code>	Number of shared blocks read from disk (cache miss)
<code>shared_blks_dirtied</code>	Number of modified shared blocks
<code>shared_blks_written</code>	Number of shared blocks written to disk

Block I/O (local buffer)

Column	Description
<code>local_blks_hit</code>	Number of blocks read from local buffer cache
<code>local_blks_read</code>	Number of local blocks read from disk
<code>local_blks_dirtied</code>	Number of modified local blocks
<code>local_blks_written</code>	Number of local blocks written to disk

Temporary blocks

Column	Description
<code>temp_blks_read</code>	Number of temporary blocks read
<code>temp_blks_written</code>	Number of temporary blocks written

I/O time

Column	Description
<code>blk_read_time</code>	Time spent reading blocks (unit: milliseconds). Requires <code>track_io_timing</code> to be enabled.
<code>blk_write_time</code>	Time spent writing blocks (unit: milliseconds). Requires <code>track_io_timing</code> to be enabled.

Table fetch

Column	Description
<code>table fetch by rowid</code>	Number of times table rows were directly fetched via index
<code>table fetch continued row</code>	Number of times additional blocks were read when a row spans multiple blocks

Data interpretation guide

Identifying SQL to tune

- SQL with high `total_time` and high `calls`
Frequently executed and slow → Top priority tuning target
- SQL with high `rows`

Processing a large number of rows → Check whether the query is unnecessary

- SQL with high `total_time / calls` (average execution time)

Individual executions are slow → SQL tuning, execution plan check

Cache efficiency check

- **Cache hit rate** = $\text{shared_blks_hit} / (\text{shared_blks_hit} + \text{shared_blks_read})$
 - If below 90%, consider increasing `shared_buffers` or checking inefficient queries
- SQL with high `shared_blks_read` : Causing a lot of disk I/O → Index check, consider increasing `shared_buffers`
- SQL with high `shared_blks_dirtied` : Modifying a lot of data → May cause checkpoint load

Temporary block check

- If `temp_blks_read` and `temp_blks_written` are high, memory is insufficient during sorting or hash joins, causing disk usage
- Consider increasing `work_mem`

I/O time check

- SQL with high `blk_read_time` : Large disk read wait → Check storage performance or cache efficiency
- `track_io_timing = on` must be set in `postgresql.conf` to check `blk_read_time`

Practical usage scenarios

Scenario 1. Finding slow SQL

1. Sort in **descending order** by `total_time`
2. Check `calls` of top SQL → SQL that is frequently executed and slow is the top priority tuning target

Scenario 2. Finding SQL with low cache efficiency

1. Check SQL with high `shared_blks_read`
2. Compare with `shared_blks_hit` to calculate cache hit rate
3. If cache hit rate is low, check the **Plan** (execution plan) for the SQL to decide whether to add an index

Scenario 3. Finding SQL with excessive disk I/O

1. Check SQL with high `blk_read_time` (requires `track_io_timing = on`)
2. → SQL spending a lot of time on disk reads → Index check, remove unnecessary sequential scans

Additional features

SQL details

Click the SQL column to open the details window. Refer to the following for descriptions of the SQL details window.

DB reference

`pg_stat_statements`

Since the data on this screen is retrieved from the DB's `pg_stat_statements` view, **changes to DB settings or statistics reset directly affect the collected data.**

`pg_stat_statements.max`

This parameter determines **how many different SQL patterns (queryid) can be stored**. The default value is `5000` .

```
-- Example: The following two SQLs are normalized to the same queryid.
SELECT * FROM users WHERE id = 10;
SELECT * FROM users WHERE id = 20;
→ SELECT * FROM users WHERE id = $1
```

When the number of storable SQL patterns exceeds the limit, SQL with the fewest executions is removed first. In this case, statistics for some SQL may be missing.

```
-- Check current limit
SHOW pg_stat_statements.max;

-- Check current number of stored SQL patterns
SELECT COUNT(*) FROM pg_stat_statements;
```

If the number of SQL patterns is approaching the limit, increase the `pg_stat_statements.max` value. PostgreSQL restart is required after changes.

`pg_stat_statements_reset()` initialization

Resetting statistics with the `pg_stat_statements_reset()` function **deletes all** existing data and records new data. Note that the delta values collected by the agent are also reset.

```
SELECT pg_stat_statements_reset();
```

Database Size

Home > Select Project > **Stat/Report** > **DB size**

It aggregates the sizes of tables for each database and provides trend data in a chart. You can identify and manage the trend in database capacity growth. You can prepare for rapid capacity increases by identifying them in advance.

Database size can affect the performance. You can prepare to troubleshoot performance issues because the increase in size can delay the query execution. It is also required to identify and prevent problems in advance that may increase the size due to incorrect queries or database structure issues.

The trend in database size helps you understand usage patterns and data growth. You can predict future capacity requirements and take possible actions. Monitoring these changes helps you optimize the resources and costs.

Before use

Assigning the query role

The total database size is queried with the `pg_database_size` function, and the monitoring account must have the connect or `pg_read_all_stats` role.

Individual table sizes can be queried with the `pg_relation_size`, `pg_indexes_size`, and `pg_total_relation_size` functions, and the query role for the corresponding schema is required.

```
grant usage on schema {schema_name} to whatap;  
grant select on all tables in schema {schema_name} to whatap;
```

- ⓘ • If the `skip_schema` and `skip_db` options are set in the agent, the database size is not collected.
- The collection cycle is collected once every hour.

Setting the agent options

You can set the following agent options to collect the database sizes.

- **tables** Boolean

Default `true`

It collects the table sizes. In **Stat/Report > DB size**, you can see the table size by selecting the database.

- **tables_hour** Int

Default `5`

Set the time when data is to be collected to store the table sizes. The default value is `5` and collection starts at 5 AM.

- **tables_min_size** Int

Default `10`

When collecting the table information, table data of the size smaller than the set value is not collected. The minimum unit for collecting table data is 10 MB based on `pg_total_relation_size`.

- **skip_schema** String

Default: `pg_catalog, sys, information_schema, pg_toast`

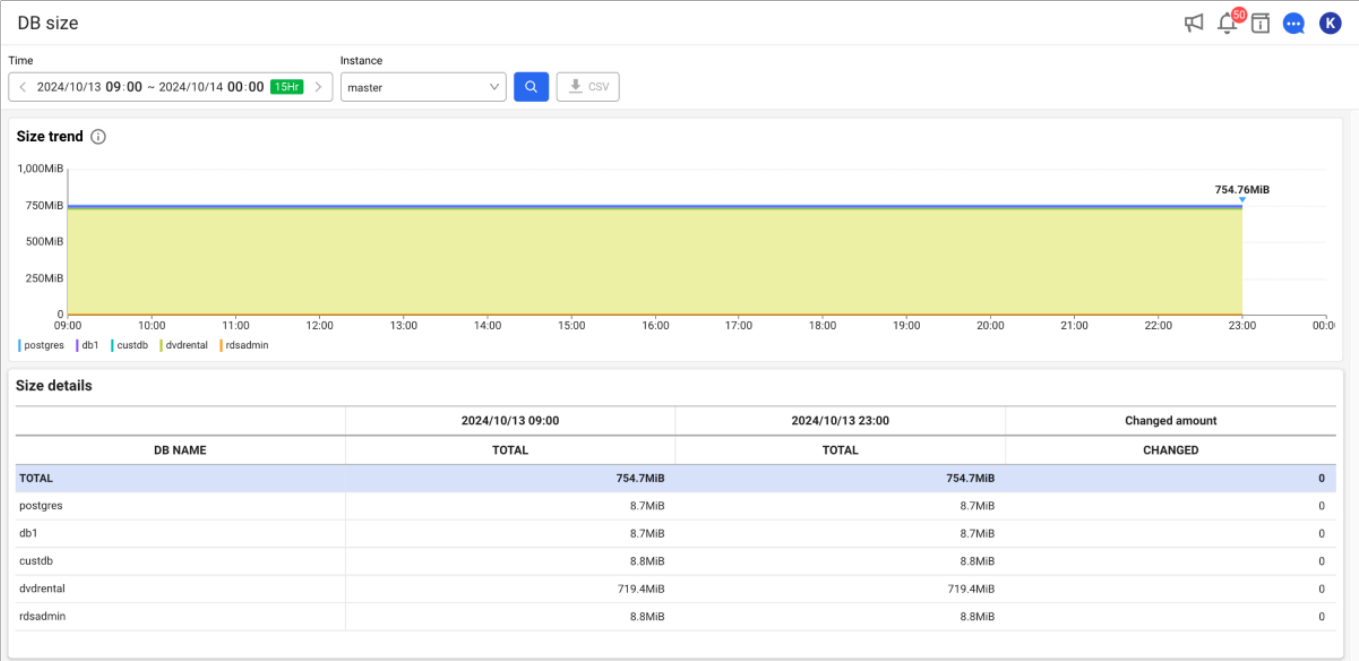
It does not collect tables for the schema set with the option value.

- **skip_db** String

Default: `edb, postgres`

It does not collect tables for the database set with the option value.

Basic screen guide



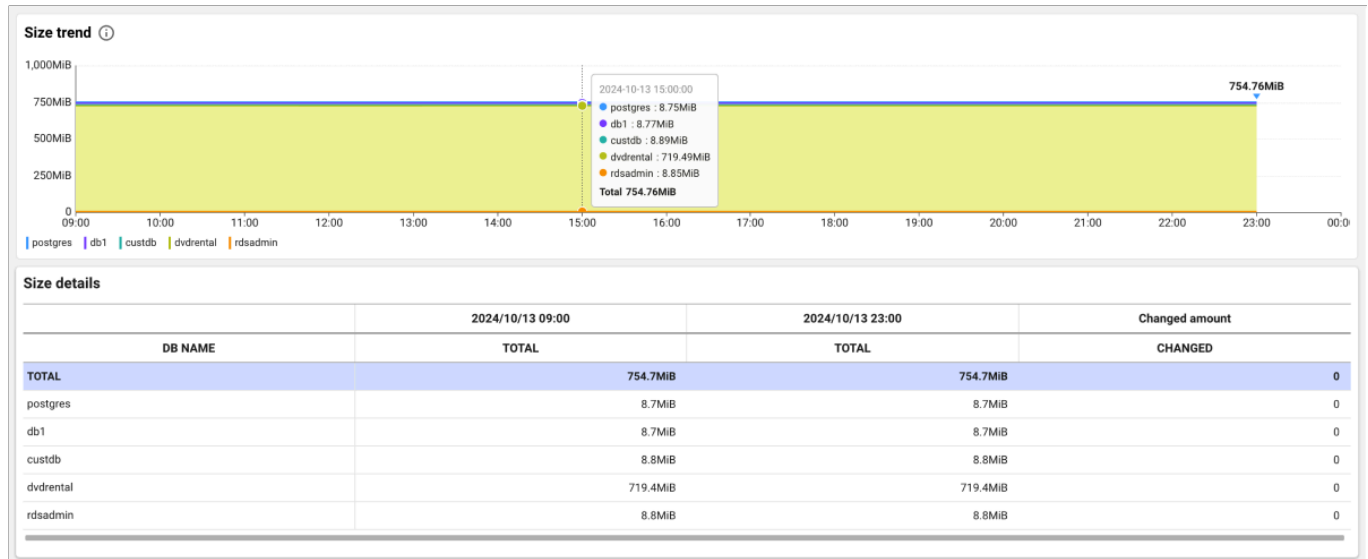
1.

Set the time to search in **Time**. You can also select a lookup time by clicking the green button.
2.

Select a target to view in **Instance**.
3.

Select

Checking the changes over time



- **Size trend**

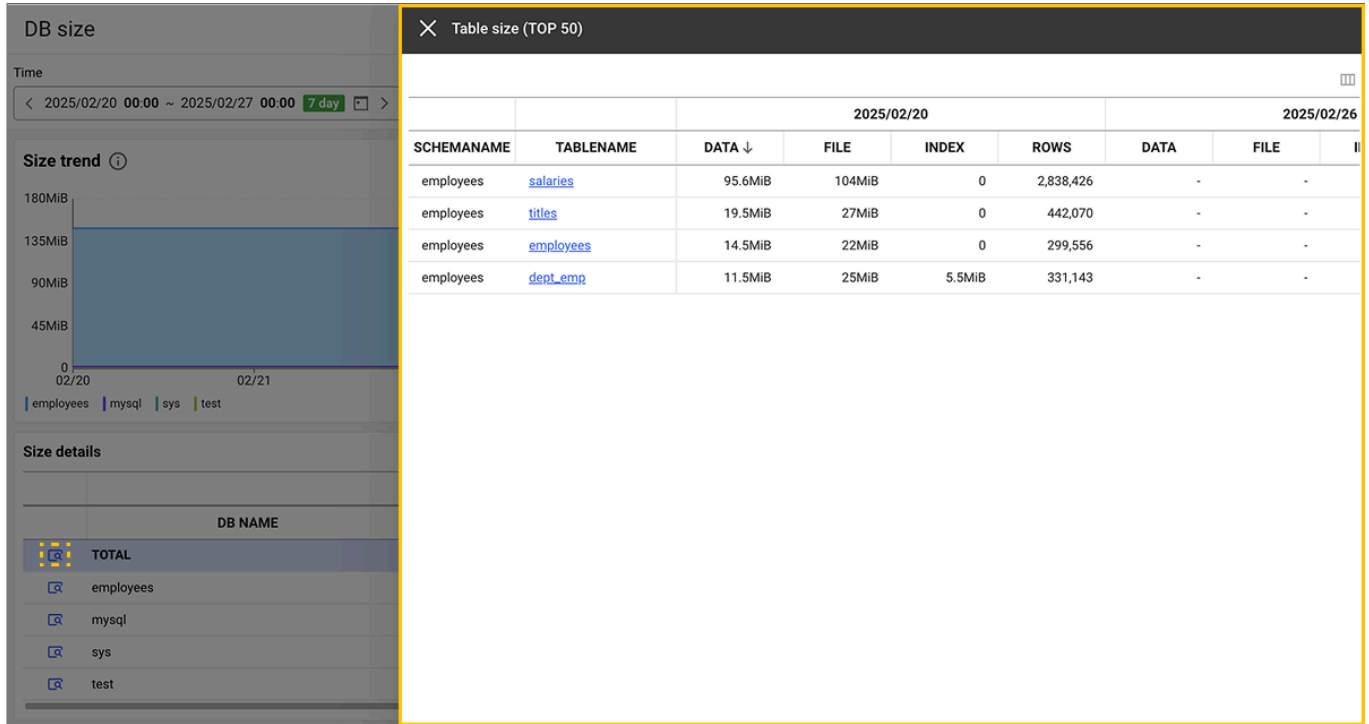
You can visually see the changes in the total size and usage of the database over time. The chart displays the sizes for each database in different color. Hover your mouse over the chart and move it left and right to view the details. You can see information details on the sizes for each database at a specific time point through the tooltip.

