

Oracle Pro Monitoring

release date. 2026. 09. 11.



Table of Contents

• Oracle Proモニタリング	5
◦ はじめに.....	7
◦ サポート環境.....	9
◦ インストール	
▪ エージェントのインストール	14
▪ インストールチェックリスト	27
▪ インストール問題の解決	29
◦ 設定	31
▪ DMXエージェント設定.....	33
▪ XOSエージェント設定	43
◦ 管理.....	46
◦ ダッシュボード	47
▪ インスタンスリスト	49
▪ インスタンスマップ.....	51
▪ インスタンスモニタリング.....	60
▪ マルチインスタンス	90
▪ RACモニタリング	116
▪ Flexボード	145
▪ Flexボードの使用方法.....	148
▪ ウィジェットの管理.....	159
◦ 分析	
▪ パフォーマンス比較	
▪ 推移の比較	164
▪ Top SQL比較	175

▪ ワークロード分析	
▪ Wait分析.....	179
▪ Wait別 Top SQL.....	183
▪ キューブ	188
▪ セッション分析	
▪ ロックツリー.....	191
▪ 並列クエリツリー.....	203
▪ セッション履歴	211
▪ SQL分析	
▪ Top SQL	213
▪ SQL Plan 分析	223
▪ メトリクス分析	
▪ メトリクスチャート	228
▪ メトリクストレンド分析.....	251
◦ メトリクス	254
▪ Oracle Proメトリクス	256
▪ メトリクスチャート	266
▪ メトリクス照会.....	269
▪ メトリクス異常検出	271
◦ 統計/レポート	
▪ テーブルスペースサイズ	274
▪ SGAサイズ	284
▪ データベースパラメータ	286
▪ レポート	288
▪ SQL統計.....	291
◦ ログ	296

▪ 適用方法	298
▪ ライブテール	300
▪ ログエクスプローラー	312
▪ ログトレンド	319
▪ ログの検索.....	327
▪ ログの設定.....	341
▪ ログパーシング.....	377
◦ アラート通知.....	392
▪ イベント設定	0
▪ メトリクス	399
▪ 複合メトリクス	420
▪ 異常値.....	427
▪ リアルタイムログ.....	431
▪ 複合ログ	443
▪ イベント設定	457
▪ 通知設定	461
▪ イベント通知フォーマット	483
▪ イベント履歴	491
▪ リアルタイムアラート	499
▪ システムイベント.....	502
◦ 管理	
▪ 連携プロジェクト管理.....	504
◦ 実験室	511
▪ OpenMetricsエクスプローラー	512

Oracle Proモニタリング

データベースのパフォーマンス管理は、データベースのパフォーマンスをリアルタイムでモニタリングし、問題点を診断、分析します。問題のあるSQLをチューニングして適用し、問題が解決されたかをリアルタイムでモニタリングします。

WhaTapデータベースモニタリングサービスは、これら一連のタスクを循環管理できるように設計されています。データベースの動作をモニタリング、診断、分析し、可用性とパフォーマンスを一定のレベルに維持できるように構成されています。データベースに問題が発生した場合、担当者にアラームを送信して状況を認識し、収集されたデータからデータベースの状況をよりに分析することで、Root Causeを明確にする支援サービスです。

❗ データベースプラットフォームごとの対応機能の詳細については、[次の文書](#)を参照してください。

Oracle Proモニタリングの特長

WhaTapのOracle Proモニタリング製品は、従来のクエリ(Query)ベースのデータ収集方式を一段階上回る革新を提示します。既存のクエリベースの方法は、データベースサーバーにかなりの負荷を与え、これはシステム性能の低下につながる可能性があります。一方、Direct Memory Access(DMA) 技術を活用したWhaTapのアプローチは、サーバーのメモリに直接アクセスして必要なデータを抽出することによって、データベースサーバーに負荷をほとんど、または全く発生させません。これは、特に高性能を必要とするデータベース環境において大きな利点として機能します。

また、クエリベースの収集方式はデータ更新サイクルが遅く、特定のクエリ実行時点にのみデータを収集できます。これに比べ、Oracle Proモニタリングはメモリへのアクセスを通じてデータをほぼリアルタイムで収集でき、少なくとも1/20秒の間隔でセッション及びSQL性能データを収集できます。これにより、データベース管理者はシステムの性能をより詳細にモニタリングし、問題が発生したときにすぐに対応できるようになります。