- **Size details**

It provides capacity details for each database in a table format. You can check the names (**DB NAME**) for each database, total capacity (**TOTAL**), and changed amount (**CHANGED**). If you select a database, you can see information details in the **Table size (TOP 50)** section.

Checking the sizes for each table

If you select a database in the **Size details** section, you can check the list of sizes for each table corresponding to the top 50 sizes in the database.



You can query the schema name (**SCHEMANAME**), table name (**TABLENAME**), table size (**TABLE SIZE**), index size (**INDEX SIZE**), and total capacity (**TOTAL SIZE**) for each table in the database. You can check the changes in the start time and end time for the query. This allows you to easily detect large tables and supports management efforts for capacity optimization.

When you select the **TABLENAME** column in the table list of the **Table size (TOP 50)** section, the [Object Detail](#) window appears, where you can check the configuration of **column** and **index**.

- This function is supported in DBX agent 1.6.15 or later. For more information about the database roles and related settings, see [the following](#).
- To change the order of table columns or hide specific columns in the **Table size (TOP 50)** section, select  on the upper right. For more information about the column configuration, see [the following](#).

Downloading the viewed data

You can download the searched data as a CSV format file. Select  and then select **Size details** or **Table size (TOP 50)**.

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.

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.

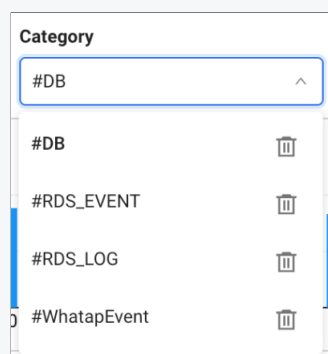
 Parsing the logs

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=/opt/postgresql/logs/postgresql.log
file1=/opt/postgresql/data/logs/pg10/postgresql-Sun.log
file2=/opt/postgresql/data/logs/pg10/postgresql-Mon.log
file3=/opt/postgresql/data/logs/pg10/postgresql-Tue.log
```

- ⓘ Before application, install the XOS agent and then proceed with agent configuration. For more information, see [the following](#).

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

[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.



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
												threadName	http-nio-19090-exec-2
													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
												_event_status_literal_name_	level
													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
												_event_status_literal_name_	level
													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
												_event_status_literal_name_	level
													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
												_event_status_literal_name_	level
													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

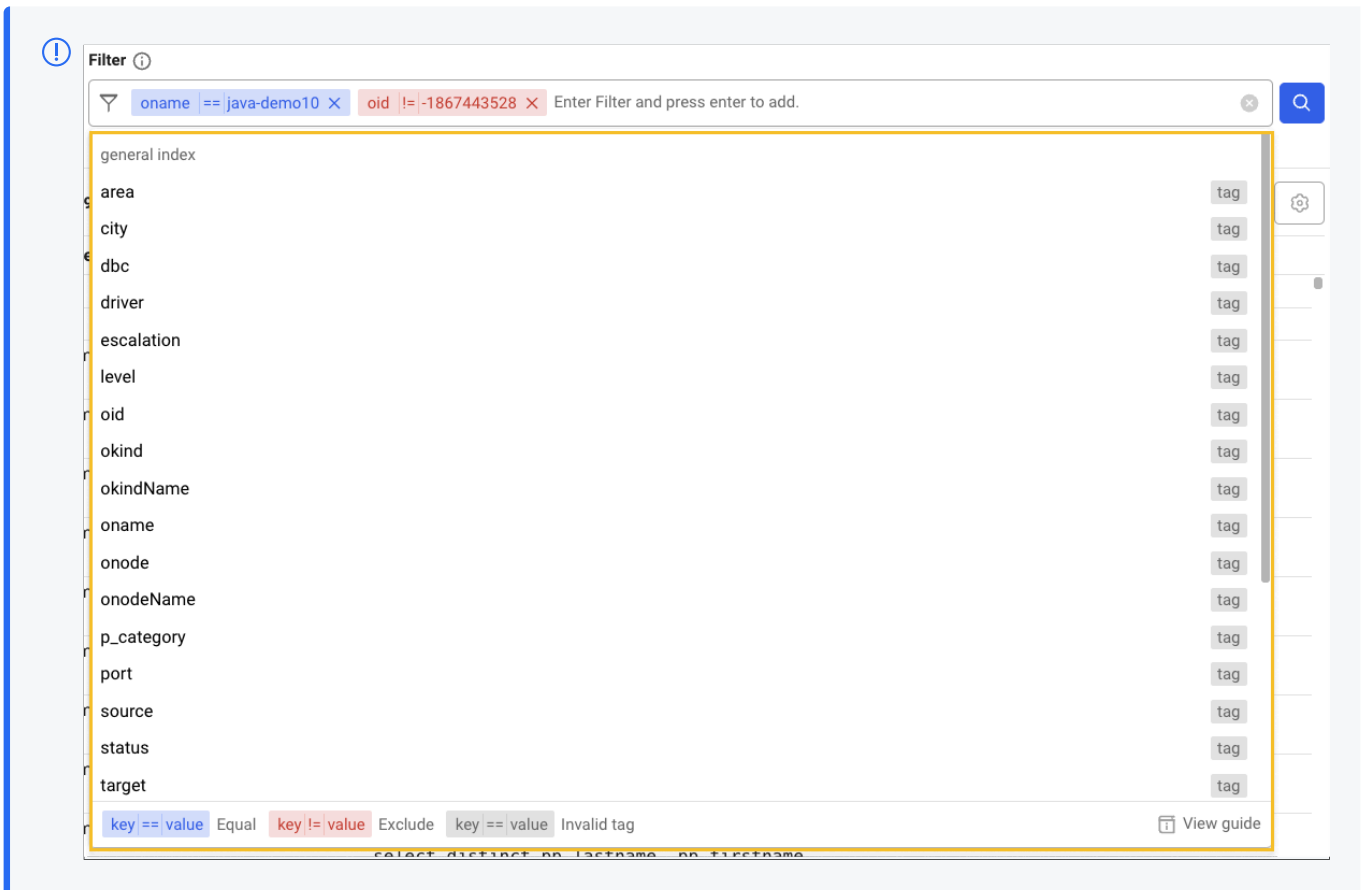
When a filter is applied, the logs that match the specified conditions are filtered. You can enter multiple filters. OR (`||`) if the filters have the same tags, otherwise AND (`&&`).

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**.

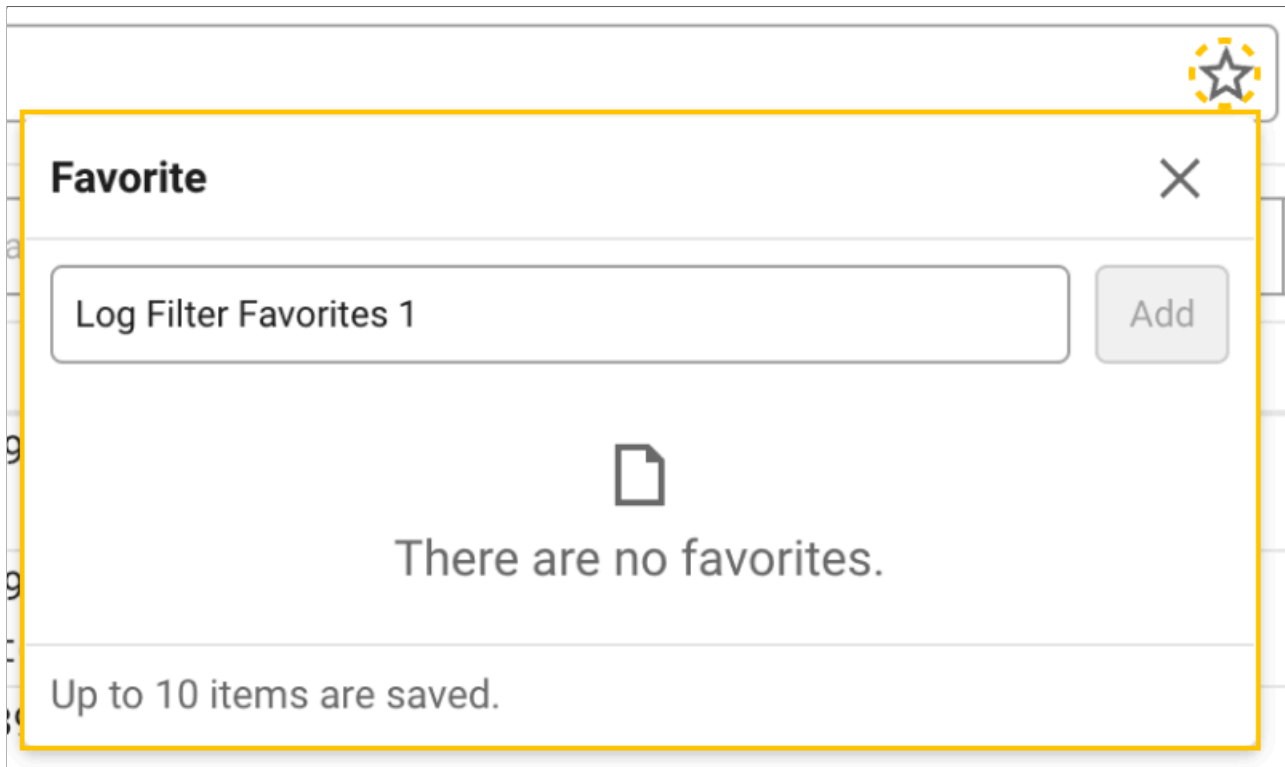
You can also select a recommended value from the **Guide UI** below the input field by clicking it or navigating with the arrow keys.

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.

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	-
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*Ilo*</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.

Live Tail

Category: ApplLog Filter: content == *select* Enter Filter and press enter to add.

select

	oname	Timestamp	Log