Oracle Proモニタリング製品のもう一つの重要な特徴は、システム障害やHang状況でもデータ収集が中断されないことです。従来のクエリベースの方法では、これらのシナリオではデータの収集が困難になる可能性があります。Oracle Proを使用することで、データベース管理者はいかなる状況でもデータの連続性を保証され、これによってシステムの安定性と可用性を最大化できます。

結局、WhaTapのOracle Proモニタリング製品は、データベースモニタリングの効率性と正確性を向上させると同時に、システム負荷を最小限に抑え、ビジネス継続性を保証するために欠かせない技術です。これは、企業がより速く、より正確で、より信頼性の高い意思決定ができる基盤を提供します。

- **メモリのアクセス方式**

WhaTapのOracle Proは**Direct Memory Access(DMA)** 技術を通じてDBサーバーメモリに直接アクセスし、データを収集します。既存のクエリ方式の限界を超えて、サーバーの負荷なしで迅速かつ効率的なデータ収集を提供します。

- **負荷のないモニタリング情報の収集**

Oracle ProはDBサーバーメモリに直接アクセスしてデータを抽出し、1秒に20回の性能情報を収集してもサーバー負荷をほとんど発生させません。これは高性能DB環境でも安定的かつ効率的なモニタリングをサポートします。

- **精密なSQL性能データの収集**

Oracle Proは1秒に20回データを収集し、精密な分析が可能であり、従来のクエリーベース方式の5秒間隔SQL収集よりもはるかに詳細な情報を提供します。これにより、管理者はリアルタイムで問題を特定し、迅速に対応できます。

- **DB Hang状態でもモニタリング分析が可能**

DB障害やHang状況でもOracle Proはデータ収集が中断されません。これにより、データの連続性が保証され、障害のある状況でも分析および対応が可能になり、システムの安定性と可用性が最大化されます。

はじめに

WhaTapモニタリングは次の手順で開始します。

[ステップ1. 会員登録](#)

[ステップ2. 製品選択](#)

[ステップ3. プロジェクト作成](#)

[ステップ4. アクセスキー確認](#)

その後、サポート環境を確認してからエージェントのインストール手順に進みます。

会員登録

WhaTapモニタリングサービスを利用するには、まず[会員登録](#)が必要です。会員登録に関する詳細は、[アカウント管理 > 会員登録](#)ドキュメントを参照してください。

製品選択

1. [WhaTapモニタリングサービス](#)にログインします。
2. ≡ [プロジェクト全体](#) > [+ Project](#)ボタンをクリックします。
3. [商品選択](#)画面でモニタリングするGo製品を選択してから作成ボタンをクリックしてください。

プロジェクト作成

モニタリングする製品を選択したら、エージェントをインストールする前にまずプロジェクトを作成します。

1. [プロジェクト作成](#)画面でプロジェクトを作成するには、次の項目を入力または選択してください。
 - [プロジェクト名](#)(必須): プロジェクト名を入力します。
 - [データサーバーリージョン](#)(必須): データサーバーが位置するリージョン(地域)を選択します。リージョンはクラウドサービスを提供するデータセンターのグループです。選択したリージョンにユーザーのデータが保存されます。

- **プロジェクトタイムゾーン(必須):** アラートおよびレポート生成時の基準となる時間を設定します。
- **通知言語の設定(必須):** プロジェクトで発生する警告通知メッセージの言語を設定します(韓国語、英語対応)。
- **プロジェクトグループ:** 複数のプロジェクトをグループにまとめて管理できます。所属するグループがあれば選択してください。
- **プロジェクト説明:** プロジェクトに関する追加説明を入力します。

2. すべての設定を完了したら**プロジェクトを作成する**ボタンをクリックしてください。

❗ 組織を選択した状態でプロジェクトを追加する場合、**組織下位のグループ**を必須で設定する必要があります。グループに関する詳細は、[次のドキュメント](#)を参照してください。

アクセスキー確認

アクセスキーは、WhaTapサービスを有効化するための固有IDです。

インストールガイドセクションで**プロジェクトアクセスキーの発行**ボタンを選択してください。アクセスキーを自動的に発行した後、次のステップに進みます。

❗ プロジェクトを作成すると、自動的に**エージェントインストール**ページに移動します。エージェントインストールページに移動しない場合は、画面左側のメニューから**管理 > エージェントインストール**を選択してください。

サポート環境

Oracle Proモニタリングを開始する前に、次のサポート環境を確認してください。

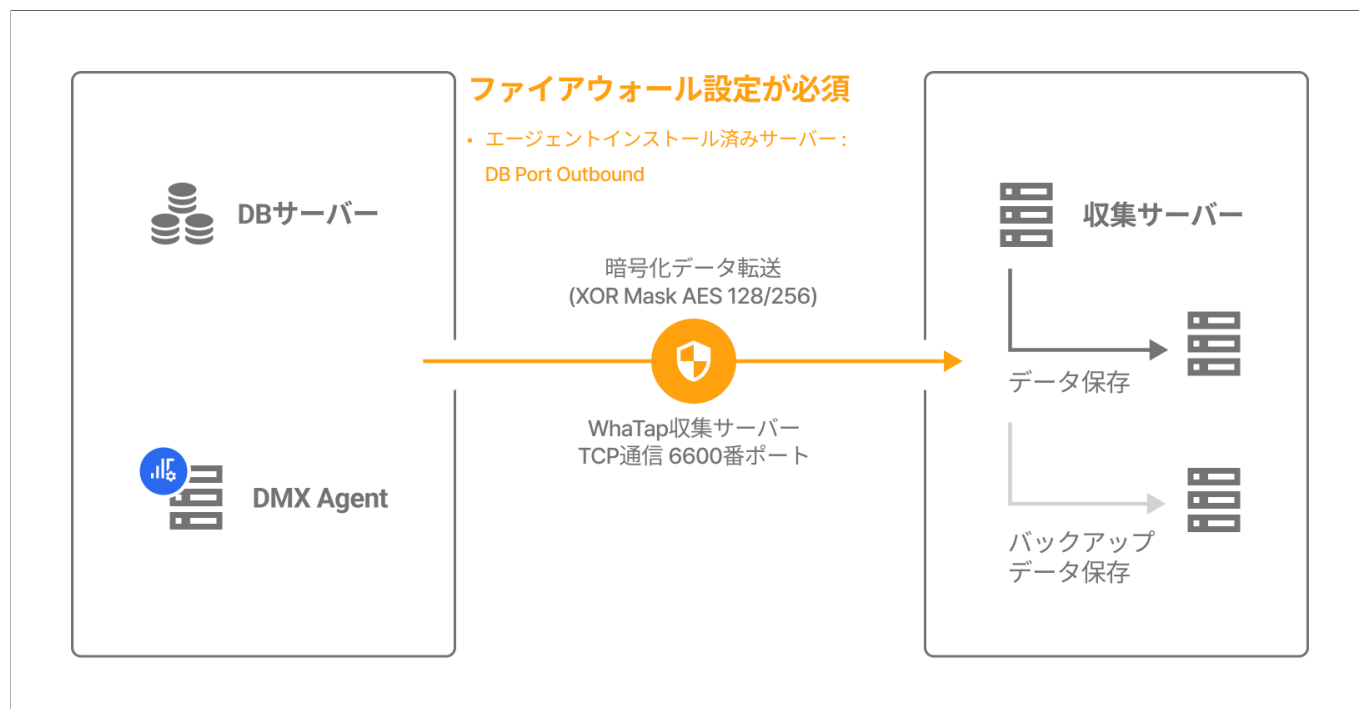
データベース対応バージョン

Linuxサーバー、Oracle 11gR2以降に対応します。

ⓘ DBXエージェントがWhaTapサーバーにデータを転送できるように、6600ポートのOutboundを開く必要があります。

エージェント構成

Oracle Proのモニタリングは、データベースのサーバーにエージェントを直接インストールしてモニタリングします。WhaTapの他のデータベース商品と異なりXOSエージェントのインストールなしでXOS指標を収集できます。



サービス提供形態

WhaTapモニタリングは、SaaSまたはオンプレミス（インストール型）方式でご利用いただけます。どちらを選択しても、同じエージェントで同じモニタリング機能をご利用いただけます。

Deployment	Support	Description
SaaS	✓	WhaTapが運用する収集サーバーに接続。リージョン別アドレスはファイアウォール項目を参照
オンプレミス（インストール型）	✓	収集サーバーを自社で構築して運用（閉域網環境に対応）。導入範囲は営業担当者と協議