b	demo-8105	2023-11-24 14:08:35.620	@txid 5935183317842444630 pcode 5490 oname demo-8105 onodeName node-1 oid -857948929 okindName demo-okind-1 okind -628198688 select
	demo-8105	2023-11-24 14:08:35.624	@txid 4922255873619405363 pcode 5490 oname demo-8105 onodeName node-1 oid -857948929 okindName demo-okind-1 okind -628198688 select productmodelid, avg(listprice) as [average list price]
	demo-8105	2023-11-24 14:08:35.630	@txid 4922255873619405363 pcode 5490 oname demo-8105 onodeName node-1 oid -857948929 okindName demo-okind-1 okind -628198688 select productid, avg(unitprice) as [average price]
	demo-8105	2023-11-24 14:08:35.635	@txid 4922255873619405363 pcode 5490 oname demo-8105 onodeName node-1 oid -857948929 okindName demo-okind-1 okind -628198688 select

- 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
 - `^` / `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 <code>'</code> or <code>"</code> .	Whatap is good.
"Whatap\\ is good."	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

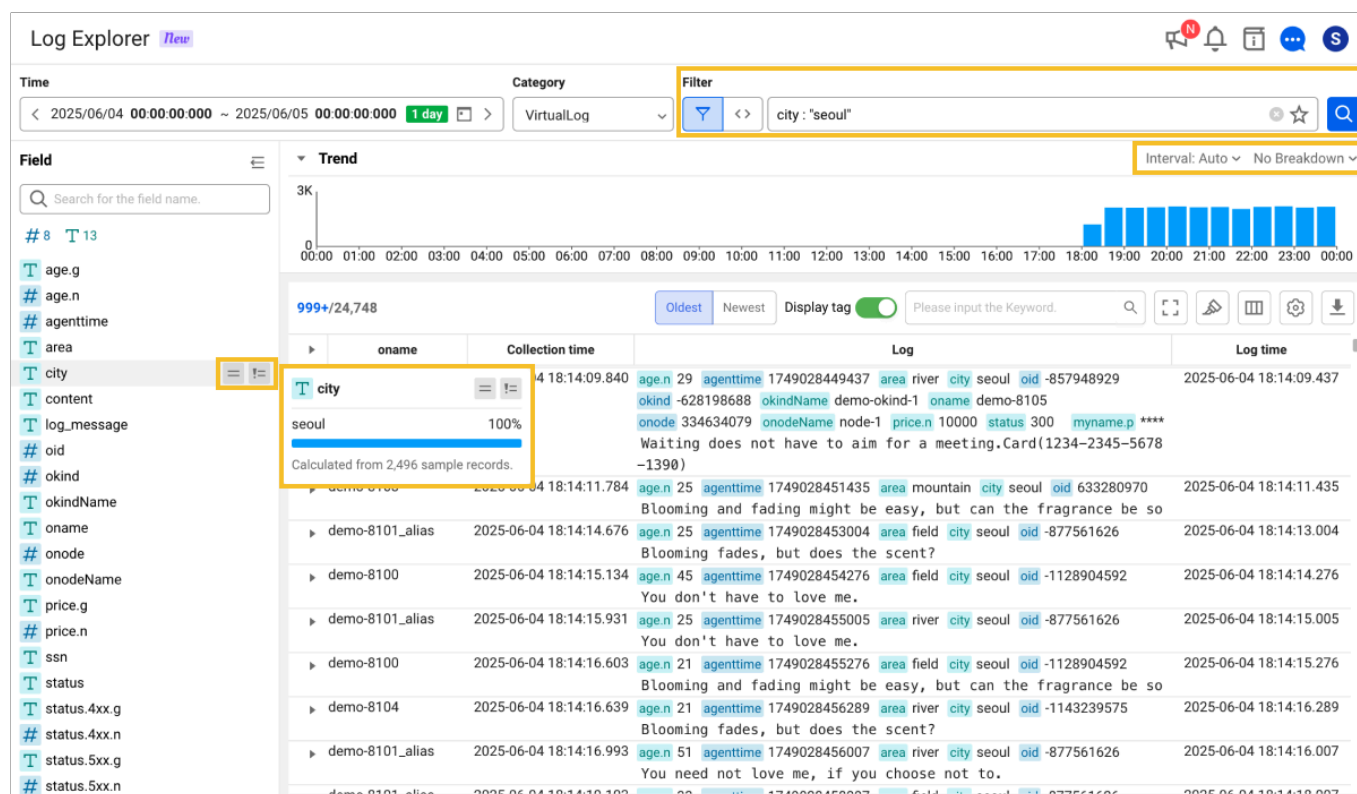
ⓘ Without the **Log Read** role, you cannot enter the menu.

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

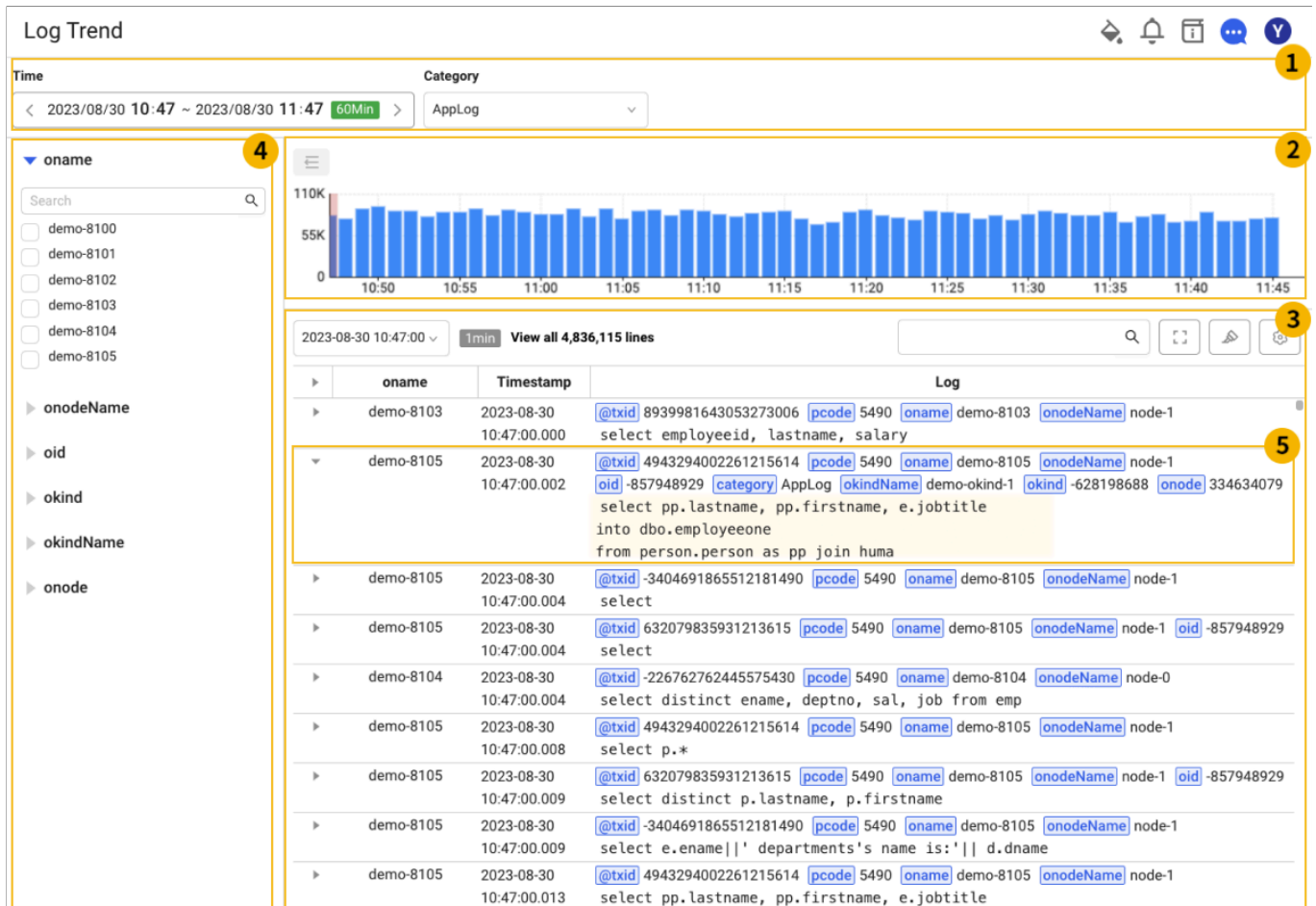
ⓘ Without the **Log Read** role, you cannot enter the menu.

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:

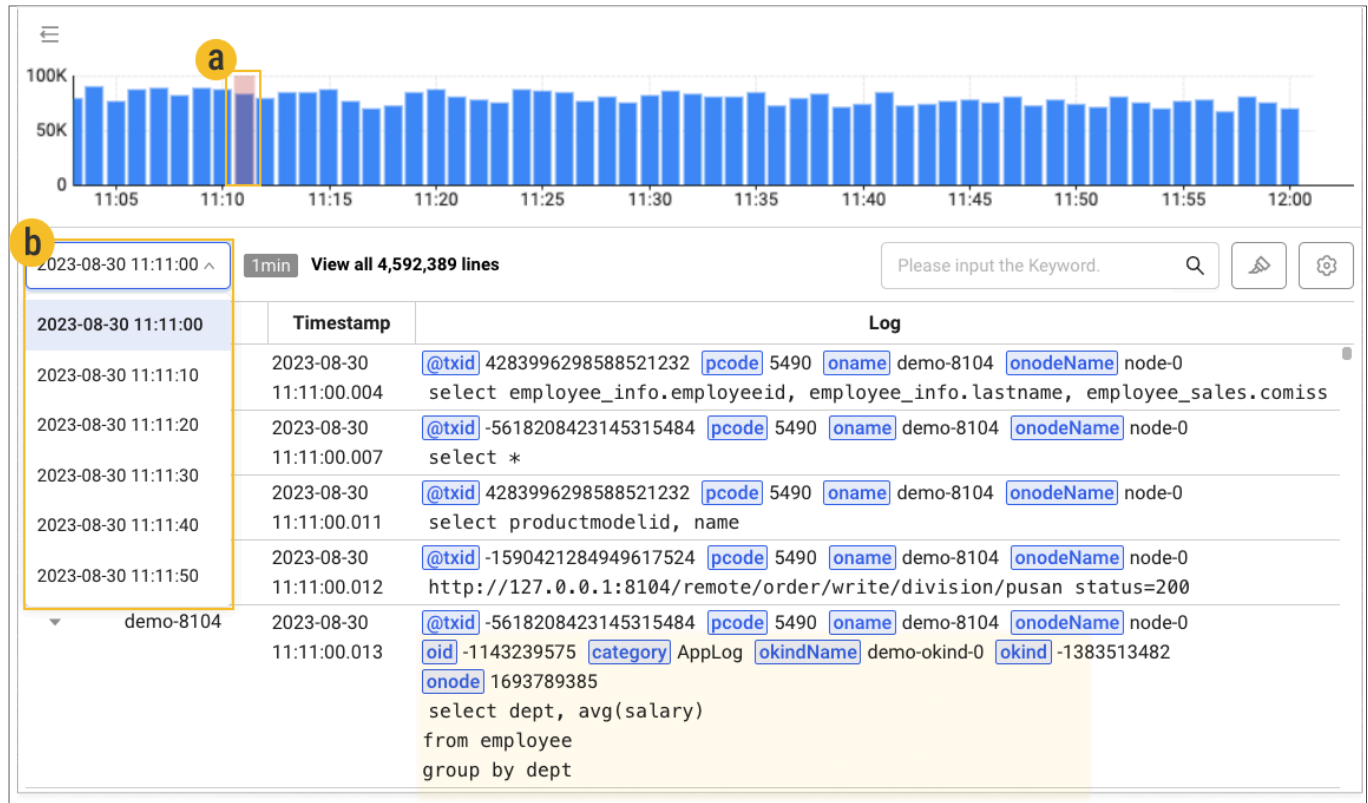
String to input	Description	Result
a b c	Separate each keyword with a space.	a, b, c
"Whatap is good."	To include spaces in keywords, wrap them in "'" or "\"".	Whatap is good.
"Whatap\\ is good."	If \ is included in the keyword surrounded by "", enter it as \\.	Whatap\ is good.

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, Log, and Log. The table lists logs for DBX-1-203-5432, including details like app, pcode, timezone, KST, log_level, LOG, pid, oid, duration, file, and message. The logs show various database operations such as locking tuples, updating data, and acquiring locks.

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.

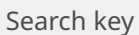
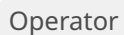
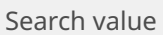
- The first line of the log column is the log's parsed key and value, and the second line is the log's content.
- 3 For each row (log) of the log table, the **More** button exists. If you select the **More** button, you can check the entire content of the log like 4.

- 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

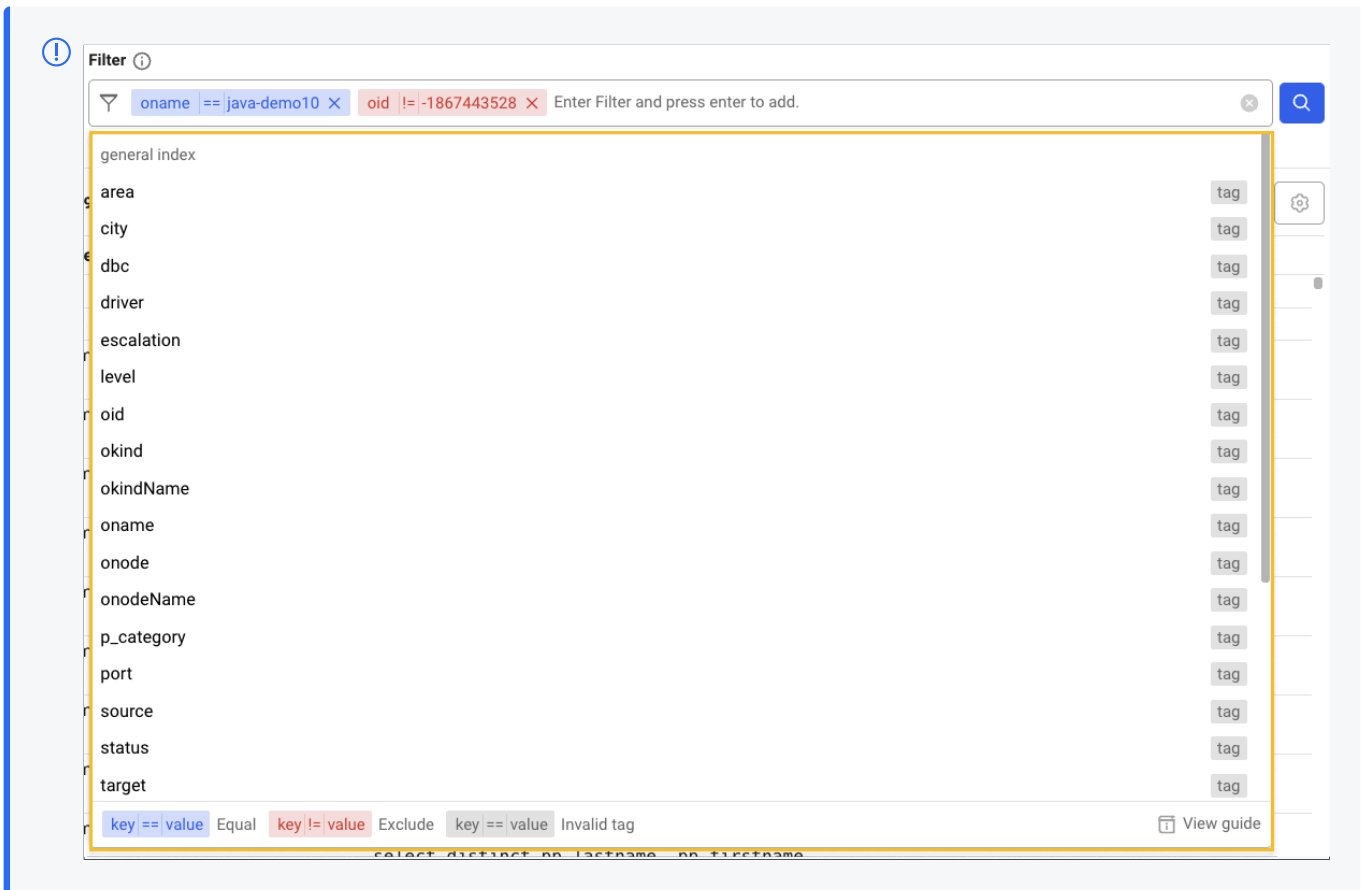
Apply Filter

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**.

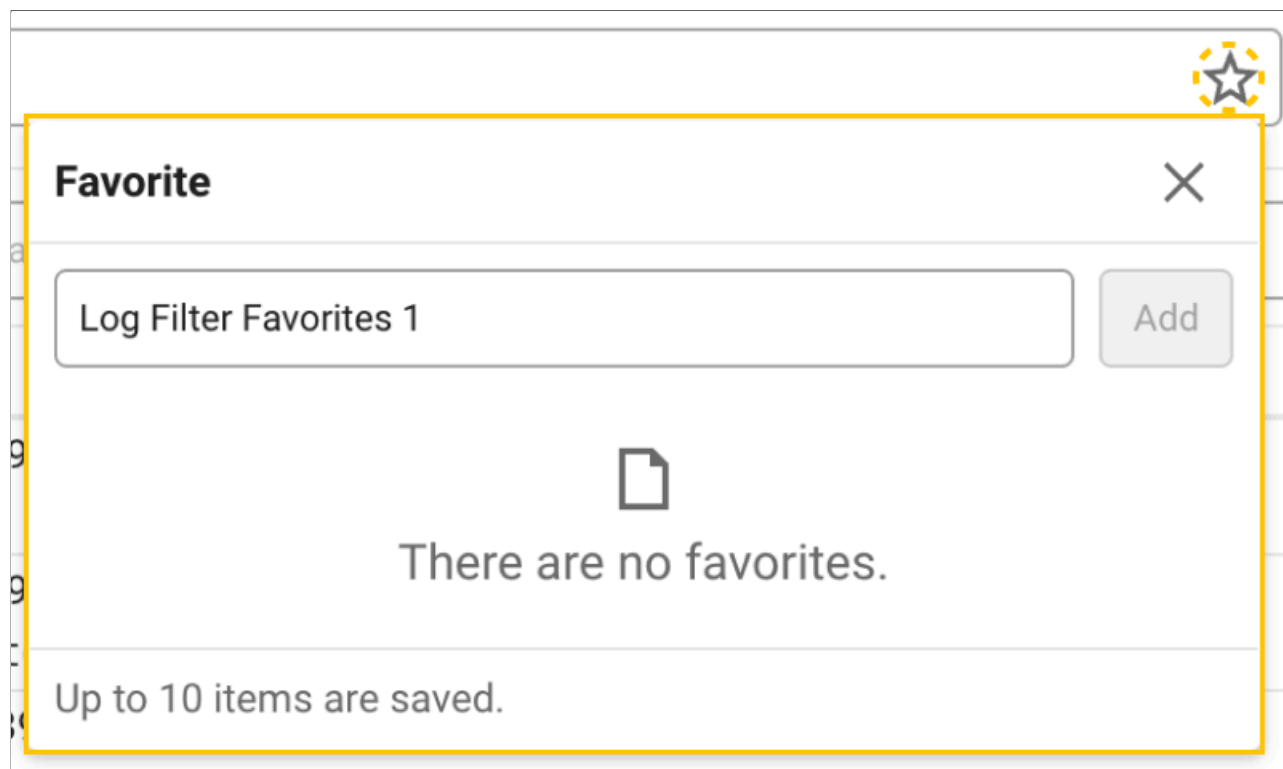
You can also select a recommended value from the **Guide UI** below the input field by clicking it or navigating with the arrow keys.

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*

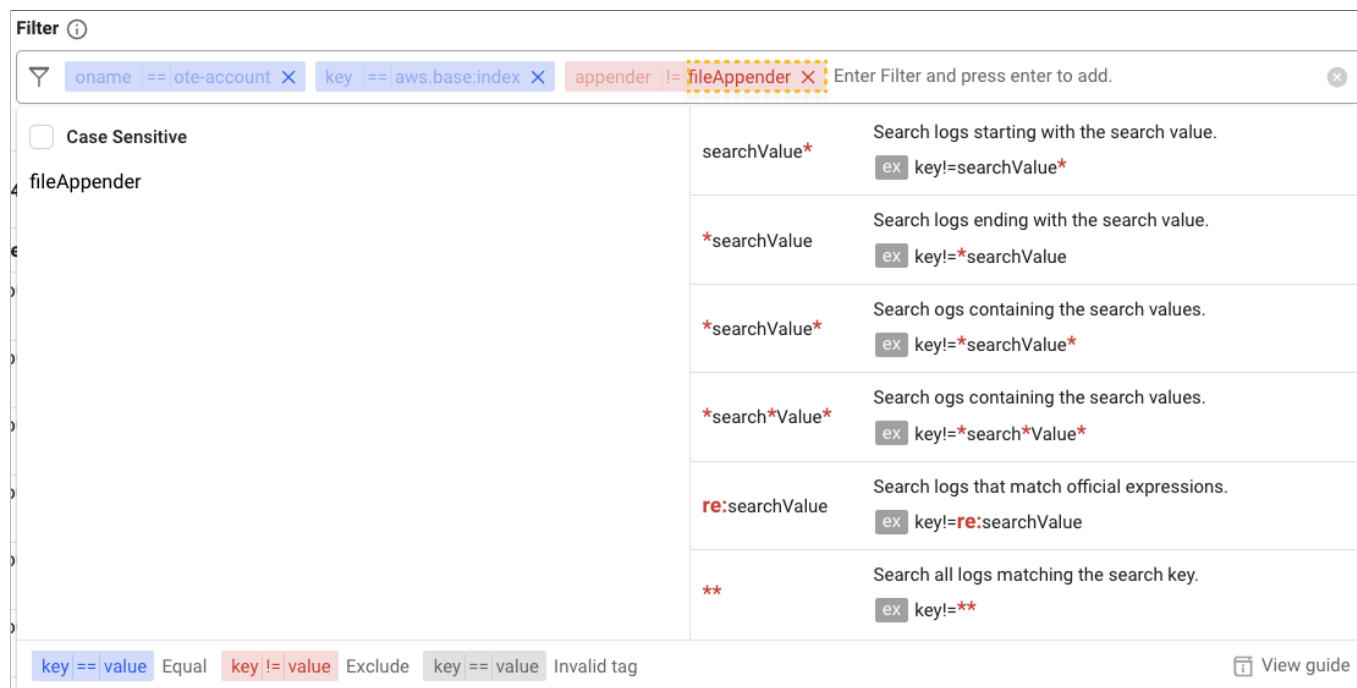
1. Select **Category**. It is required to set the category.
2. In the **Filter** field, enter a desired keyword for a search after leaving spaces on the **content** basis.
e.g. `content *select*`
3. Click **Search** to view the logs. Several logs are viewed first. You can view a maximum of 10,000 logs 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.	
<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>
<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>

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>

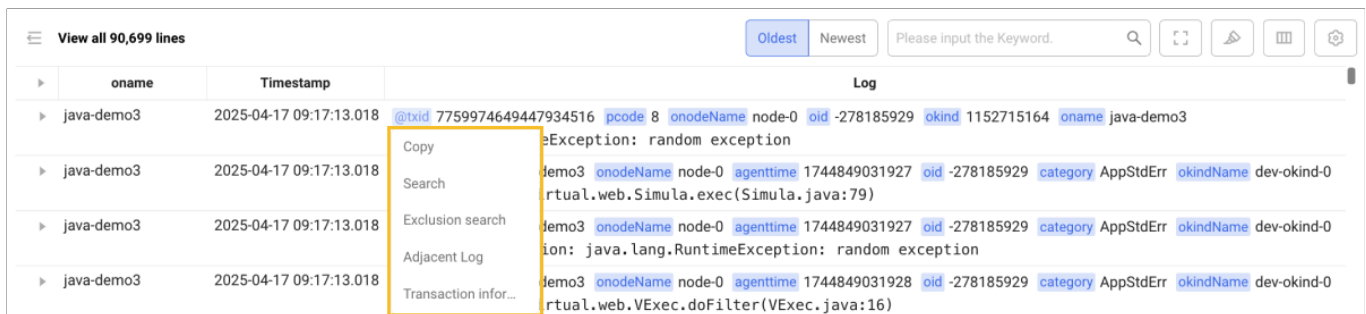
Viewing personal information

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
 - `^ / 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
<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' page with the 'Getting started with log monitoring' tab selected. The page is divided into two main sections: '1 Set up the agent and enable log monitoring' and '2 Log monitoring data setting'.

1 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.

```

# Example configuration for xos.conf
log_level = INFO
log_file = /var/log/xos.log
log_max_size = 100M
log_max_count = 10

```

[Copy](#)

3. Activate log monitoring

Click the toggle button to activate log monitoring after setting is complete.

☒

2 Log monitoring data setting [View guide](#)

[Initialized](#) [Save](#)

Log usage 0Lines

Log lookup password ☐ Turn on to use the log lookup password.

Data retention 1 Days

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.

```
log_level = info
log_file = /var/log/xos.log
log_max_size = 100M
log_max_count = 10
```

[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

Log Configuration						
Getting started with log monitoring Log primary parser setting Log secondary parser setting Fast index setting Log long-term archive statistics Log 1-hour statistics widget data settings						
Log secondary parser setting This parser can be used if a Grok or JSON parser has already been parsed. + Add Save						
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.

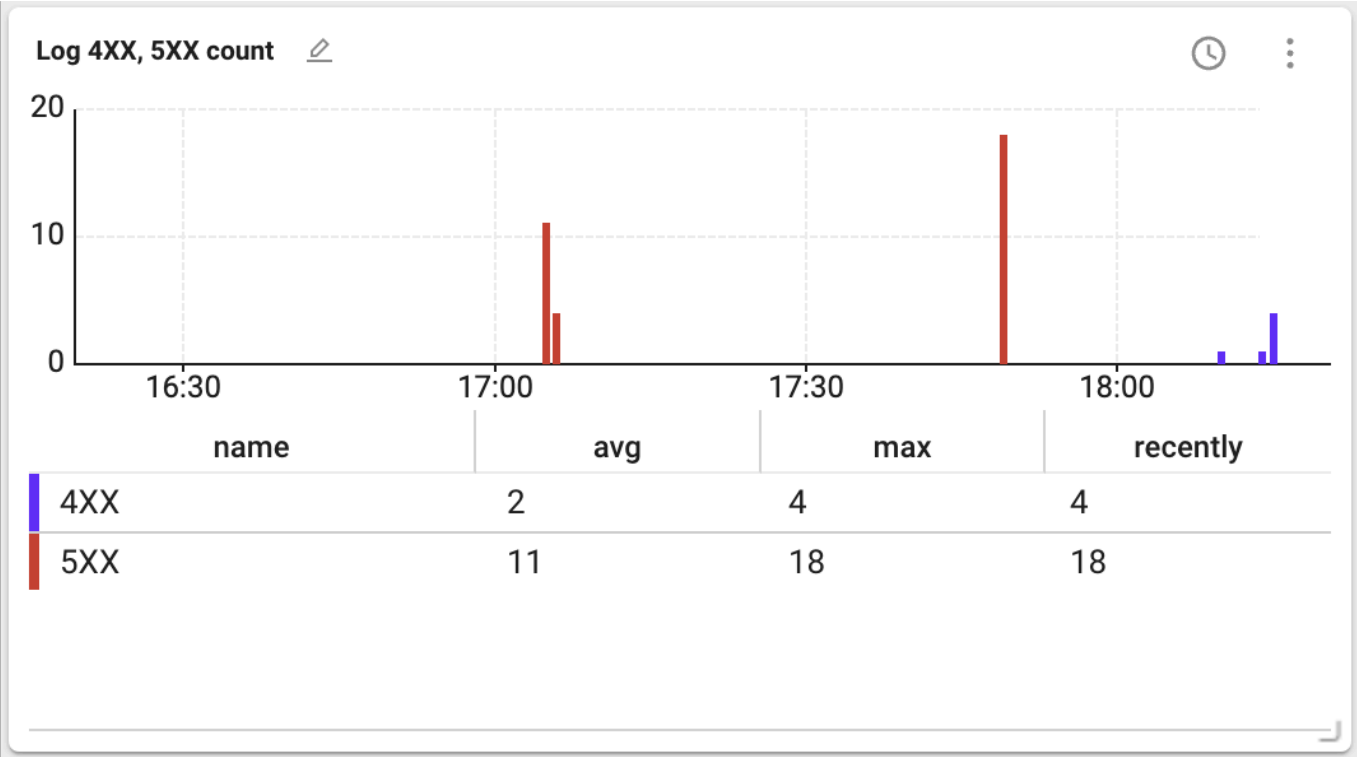
Widget templates

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.

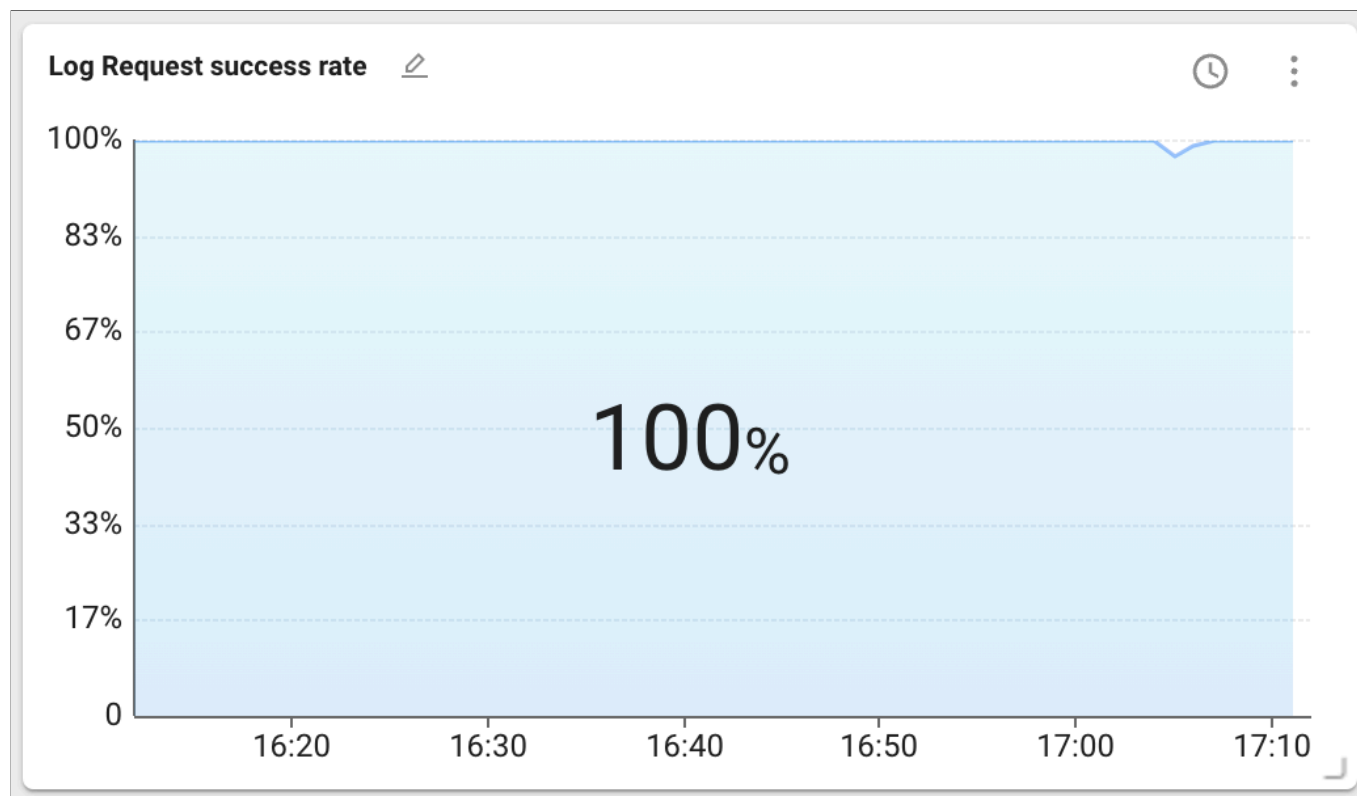
Widget templates

⇌ 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>)

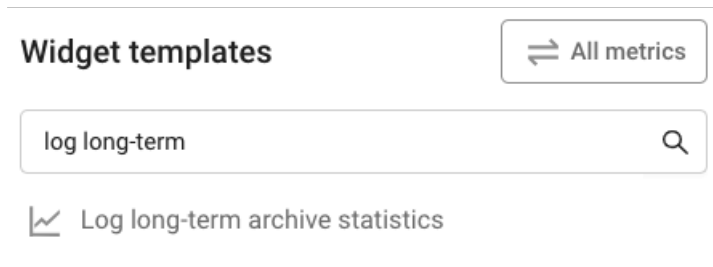
Example

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`.

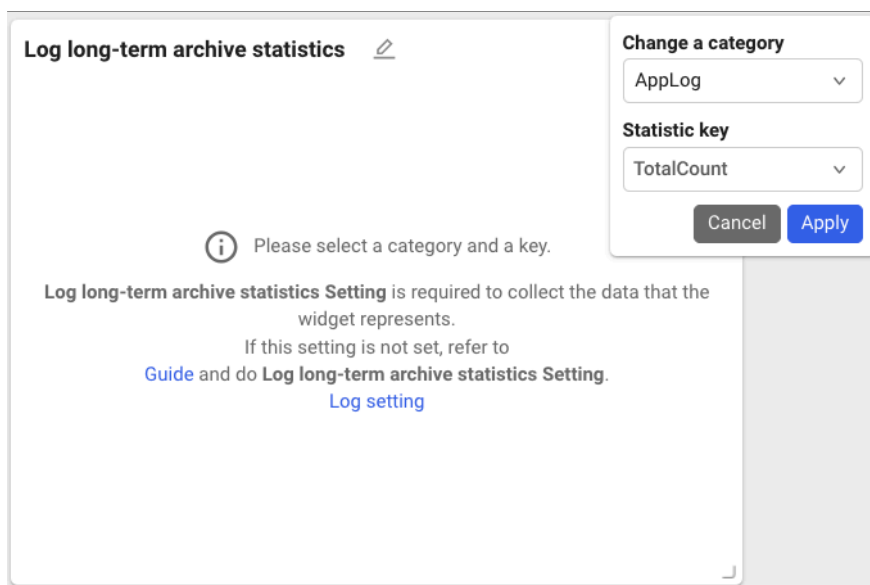
Log long-term archive statistics You can generate statistical data about how many logs have been collected that meet certain conditions. 						+ Add Save	
Category	Statistic key	Log detection condition	Data retention period(disk usage)	Enable			
AppLog	TotalCount	status 200,300	15 Days (About 3 MB)	☐			
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

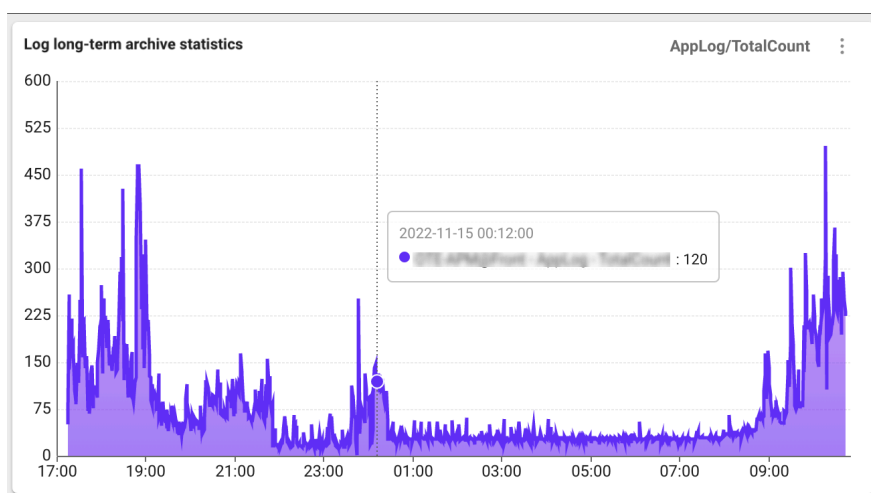
1. Create a widget by searching **Log long-term archive statistics** in **Widget templates** of **Integrated Flex Board**.



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

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.