ⓘ インターネットに接続しない閉域網環境でも、オンプレミスで構築できます。

提供形態による機能の違い

どの形態を選択しても、モニタリング機能は同じです。モバイルアプリのみが例外で、service.whatap.io SaaSでのみご利用いただけます。

① オンプレミス環境で一部の機能をご利用いただくには、別途契約が必要です。

ファイアウォール

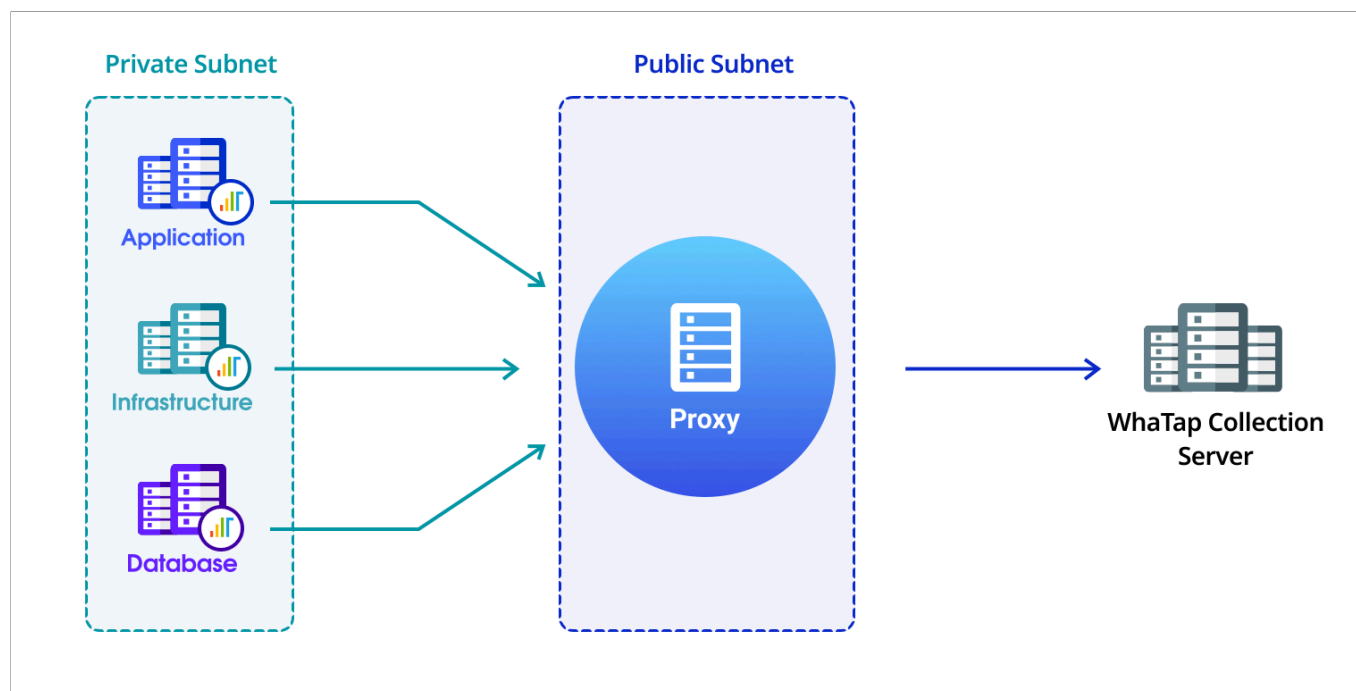
WhaTap エージェントは、収集サーバーの **TCP 6600** ポートにアクセスする必要があります。監視対象に最も近い収集サーバーのアドレスを許可してください。

出発地：WhaTapエージェント

クラウド	目的地	目的地IP	ポート番号
AWS	WhaTapソウル収集サーバー	13.124.11.223 / 13.209.172.35	TCP 6600
	WhaTap東京収集サーバー	52.68.36.166 / 52.193.60.176	TCP 6600
	WhaTapシンガポール収集サーバー	18.138.0.93 / 18.139.67.236	TCP 6600
	WhaTapジャカルタ収集サーバー	108.136.91.69 / 108.137.158.44	TCP 6600
	WhaTapタイ収集サーバー	43.209.109.219 / 43.209.48.86	TCP 6600
	WhaTapムンバイ収集サーバー	13.127.125.69 / 13.235.15.118	TCP 6600
	WhaTapカリフォルニア収集サーバー	52.8.223.130 / 52.8.239.99	TCP 6600
	WhaTapバージニア収集サーバー	107.23.220.101 / 54.236.221.105	TCP 6600
	WhaTapフランクフルト収集サーバー	3.125.142.162 / 3.127.76.140	TCP 6600

クラウド	目的地	目的地IP	ポート番号
Azure	WhaTapソウル収集サーバー	52.231.66.38 / 20.194.5.115	TCP 6600
	WhaTap東京収集サーバー	52.246.169.54 / 20.210.27.232	TCP 6600
Kakao	WhaTapソウル収集サーバー	61.109.237.237 / 61.109.238.166	TCP 6600

エージェントから収集サーバーに直接接続できない場合は、提供するProxyモジュールを使用して経由します。



対応ブラウザ

WhaTapモニタリングサービスは、ウェブブラウザとモバイルアプリで利用できます。

Webブラウザ	推奨	対応バージョン
Google Chrome	0	111以降

Webブラウザ	推奨	対応バージョン
Mozilla FireFox	X	最新バージョン
Edge	X	最新バージョン
Safari	X	最新バージョン

❗ サポート案内

- ブラウザの互換性および性能の観点から、Chrome の最新バージョンの使用を推奨します。
- ユーザーインターフェース（User Interface, UI）は HTML5 標準技術で実装されているため、Internet Explorer はサポートしていません。

❗ 制約事項

WhaTap の Web インターフェースはモバイルブラウザをサポートしていません。モバイル端末から WhaTap にアクセスするには、Android または iOS 用の WhaTap アプリをインストールしてください。WhaTap モバイルアプリはモバイル環境に最適化されています。WhaTap モバイルアプリの詳細については、[こちらのドキュメント](#) を参照してください。

エージェントのインストール

WhaTapデータベースモニタリングサービスを使用するための基本的なインストール方法について説明します。

1 プロジェクト作成

エージェントをインストールする前にプロジェクトを作成してください。

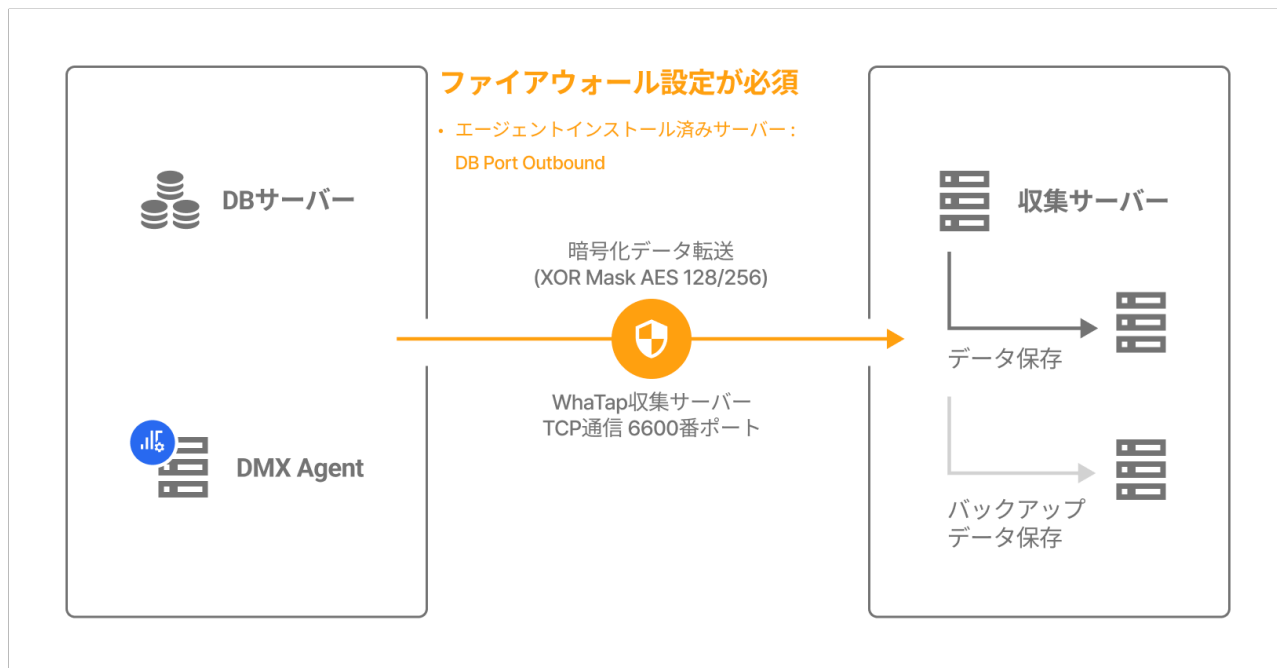
1. [WhaTapモニタリングサービス](#)にログインしてください。
2. 左側のサイドメニューで **プロジェクト全体** > **+ Project** ボタンをクリックします。
3. **商品選択** 画面でインストールする製品を選択します。
4. 以下の項目を入力または選択します。
 - **プロジェクト名**：プロジェクト名を入力します。
 - **データサーバーリージョン**：データサーバーの設置されているリージョンを選択します。リージョンとは、クラウドサービスを提供するデータセンターの集合体です。選択したリージョンにユーザーデータが保存されます。
 - **プロジェクトタイムゾーン**：アラートおよびレポート生成時の基準となる時刻を設定します。
 - **通知言語の設定**：アラート通知メッセージの言語を設定します。（韓国語、英語に対応）
 - **プロジェクトグループ**：複数のプロジェクトをグループとしてまとめて管理できます。所属するグループがある場合は選択してください。
 - **プロジェクト説明**：プロジェクトに関する補足説明や詳細情報を入力します。
5. すべての設定が完了したら、**プロジェクトを作成する** ボタンをクリックします。