```
},  
"BASIC": [  
  {  
    "criticalEnabled": "false",  
    "warningEnabled": true,  
    "silentPeriod": 300000,  
  }  
]
```

❗ "criticalEnabled" requires "boolean" type value, not "string" type.

Load

Cancel

Overwrite

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.

The screenshot shows the 'Metrics finder' window with a search bar containing 'cpu'. Below the search bar, it indicates '13 cases/ Total 150 cases'. A table lists various metrics with columns for Category, Name, Unit, Type, Description, Merge, and a Select button.

Category	Name	Unit	Type	Description	Merge	
Application Process Counter	CpuTime	ms	field	CPU usage time	OM: avg TM: avg	Select
JavaOS	ProcessCpuTime	ms	field	Process CPU Time	OM: avg TM: avg	Select
JavaOS	SystemCpuLoad	pct	field	Number of Avg System CpuLoad Percent	OM: avg TM: avg	Select
JavaOS	ProcessCpuLoad	pct	field	Number of Avg Process CpuLoad Percent	OM: avg TM: avg	Select
JavaOS	CpuLoad	pct	field	Number of Avg CpuLoad Percent	OM: avg TM: avg	Select
Application Host Resource	CpuProc	pct	field	CPU utilization in the Proc area	OM: avg TM: avg	Select
Application Host Resource	CpuCores	count	field	Number of CPU cores	OM: avg TM: avg	Select
Application Host Resource	CpuWait	pct	field	CPU utilization taken to wait for disk IOs	OM: avg TM: avg	Select

At the bottom of the table, there is a pagination control showing '< 1 >' and '50 / page'.

3. Select the event level (**Critical**, **Warning**, Info). You can configure multiple levels for a single metric.

❗ **Event Level Behavior**

Even when multiple conditions are satisfied, only the highest-priority level (**Critical** → **Warning** → Info) triggers an event. For details, see the [Level Action Guide](#).

- Event levels are classified as Critical (**Critical**), Warning (**Warning**), and Info (Info).



- It is recommended to configure multiple levels for the same metric with stepwise thresholds.

4. Enter the field, operator, and value for the selected event level.

a. Click + **Add** to add more conditions.

- **&&** : An event is triggered only when **all conditions are satisfied simultaneously**
- **||** : An event is triggered when **at least one condition is satisfied**

b. Click — to remove the condition.



For condition syntax and supported operators, refer to the [Condition Settings Guide](#).

Trigger Count

Configure how many times a condition must be met before an event is triggered. This is useful for detecting persistent issues only.

1. Select the trigger frequency.

- **Consecutive:** An event is triggered when the condition occurs consecutively for the configured number of times.
- **Recent:** An event is triggered when the condition occurs the configured number of times within the selected time window.
 - Available durations: 5 seconds , 10 seconds , 15 seconds , 30 seconds , 1 minute , 2 minutes , 3 minutes , 4 minutes , 5 minutes , 10 minutes , 30 minutes , 1 hour

2. Enter the trigger count.

- The **Interval** value corresponds to the data collection interval of the selected category. (Interval: 10s)

Pause

To prevent excessive event generation, select a pause duration for event execution. After an event occurs, the same event will not be triggered again during the configured time period.

- Available durations: Disabled , 5 minutes , 10 minutes , 15 minutes , 20 minutes , 30 minutes , 1 hour , 2 hours , 3 hours , 6 hours , 12 hours , 1 day

! When recovery notifications are enabled

Even if the same event condition is met again, no notification is sent until the configured time has passed since the **RECOVERED** notification was received.

Resolved Notification

When the resolved notification feature is enabled, you can track state changes after an event occurs (in progress → resolved). The event is shown as an ongoing event in the **Event History** menu.

- Resolved notifications apply only to **Critical** and **Warning** levels. Info-level events always operate as one-time events.
- When a **Critical** or **Warning** event is resolved, a notification with the RECOVERED (normal) state is sent.
- Click the toggle button to **enable** or **disable** this feature.
 -  **Enabled:** Events operate based on state. They enter an in-progress state when the threshold is exceeded and transition to a resolved state when the value recovers below the threshold.
 -  **Disabled:** A one-time event is generated each time the threshold is exceeded.

Event Operation Time

Configure tags so that events operate only during specific time periods (such as business hours, non-business hours, or maintenance windows). If no tag is configured, events operate continuously for 24 hours while enabled.

1. Click **+ Add**.
2. In Event Operation Time, click **+ Create New Tag**.

3. In **Create Event Operation Time Tag**, select the **tag name, days, time, and color**, then click the [Apply] button.
4. You can view created tags in the tag list, and apply them by selecting the checkbox.
5. To modify or delete a tag, click the  icon and update or delete it in **Edit Event Operation Time Tag**.

 When a tag is deleted, it is removed from all users to whom it was applied. However, tags that are currently in use by event rules cannot be deleted.

Simulation

The simulation feature lets you pretest configured event conditions using historical data. If the expected number of events is too high or too low, you can adjust the conditions to find optimal threshold values.

- **Rule Validation:** Verify how newly created event rules would behave when applied to historical data.
- **Threshold Tuning:** Adjust thresholds to an appropriate level when events occur too frequently or not at all.

Start Simulation

1. After completing event condition settings, click the [Simulation] button.
2. Review the expected number of events and the visualized results.

Understanding Simulation Results

1. Time-Series Data

A graph displays metric changes over time along with threshold lines for each level, allowing you to intuitively identify when events are triggered.

2. Event Counts by Level

The number of events for the Critical, Warning, and Info levels is shown on the right side of the screen. This helps you understand overall event distribution and frequency.

3. Target-Level Details

Hover over a specific point on the chart to see the exact metric value and monitoring target at that moment. When multiple targets exist, they are distinguished by color. If patterns differ by target, you can use filtering to analyze them individually.

4. Peak Interval Identification

Identify periods where metrics spike sharply to understand when issues occur most frequently or to recognize recurring patterns.

✓ Simulation Tips

- **Simulate Different Time Ranges**

Run simulations for weekdays vs. weekends and peak vs. off-peak hours to set optimal thresholds for each time period.

- **Include Actual Incident Periods**

Include past outage periods in simulations to verify whether current rules accurately detect real incidents.

- **Add Target Filtering**

When many targets are monitored, simulation results can become complex. Use filters to focus on specific targets for clearer insights.

- **Adjust Thresholds Gradually**

Instead of making large changes at once, adjust thresholds incrementally, such as in 10 percent steps, to find optimal values.

Select Event Target

If no event target is specified, events are triggered for all targets in the project, which can result in excessive events. It is therefore recommended to clearly define the targets to which the event applies.

ⓘ If you change the event targets, the number of events may change. Click the **Simulation** button and run it again to check the updated expected event count.

Target Selection

You can specify the targets to which the event applies using either **Select Input** or **Direct Input**.

- **Select Input** **Direct Input**

1. Select or enter a tag, operator, and value.
 - a. If it is difficult to find a tag, you can search for it using the **Metrics Finder**.
2. Click + **Add** to add event targets using conditions (`&&` , `||`).

Enter the tag, operator, and value of the target to monitor directly.

- When you choose direct input, a list of available tags (**Tag Select Input**) is provided below the input field.

```
ex. endsWith(okindName, 'example_name') && container == 'prod.billing'
ex. ${4xxErrorType} == '401'
```

 If you change the input method, any content you have entered will be reset.

Basic Information and Notification Settings

Configure the **event name**, **message**, and **recipients**. Because this information is displayed as is in event history and actual notification messages, it should be clear and easy to identify.

Event Activation

Configure whether the event is triggered when the defined conditions are met.

-  **Enabled:** The event is triggered when the configured conditions are satisfied.
-  **Disabled:** Conditions are not evaluated.

Event Name

Enter the event name to be used as the event title. The configured event name is displayed in the **Event List** and **Event History** menus and can be used as a search keyword.

Message

Enter the event message. The message you enter is used in event history and notifications. Click the  icon to view previously written message history. You can include variables in the message to insert actual values at the time the event occurs.

- **Variable Usage Rules**

- Variable format: `${metric_name}`
- Only metrics within the **same category as the configured category** can be used as variables.
- You can combine multiple variables to create a more detailed message.
- The list of available variables can be found in the **Metrics Query** menu.

- **Variable Usage Example**

	Event Configuration	When Event Occurs
Title	Disk Usage Increase	Disk Usage Increase
Message	Disk = <code>\${disk}%</code>	Disk = 89.9%

Event Recipients

Specify the members who will receive event notifications.

- **Receive All Receive by Tag Selection**

Notifications are sent to all targets (members and channels) in the project that have event reception enabled.

Notifications are sent to project members who have the selected tag.

1. Select **Receive by Tag Selection**.
2. In the **Reception Tag** section, click the [+ Add Tag] or [+] button.
3. Under the **Event Reception Tag** window, click + **Create New Tag**.
4. In the **Create Tag** window, enter a tag name, select a color, and click the [Create Tag] button.
5. You can edit or delete created tags by clicking the  icon in the tag list.

6. In the **Event Reception Tag** window, select the desired tag from the **Tag List** to apply it.

ⓘ Reception Tags

Reception tags are a feature for managing users or channels that receive notifications **at a group level**. When a reception tag is assigned to an event, notifications are delivered only to users associated with that tag. If no reception tag is set, notifications are sent to all users in the project.

Edit/Delete Event

1. Go to **Alert Notifications > Event Settings** and open the **Metrics** tab.
2. In the event list, click the  icon for the event you want to edit or delete.
3. In the **Edit Event Rule** window, update the options and click the [Save] button.
 - a. To delete the selected event, click the [Delete] button in the upper-right corner of the **Edit Event Rule** window.

Event Sharing

You can save metric event settings as a JSON file to share configurations with other users or import settings created by others.

- JSON file name: `event-rules-YYYYMMDD.json`

Export

1. Click the [**JSON** ] button at the top right of the screen.
2. When the JSON editor opens, click the [ Export] button.
 - If you use the export function after searching for events, only the searched list will be downloaded as a JSON file.
3. After the JSON file is downloaded, share it with other users.

Import

1. Click the [] button at the top right of the screen.
2. Select the JSON file downloaded using the **Export** function.
3. When the JSON editor opens, choose either [Add to List] or [Overwrite].

 It is recommended to use this feature between projects of the same product type. You can import event settings from projects of different products, but they may not function correctly.

Edit in JSON Format

1. Click the [**JSON** ] button at the top right of the screen.
2. When the editor opens, modify the content according to the JSON format.
3. After completing the edits, click the [Save] button at the bottom of the screen.

 If the modified content does not conform to the JSON format, an error message will be displayed at the bottom of the screen and the content cannot be saved. The error message may vary depending on the type of formatting issue.

JSON error

JSON Data Structure