❗ 組織を選択してプロジェクトを追加する場合は、**組織下位のグループ**を必須に設定する必要があります。
グループに関する詳細は、[次のドキュメント](#)を参照してください。

2

構成図の確認

Oracle Proモニタリングのためにデータベースサーバーにエージェントをインストールする構成です。



ファイアウォールの設定は必須であり、すべてのデータは暗号化されて送信されます。WhaTap収集サーバーとの通信には6600ポートを使用します。

3

アカウント生成

データベースモニタリングをするために必要な権限を持つアカウントを作成します。root(admin) アカウントでログインしてアカウントを作成してください。

- 一般的な場合

SQL

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

- マルチテナントDB(Oracle 12c以降)で基本モニタリングを使用する場合: c## アカウント作成

SQL

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

- kill sessionを使用する場合

SQL

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

- ヒドゥンパラメータを照会する場合

SQL

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

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

権限による対応機能を確認する



基本権限外の権限は、ユーザーの環境によって付与する必要はありません。ただし、一部の機能が動作しないことがあります。

権限	command (Single DB)	command (マルチテナント DB)	メニューと機能	詳細履歴
基本	grant create session	grant create session	◇◇	-

!				
権限	command (Single DB)	command (マルチテナント DB)	メニューと機能	詳細履歴
権限	to whatap;	to c##whatap;		
	grant select any dictionary to whatap;	grant select any dictionary to c##whatap;		個別権限の付与時、次の表の内容を参照
	-	alter user c##whatap set container_data = ALL container = CURRENT;	インスタンスモニタリング / マルチインスタンス / SQL統計	マルチテナントDBを使用する場合(12C以降) pdb照会のための必須権限である
セッション中止	grant alter system to whatap;	grant alter system to c##whatap container=all;	アクティブセッション>セッション中止	-
ヒドウンパラメータの照会	create view sys.whatap_vw_ksppi as select * from sys.x\$ksppi; grant select on sys.whatap_vw_ksppi to DB_User; create synonym DB_User.whatap_vw_ksppi for sys.whatap_vw_ksppi; create view sys.whatap_vw_ksppcv as select		パラメータ照会メニューにヒドウンパラメータの内容を含む	ヒドウンパラメータを照会するには、別のビュー/シノニムの生成および権限付与が必要



権限	command (Single DB)	command (マルチテナント DB)	メニューと機能	詳細履歴
	<pre>* from sys.x\$kspcv;</pre> <pre>grant select on sys.whatap_vw_kspcv to DB_User;</pre> <pre>create synonym DB_User.whatap_vw_kspcv for sys.whatap_vw_kspcv;</pre>			

select any dictionary の代わりに、次の権限を個別に付与しても構いません。

権限	command (Single DB)	command (マルチテナント DB)
アクティブセッション / ロックツリー / メトリクス情報	grant select on v_\$sqlarea to whatap;	grant select on v_\$sqlarea to c##whatap;
	grant select on v_\$transaction to whatap;	grant select on v_\$transaction to c##whatap;
	grant select on v_\$instance to whatap;	grant select on v_\$instance to c##whatap;
	grant select on v_\$sysstat to whatap;	grant select on v_\$sysstat to c##whatap;
	grant select on v_\$lock to whatap;	grant select on v_\$lock to c##whatap;
	grant select on v_\$sesstat to	grant select on v_\$sesstat to