```
{
  "version": 2,
  "eventId": "zcef7s7f27rum1",
  "enabled": true,
  "stateful": false,
  "title": "Active Transaction",
  "message": "Active Transaction = ${active_tx_count}",
  "category": "app_counter",
  "alertLabel": [
    "oid"
  ],
  "repeatCount": 1,
  "repeatDuration": 0,
```

```

"silent": 300000,
"receiver": [],
"timeTag": [],
"selectString": "",
"conditions": [
  {
    "level": 20,
    "enabled": true,
    "rule": "active_tx_count > 100"
  }
],
"createTime": 1763632674345,
"lastModifiedTime": 1763632674347,
"lastModifiedUser": "support@whatap.io",
"basic": true,
"metaId": "java004",
"selectCondition": {
  "oid": [
    "-1010758404",
    "-250906941"
  ]
}
}

```

Table | JSON Data Structure

JSON Field	Type	Description	Notes
version	Integer	Version of the event rule	Restricted
eventId	String	Unique identifier of the event rule	
enabled	boolean	Whether the event is enabled	
stateful	boolean	Whether the event is state-based	
title	String	Event rule name defined by the user	
message	String	Event rule message defined by the	

JSON Field	Type	Description	Notes
		user	
category	String	Data category	
alertLabel	List<String>	Default identifier (primary key) value assigned per category for event state management	Restricted
repeatCount	Integer	Number of repetitions	
repeatDuration	Long	Duration for repetition	
silent	Integer	Pause duration	
receiver	List<String>	List of recipient tag keys	
timeTag	List<String>	List of operation-time tag keys	
selectString	String	Target selection	
conditions	List<Object>	List of trigger conditions by level	
◇ level	byte	Level value	Critical : 30, Warning : 20, Info : 10
◇ enabled	boolean	Whether the level is enabled	
◇ rule	String	Trigger condition for the level	
createTime	Long	Initial creation time of the event rule	
lastModifiedTime	Long	Last modification time of the event rule	
lastModifiedUser	String	Account that last modified the event rule	

JSON Field	Type	Description	Notes
basic	boolean	Whether the event was created using Quick Setup	Restricted
metaId	String	Unique identifier of the Quick Setup template	Restricted: This value exists only for events created via Quick Setup (basic=true).
selectCondition	Object	Target selection value for Quick Setup	This value exists only for events created via Quick Setup (basic=true).

 Fields marked as “Restricted” may affect event behavior, so it is recommended not to modify their values.

Appendix

Trigger Conditions and Target Selection Guide

Metric alert trigger conditions and event target selection use the same syntax. However, trigger conditions use the field key as the variable, while target selection uses the tag key as the variable.

Level Behavior Guide

Event levels are categorized into Critical, Warning, and Info.

1. Priority-based Event Triggering

When multiple level conditions are met at the same time, only the event with the highest priority is triggered.

Example

Configuration:

- Warning: CPU > 70%
- Critical: CPU > 90%

Current state: CPU 95%

→ Result: Only the Critical event is triggered (Warning is suppressed)

2. Level Escalation

For events that maintain an in-progress state, when a higher-level condition is met while a lower-level event is already in progress, both levels enter the in-progress state.

Example

Configuration:

- Warning: CPU > 70%
- Critical: CPU > 90%

Scenario:

- 1) CPU 80% → Warning triggered (in progress)
- 2) CPU 95% → Warning (in progress), Critical (in progress)

3. Level De-escalation

For events with an in-progress state, when the level transitions from a higher level to a lower level, the higher-level event is resolved while the lower-level event remains active.

Example

Configuration:

- Warning: CPU > 70%
- Critical: CPU > 90%

Scenario:

- 1) CPU 97% → Critical triggered
- 2) CPU 85% → Critical resolved, Warning remains
- 3) CPU 65% → Warning resolved

4. Execution Flow

- Warning: CPU > 70%

- Critical: CPU > 90%

Metric value: 60% → 75% → 92% → 85% → 60%

State: Info Warning Critical Warning Info
 (triggered) (triggered) (maintained) (resolved)
 Warning
 (maintained)

- ⓘ Events that have an in-progress state remain active as long as their conditions are met and continue until those conditions are cleared.

Condition Configuration Guide

- ⓘ If a metric name starts with a number or contains special characters, you must wrap it in the `${metric_name}` format.

- Brackets `()`, `[]`
- Operators `+`, `-`, `*`, `/`, `%`
- Delimiters `:`, `@`, `#`, `,`, `,`, (space)
- Other special characters `!`, `^`, `&`, `|`, `~`, `\"`, `=`

Examples using `${}` :

```
${cpu(xos)} > 50
${mem[0]} >= 100
${cpu-usage} > 80
${namespace:cpu} > 70
${metric name} > 60
${4xx_error} > 10
```

Usable without `${}` : Metric names that contain only alphabetic characters, underscores (`_`), or dots (`.`)



```
cpu > 80
cpu_usage > 50
CPUUtilization.Average > 0.8
```

1. Comparison Operators

Numeric Comparison

```
Greater than:      cpu > 80
Greater than or equal:  cpu >= 80
Less than:         memory < 1000
Less than or equal:  memory <= 1000
Equal to:          status == 200
Not equal to:       error != 0
```

String Comparison

```
status == 'OK'
region == "us-east-1"
```

2. Arithmetic Operators

- Basic arithmetic operations

```
Addition:    cpu + 10 >= 90
Subtraction:  memory - 100 >= 500
Multiplication:  cpu * 2 >= 100
Division:       disk / 1024 >= 100
Modulo:         value % 10 == 0
```

- Controlling precedence with parentheses

```
(cpu + memory) * 2 >= 200
(disk - used) / total >= 0.2
((cpu + memory) / 2) >= 50
```

- Negative values

```
cpu > -100
```

3. Logical Operators

- AND operator (&&)

```
cpu > 80 && memory > 1000  
${cpu(xos)} > 50 && ${mem(xos)} > 60
```

- OR operator (| |)

```
cpu > 90 | | memory > 90  
disk < 10 | | network > 1000
```

- Combined logical operations

```
(cpu > 50 && memory > 50) | | disk < 20  
cpu > 80 && (memory > 1000 | | disk > 500)
```

4. Pattern Matching Operators

- LIKE operator

```
oname like 'prod-*'  
url like '*error*'  
message like 'WARN%'
```

- NOT LIKE operator

```
oname not like 'test-*'  
url not like '*debug*'
```

5. Built-in Functions

- Null check functions


```

isNull(value)      # Checks whether the value is null
isNotNull(value)   # Checks whether the value is not null
nvl(value, 0)      # Returns a default value if null
isEmpty(str)       # Checks whether the string is empty
isNotEmpty(str)    # Checks whether the string is not empty

```

- Aggregate functions

```

sum(cpu, memory, disk)    # Sum
avg(cpu, memory)          # Average
max(cpu, memory, disk)    # Maximum
min(cpu, memory, disk)    # Minimum
count(value1, value2, value3) # Count

```

- Math functions

```

round(cpu, 2)    # Rounds to 2 decimal places

```

- String functions

```

length(str)      # String length
startsWith(str, 'prefix') # Checks prefix
endsWith(str, 'suffix') # Checks suffix
indexOf(str, 'search') # String index
substring(str, 0, 10) # Substring
trim(str)        # Trims whitespace
replace(str, 'old', 'new') # String replacement
hasStr(str, 'search') # Checks string containment

```

- Conditional functions

- **if:** A conditional function that returns the `trueValue` if the condition evaluates to `true`, or the `falseValue` if it evaluates to `false`

Syntax

```

if(condition, trueValue, falseValue)

```

Example

```
if(value >= 90, 'High', 'Medium') == 'High'
```

- **decode:** Compares a value against conditions and returns the result of the matching condition, or a default value if no match is found

Syntax

```
decode(value, condition1, result1, condition2, result2, ..., conditionN, resultN, defaultValue)
```

Example

```
decode(value, 1, 'Low', 2, 'Medium', 3, 'High', 'Unknown') == 'Unknown'
```

- **in:** Returns `true` if the value matches any of the specified candidates; otherwise returns `false`

Syntax

```
in(value, candidate1, candidate2, ..., candidateN)
```

Example

```
in(status, 200, 201, 204) == false
```

! Writing Guidelines

- Recommended
 - Use simple variable names without `${}` (e.g. `cpu`, `memory`)
 - Always use `${}` when special characters are included
 - Use parentheses to clearly separate complex expressions
 - Enter function names exactly as defined (case-sensitive)
- Avoid



- Incomplete expressions (e.g. `cpu > , memory && )`
- Mismatched parentheses (e.g. `(cpu > 80 && cpu > 70) )`
- Consecutive or invalid operators (e.g. `cpu >> 80 , cpu >< 80 )`

Composite Metrics

Home > Select Project >  **Alert** > **Event Configuration** > **Composite metrics** tab

You can create events and send alert notifications using complex rules.

Default Options

- **Search:** Search for specific events by selecting an event name or an event recipient tag
- **JSON Download/Upload:** Edit event rules directly in the JSON editor, or download them as a file, modify them, and upload them again
- **Alert Message Settings:** Select detailed fields for alert messages and edit the content for each field
- **+ Add Event:** Add a new composite metrics event

Permissions

To configure **Composite metrics** events, you must have **Alert Settings** permission. You can grant this permission by clicking the  icon for the member in **Home > Select Project > Management > Integrated Member Management**.

Table | Composite Metrics Event List Layout

Item	Description
Event Title	User-defined event name - Up to 255 bytes
Condition	Event trigger conditions
Notify on Event Recovery	Whether to receive RECOVERED (normal) notifications when Critical or Warning events are resolved – Enabled: Sends a RECOVERED notification when the state changes from in-progress to normal, and allows you to distinguish in-progress and resolved states in event history – Disabled: Triggers a one-time event each time the threshold is exceeded

Item	Description
Event Recipient Tag	Users or groups that receive notifications for the event
	Edit or delete the event
	Event enablement status – Enabled: Event is triggered when conditions are met – Disabled: Event does not run and conditions are not evaluated

Event Configuration

Composite metrics events define event conditions based on **MXQL**, the metric data query language. The **Create from Chart** feature provides a combo box to support MXQL auto-completion.

Add Event

1. Click the **Composite metrics** tab from  **Alert > Event Configuration**.
2. Click the [+ Add Event] button on the right side of the screen.
3. In the **Composite metrics** window, click **Create from Chart**.
 - a. You can use the provided templates.
4. In the **Event Configuration** screen, first query the event data, then configure the event conditions.

Query Event Data

Query event data to build charts. In the **Event Configuration** screen, select either the **Widget** or **Text** option under **Query event data** on the right to configure events.

- **Widget** Text

Using the options for configuring a time series chart, you can automatically complete the **MXQL** used for event configuration.

1. Select the date range for querying data.
2. Select the **Category** (target) for querying data.
3. Click + **Add Filter** under **Filter** to create event conditions.
 - a. In the filter window, select an expression, tag, and filter value.
 - b. Click + **Add** to add additional conditions.
 - c. To delete a condition, click the – icon next to it.
4. Under **Grouping**, select the grouped metric data. Multiple selections are supported.
5. Under **Time Unit**, set the time interval for grouped data by selecting `seconds` , `minutes` , or `hours` .
6. Under **Field**, select the fields to be used for event trigger conditions. Multiple selections are supported.

 MXQL entered in text mode is not compatible with widget mode.

An editor appears where you can directly modify **MXQL**.

1. Edit **MXQL** directly and click the **Query** button.
2. After selecting a date, you can view the results as a chart or graph.

Notification

Enter the basic information for alert notification settings.

- **Enable Event:** When the toggle is enabled, the event becomes active.
- **Level:** Select the event level (Critical, Warning, Info).
 - **Notify on Event Recovery:** When the toggle is enabled, a notification is sent when the event is resolved and enters the normal (RECOVERED) state.
- **Title:** Enter a title for the event rule.
- **Message:** Enter the content of the alert message delivered when the event is triggered. You can use variables in the message to include actual values at the time the event occurs.

Message example

✔ Variable Usage Rules

- Variable format: `${metricName}`
- Only metrics within the **same category as the configured category** can be used as variables.
- You can combine multiple variables to compose a detailed message.
- The list of available variables can be checked in **Metric Query**.

Event Trigger Conditions

- **Data Query Range:** Select the real-time data query window.
 - Available ranges: 30 seconds , 1 minute , 2 minutes , 5 minutes , 10 minutes , 30 minutes , 1 hour , 2 hours , 3 hours
- **Condition:** Enter the event trigger condition. Only fields included in the event data query can be used.

Additional Information

- **Interval:** The alert condition is evaluated at the selected interval (1 minute , 5 minutes , 10 minutes , 30 minutes , 1 hour , 12 hours).
- **Silence:** After an event is triggered, the same event is suppressed for the selected duration (unset , 1 minute , 5 minutes , 10 minutes , 30 minutes , 1 hour , 12 hours).
- **Event Active Time:** Configure tags to control when the event runs, such as working hours, non-working hours, or maintenance windows. If no tag is set, the event runs continuously for 24 hours while it is enabled.
 1. Click + **Add**.
 2. Under **Event Active Time**, click + **Create New Tag**.
 3. In **Create Event Active Time Tag**, select a **Tag Name**, **Days**, **Time**, and **Color**, then click the [Apply] button.
 4. Created tags appear in the tag list and can be applied by selecting the checkbox.
 5. To edit or delete a tag, click the  icon and modify or remove it in **Edit Event Active Time Tag**.

 When a tag is deleted, it is removed from all users to whom it is applied. However, tags that are

 currently in use by event rules cannot be deleted.

- **Event Recipient Tags:** When recipient tags are configured, notifications are sent to project members who have those tags. If no recipient tag is set, notifications are sent to all users in the project.

1. Click + **Add Tag** or the + icon.
2. Click + **Create New Tag** at the bottom of the **Event Recipient Tags** window.
3. In the **Create Tag** window, enter a tag name, select a color, and click the [Create Tag] button.
4. To edit or delete a created tag, click the  icon in the tag list.
5. Select the desired tag from the **Tag List** in the **Event Recipient Tags** window to apply it.

 When configuring events, selecting event recipient tags allows notifications to be sent to project members and third-party plugins associated with those tags. You can assign tags to project members and third-party plugins separately via **Alert > Notifications**.

Test Event Rules

You can test the configured conditions against historical data to estimate how many events would have been triggered.

1. Select the date for testing the event rules and click the [Confirm] button.
2. Click the [Run] button to check the number of triggered events.

Edit/Delete Event

1. Go to the **Alert > Notifications > Composite metrics** tab.
2. In the composite metrics list, click the  icon of the event you want to edit or delete.
3. In the **Notifications** screen, update the options and click the [Save] button.
 - a. To delete the selected event, click the [Delete] button in the upper right corner of the **Notifications** screen.

Event Sharing

You can save metric event settings as a JSON file to share configurations with other users or import settings created by others.

- JSON file name: `event-rules- YYYYMMDD .json`

Export

1. Click the [**JSON** ] button at the top right of the screen.
2. When the JSON editor opens, click the [ Export] button.
 - If you use the export function after searching for events, only the searched list will be downloaded as a JSON file.
3. After the JSON file is downloaded, share it with other users.

Import

1. Click the [] button at the top right of the screen.
2. Select the JSON file downloaded using the **Export** function.
3. When the JSON editor opens, choose either [Add to List] or [Overwrite].

 It is recommended to use this feature between projects of the same product type. You can import event settings from projects of different products, but they may not function correctly.

Edit in JSON Format

1. Click the [**JSON** ] button at the top right of the screen.
2. When the editor opens, modify the content according to the JSON format.
3. After completing the edits, click the [Save] button at the bottom of the screen.

 If the modified content does not conform to the JSON format, an error message will be displayed at the

! bottom of the screen and the content cannot be saved. The error message may vary depending on the type of formatting issue.

JSON error

Alert Message Settings

You can select detailed fields for the messages to be received and edit the content for each field.

Appendix

MXQL Syntax Guide

Composite metrics operates based on MXQL queries. For detailed syntax, refer to the [MXQL Syntax Guide](#).

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.

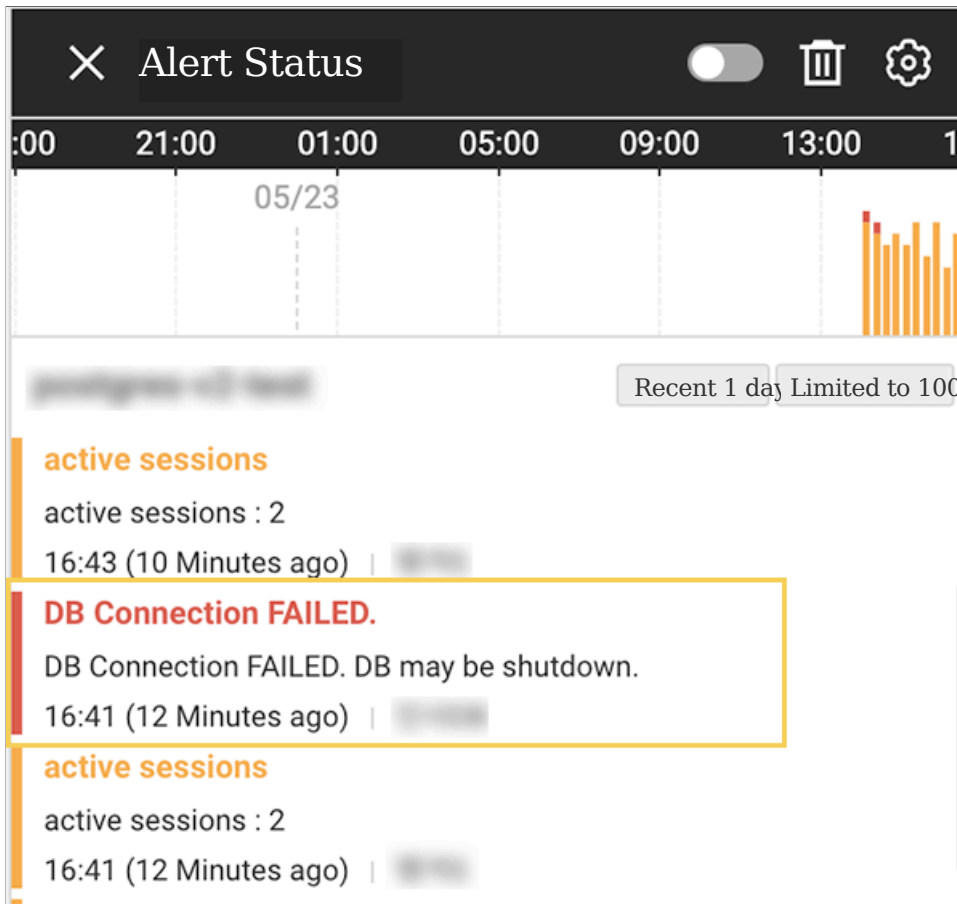
Inactive agent alert

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

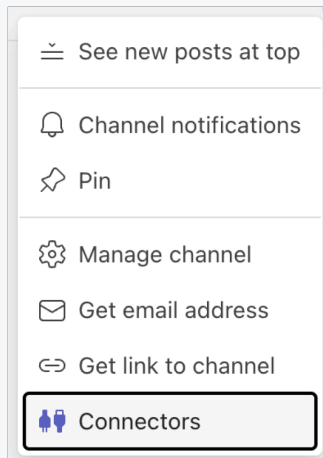


- `/remove` : Enter the project code for which you want to stop receiving alerts
- `/list` : View the list of project codes currently receiving alerts

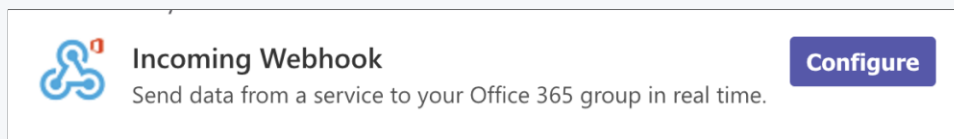
Teams



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

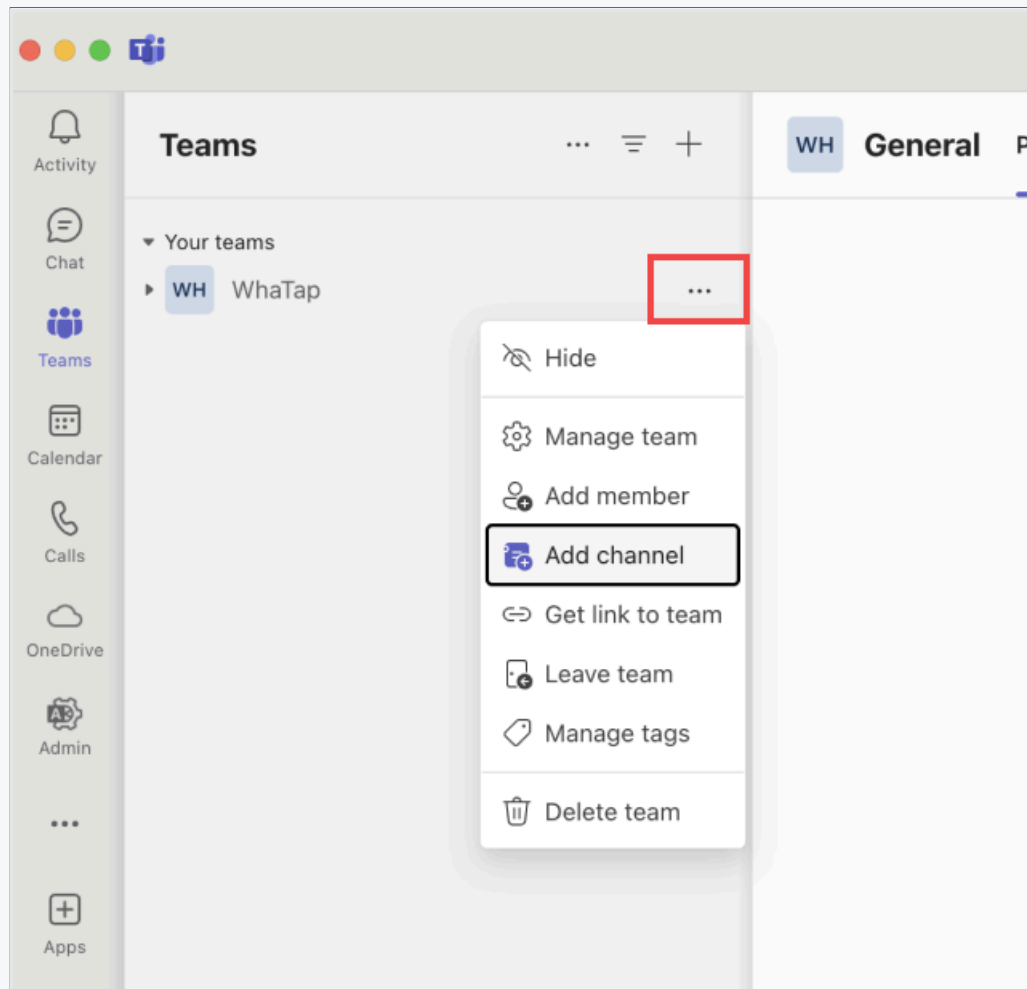
5. Return to the **Add a third-party plug-in** window in the WhaTap Monitoring service, and enter values for **Channel Name** and **Webhook URL** at the bottom of the **Teams** tab.
6. Click the [**Registration**] button.

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 * ⓘ

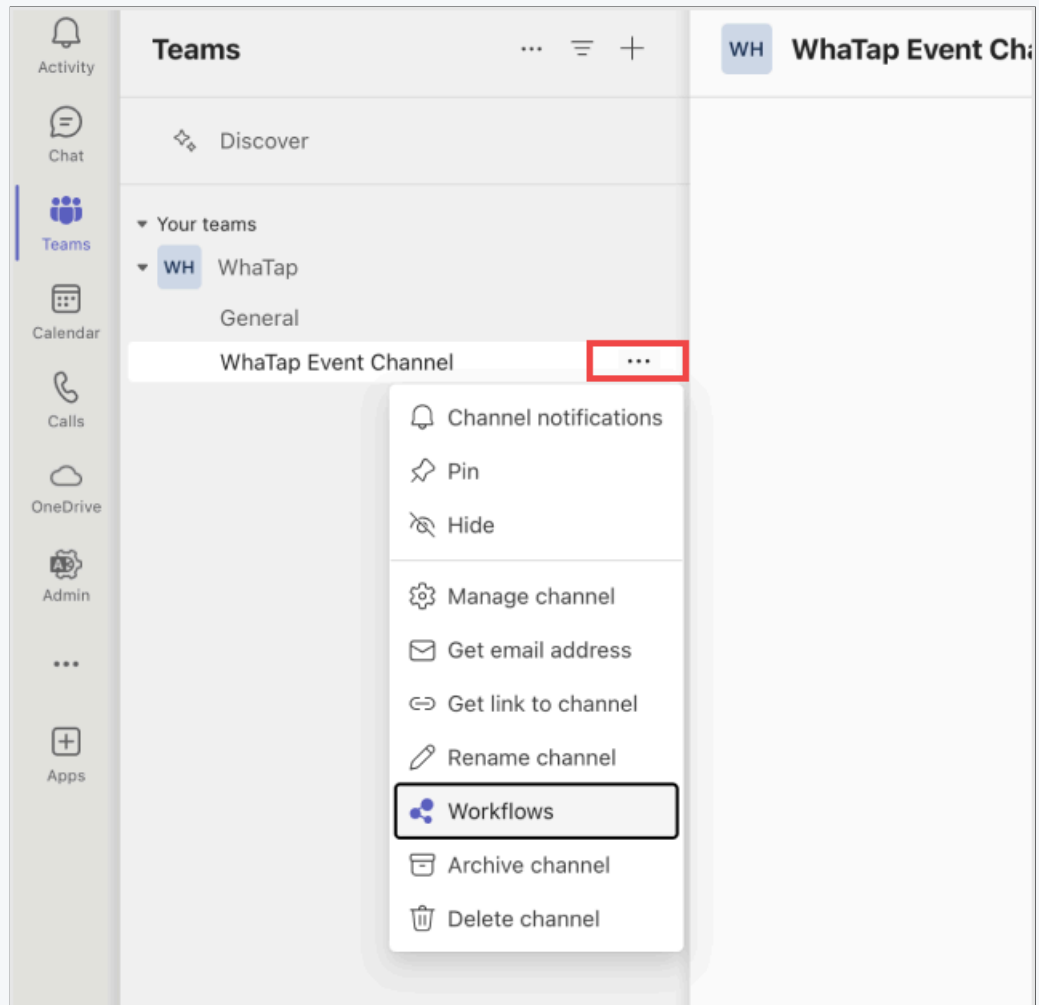
Select

- 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.

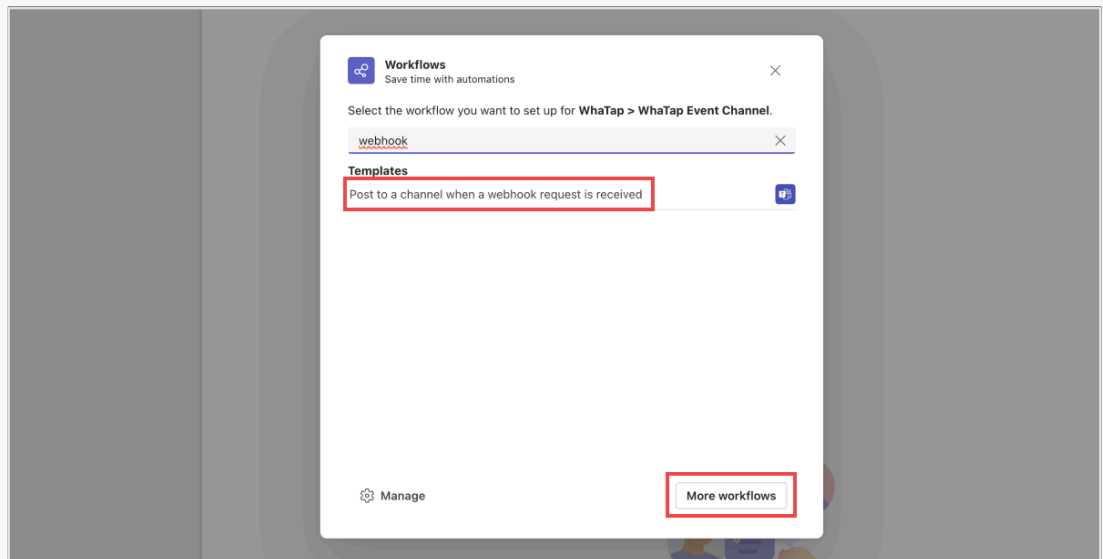
Cancel Create

ⓘ 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.

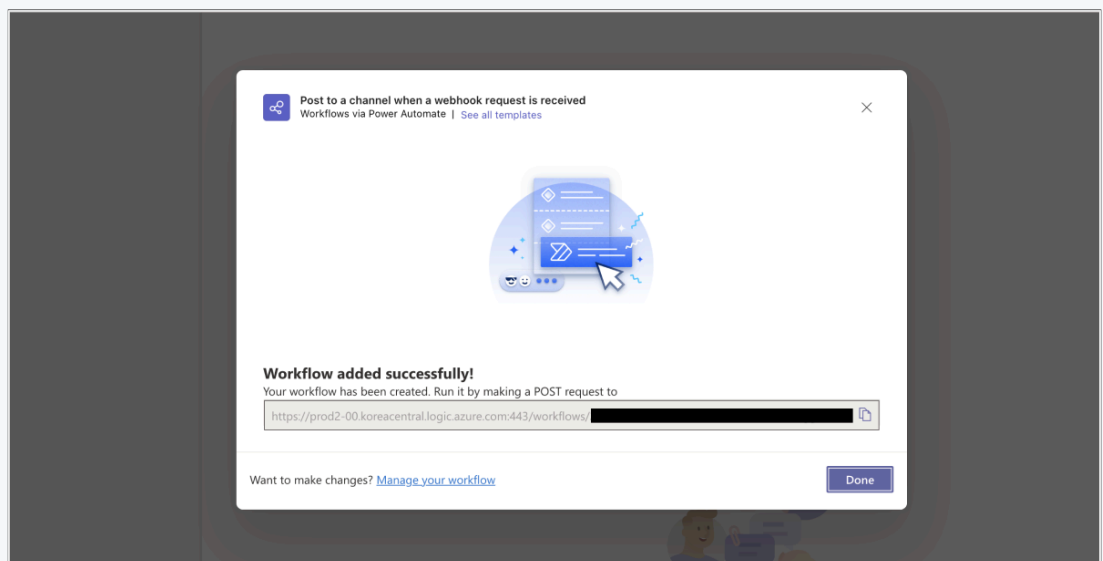
3. Click **More actions (...)** of the created channel, then select **Workflows**.



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.



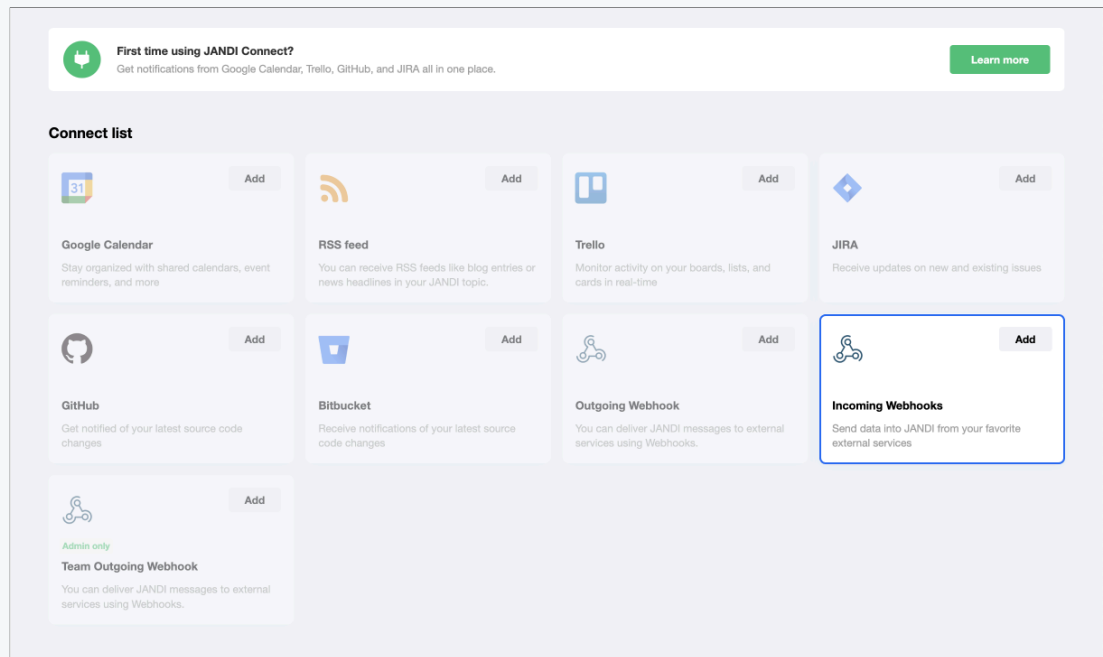
9. Click the [**Registration**] button.



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.

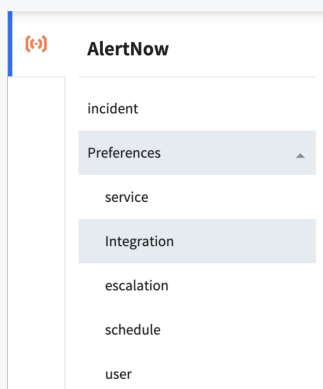
<https://wh.jandi.com/connect-api/webhook/31257583/d3...> Copy

- 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}",
}
```