権限	command (Single DB)	command (マルチテナント DB)
	<code>whatap;</code>	<code>c##whatap;</code>
	<code>grant select on v_\$event_name to whatap;</code>	<code>grant select on v_\$event_name to c##whatap;</code>
	<code>grant select on v_\$session to whatap;</code>	<code>grant select on v_\$session to c##whatap;</code>
	<code>grant select on v_\$sql to whatap;</code>	<code>grant select on v_\$sql to c##whatap;</code>
	<code>grant select on v_\$process to whatap;</code>	<code>grant select on v_\$process to c##whatap;</code>
	<code>grant select on v_\$statname to whatap;</code>	<code>grant select on v_\$statname to c##whatap;</code>
	<code>grant select on v_\$system_event to whatap;</code>	<code>grant select on v_\$system_event to c##whatap;</code>
	<code>grant select on v_\$osstat to whatap;</code>	<code>grant select on v_\$osstat to c##whatap;</code>
	<code>grant select on v_\$px_session to whatap;</code>	<code>grant select on v_\$px_session to c##whatap;</code>
	<code>grant select on v_\$sga to whatap;</code>	<code>grant select on v_\$sga to c##whatap;</code>



権限	command (Single DB)	command (マルチテナント DB)
	-	grant select on v_\$con_sysstat to c##whatap;
パラメータ情報	grant select on v_\$parameter to whatap;	grant select on v_\$parameter to c##whatap;
Wait情報/Event情報	grant select on v_\$event_name to whatap;	grant select on v_\$event_name to c##whatap;
	grant select on v_\$system_event to whatap	grant select on v_\$con_system_event to c##whatap;
	grant select on v_\$system_wait_class to whatap;	grant select on v_\$system_wait_class to c##whatap;
	-	grant select on v_\$con_system_wait_class to c##whatap;
Tablespace情報	grant select on dba_data_files to whatap;	grant select on cdb_data_files to c##whatap;
	grant select on dba_free_space to whatap;	grant select on cdb_free_space to c##whatap;



権限	command (Single DB)	command (マルチテナント DB)
	<code>grant select on dba_temp_files to whatap;</code>	<code>grant select on v_\$sort_segment to c##whatap;</code>
	<code>grant select on v_\$temp_extent_pool to whatap;</code>	<code>grant select on v_\$tempfile to c##whatap;</code>
	<code>grant select on v_\$containers to whatap; (12c以上)</code>	<code>grant select on v_\$containers to c##whatap;</code>



- 既に存在するアカウントを使用する場合は、[DBユーザーファイルの作成](#)に進みます。権限がない場合は、モニタリングを開始できません。
- コード例で `whatap` は、DBユーザーアカウント名です。ユーザーが利用するユーザーアカウント名に変更してください。
- コード例の `DB_Password` には、ユーザーパスワードを入力します。

4

アクセスキーの確認

アクセスキーは、WhaTapサービスを有効にするための固有IDです。

インストール手順セクションで[プロジェクトアクセスキーの発行](#)ボタンをクリックします。アクセスキーを自動的に発行した後、次のステップを進みます。



- プロジェクトを作成すると、自動的に[エージェントインストールページ](#)に移動します。エージェントインストールページが移動しない場合は、画面左側のメニューから[管理](#) > [エージェントインストール](#)

❗ を選択します。

5 エージェントダウンロード

エージェントのインストール画面で、ユーザーの環境と一致するOracleバージョンを選択し**ダウンロード**ボタンを選択します。

❗ セキュリティ設定によりTAR形式のファイルをダウンロードできないユーザーのためにZIP形式のファイルも一緒に提供します。インストール画面で**.zip ダウンロード**ボタンを選択します。

6 エージェント設定

エージェントはOracleアカウントでのインストールを推奨し、Oracle接続環境変数が事前に設定されている必要があります。以下の手順でエージェント設定を進めてください。

❗ Oracleアカウント以外のユーザーアカウントからインストールする場合

- Oracleこの生成した共有メモリ(shared memory)の読み込み権限(read)が必要です。
- `ipcs -m` コマンドを実行した結果で `perms` 値が**640**の場合、同じグループに属する別のアカウントでエージェントをインストールできます。
- `ipcs -m` コマンドを実行した結果で `perms` 値が**600**でありながら、**Oracle 12c R2**以降であれば、次のようにパラメータを設定し、`perms`値を**640**に変更できます。変更事項を適用するには、DBを再起動する必要があります。