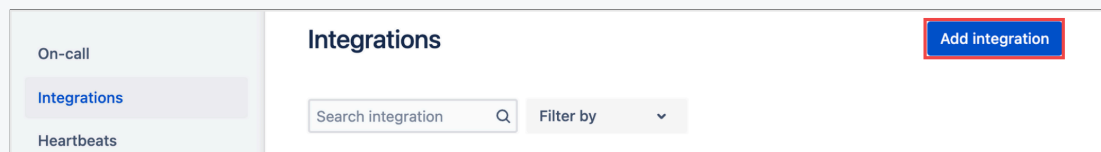
```
"metricName": "${metricName}",
"metricValue": "${metricValue}",
"metricThreshold": "${metricThreshold}",
"uuid": "${uuid}",
"okind": "${okind}",
"onode": "${onode}"
}
```

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.

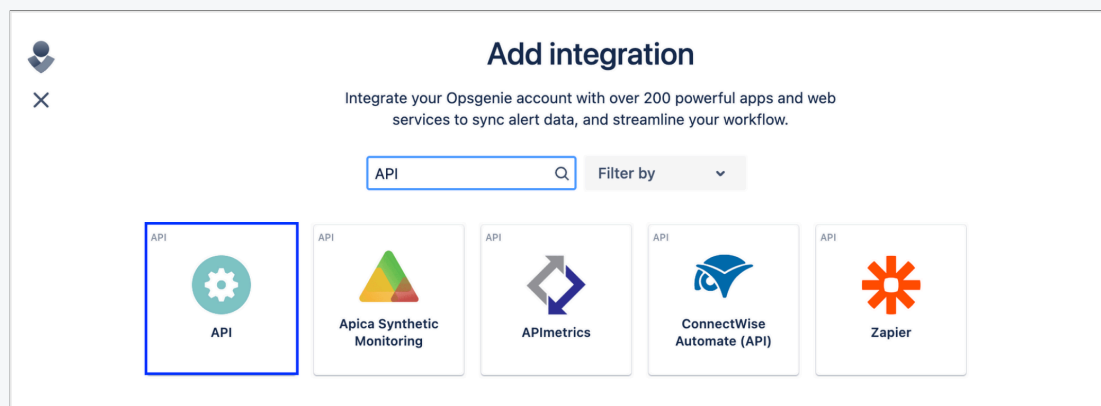
Opsgenie



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.

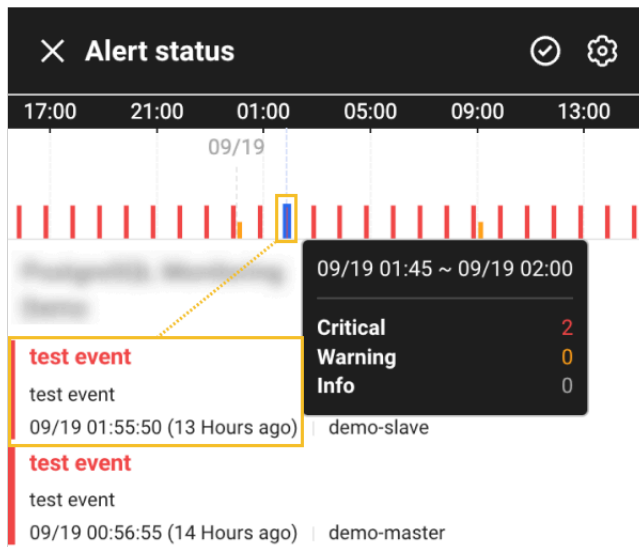
Viewing Real-time Notifications

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

☐ 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

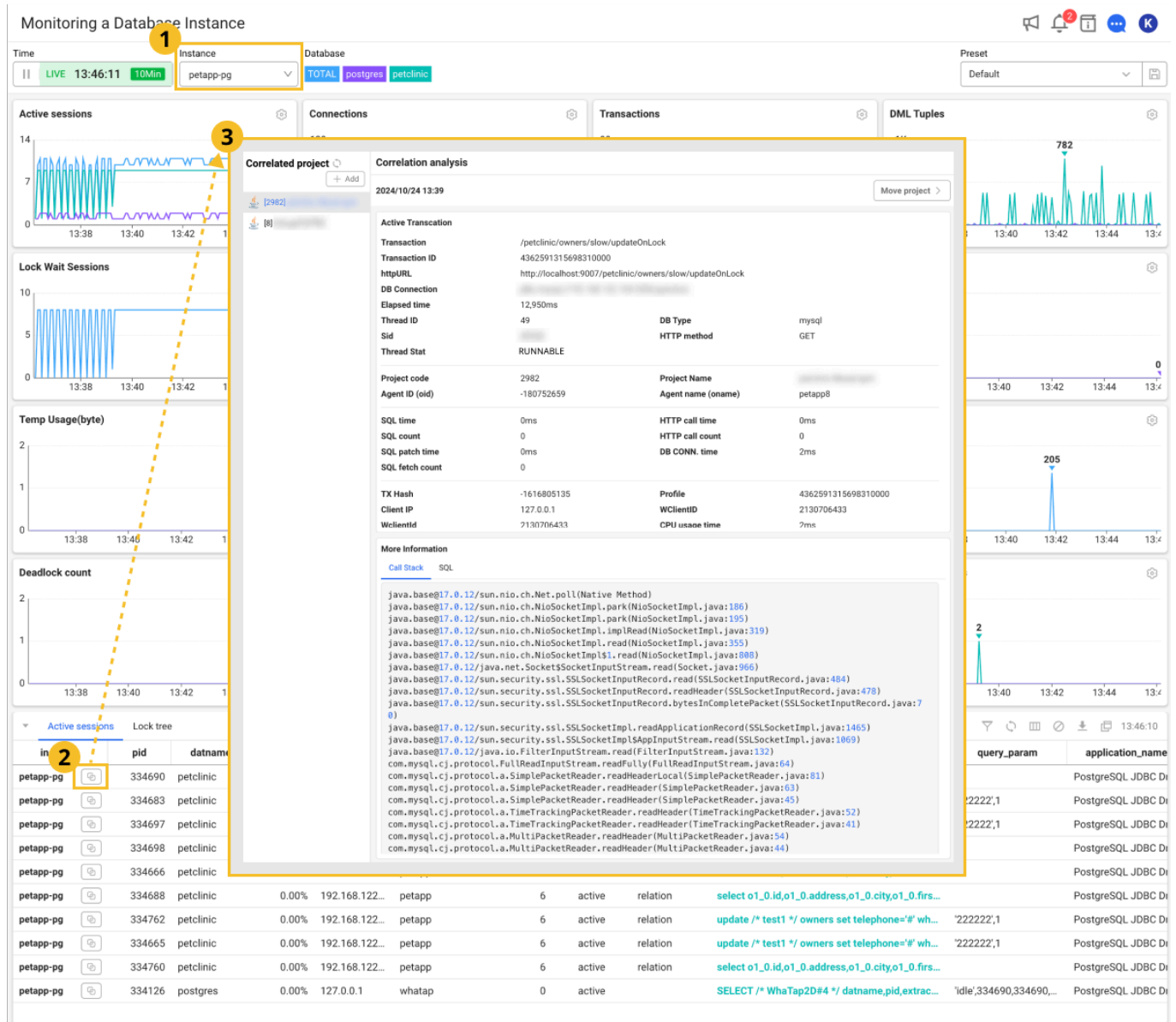
It provides real-time details about active transactions in applications associated to active sessions in the database. Users can check specific metrics such as transaction ID, thread ID, client IP, and CPU usage time, along with basic information such as transaction URL and database connection information, 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 - MySQL agent
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.

① **Sid** is the same value as the **PID** entry in the **Active sessions** table. It is also equivalent to the **SID** entry in the SQL information of the APM project.



Active sessions		Lock tree	Process Info			
instance		pid	datname	cpuusage	client_addr	username
petapp-pg		414200	petclinic	0.00%	192.168.122...	petapp
petapp-pg		414233	petclinic	0.00%	192.168.122...	petapp
petapp-pg		414285	petclinic	0.00%	192.168.122...	petapp

Correlation analysis

2024/10/24 13:39

Move project >

Active Transcation

Transaction

/petclinic/owners/slow/updateOnLock

Transaction ID

4362591315698310000

httpURL

http://192.168.122.1:5050/petclinic/owners/slow/updateOnLock

DB Connection

jdbc:mysql://192.168.122.104:3306/petclinic

Elapsed time

12,950ms

Thread ID

49

DB Type

PostgreSQL

Sid

414200

HTTP method

GET

Thread Stat

RUNNABLE

Project code

2982

Project Name

petclinic-MySQL-agent

Agent ID (oid)

-180752659

Agent name (oname)

petapp8

SQL

Connection URL

jdbc:mysql://192.168.122.104:3306/petclinic

CRUD

Select

owners

Query (SQL hash: 1083464189)

Correlated DB session

Password

SQL format

Statistics >

SID

414200

```
select c1_b.id,c1_b.address,c1_b.city,c1_b.first_name,c1_b.last_name,g1_b.owner_id,g1_b.id,g1_b.birth_date,g1_b.name,g1_b.type_id,c1_b.telephone from owners c1_b left join g1_b on c1_b.id=g1_b.owner_id where c1_b.id=1 order by g1_b.name
```

Variables mapping

Bind variables

Copy bind variables

Bind variables : 1

- **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.