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

whatap.conf設定

圧縮を解除したフォルダに移動し、**whatap.conf**ファイルを確認します。**whatap.conf** ファイルに、プロジェクトのアクセスキー情報、WhaTapサーバー情報、**ipckey**を入力します。

whatap.conf

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

- ✔ **ipckey** オプション値は、**ipcs -m** コマンドを使用して取得できます。モニタリングするOracleのKey値(16進数)です。
- 1つのサーバーに複数のデータベースをインストールした場合は、[次の文書](#)を参照して **ipckey** を入力します。

初期設定ファイル作成

初期設定ファイルを作成するには、データベースに**SYS**アカウント(または同ランクのアカウント)にアクセスして情報を受け取り、処理します。アカウントとパスワードは保存されません。

BASH

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

コマンドを実行した後、次の5つのファイルが作成されます。

dmx.init、**class.init**、**dbname.init**、**stat.init**、**event.init**

- ❗ **sys_user** と **sys_password** なしで接続できるのであれば(**sqlplus '/as sysdba'** 接続のように)、**sys_user** と **sys_password** を引数として追加する必要はありません。



BASH

```
./dmxinit.sh
```

- `sys_user` と `sys_password` 文字列に特殊文字が含まれている場合は、各特殊文字の前にエスケープ文字(Escape Character、`\`)を追加します。

BASH

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

DBユーザーの作成とパスワードの変更

データベース接続用に暗号化されたUIDを作成します。usernameとpasswordを入力し、シェルスクリプト(またはバッチファイル)を実行してください。



パスワードを変更する場合も、変更するパスワードを入力して同じ方法で進めてください。

BASH

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



- 一度の設定で暗号化されたUIDを使用してモニタリング対象データベースサーバーからデータを収集します。
- DBユーザーファイルを生成するには、プロジェクトアクセスキーを`whatap.conf`ファイルに入力する必要があります。[アクセスキーの確認](#)
- Azure Database環境の場合、`DB_USER` を`DB_USER@DB_name`の形式で入力します。
- `DB_USER` または `DB_PASSWORD` に特殊文字が含まれている場合、特殊文字の前にEscape



Character(\)を入力します。

Example

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

特殊文字が2つ以上の場合、それぞれエスケープ文字を追加

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

7

モニタリングの開始

エージェントをインストールしたパスからシェルスクリプト(またはバッチファイル)を実行してください。

```
./start.sh
```

デーモンのように使用する場合、次のコマンドを実行してください。ただし、**nohup**をインストールした環境のみで動作します。

```
./startd.sh
```

次のステップ

インストールの点検

プロジェクトの作成やエージェントのインストール、基本エージェントのオプションまですべて適用した場合は、[次の文書](#)で点検事項を確認します。

インストール問題の解決

エージェントをインストールする際に発生する可能性がある問題と、これを解決するための具体的なガイドを提供します。詳細については、[次の文書](#)を参照してください。

エージェント設定

エージェント設定([whatap.conf](#))ファイルにオプションを適用して、モニタリングのための様々な機能を提供します。詳細については、[次の文書](#)を参照してください。

モニタリングの開始

すべての設定を完了したら、エージェントがデータベースサーバーのメトリクスに対する情報収集を開始します。まず、**インスタンスリスト**メニューでモニタリングデータが収集されていることを確認します。**インスタンスリスト**の詳細については、[次の文書](#)を参照してください。

インストールチェックリスト

データベースモニタリングのインストールがすべて完了しました。インストール後、次の事項が正しくインストールされているか確認します。

エージェントのインストールチェック

データベースエージェントをインストールしたパスから、[start.sh](#) シェルスクリプトファイルまたは [start.bat](#) バッチファイルを実行します。エージェントを実行すると、データベースのモニタリング情報の収集が開始されます。シェルスクリプトを実行した後、ターミナルにWhaTapロゴが出力されている場合は、正常に実行されている状態です。

start.sh

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

```
 _ _ _ _ _  
|||/|/|/ _ _/_ _/_ _ _  
|||/_ _V_`|/|/|/_`/_ _\  
|_|/_|/|/|/_|/_|/_|/_|/_|  
      |/_|
```

Just Tap, Always Monitoring

WhaTap DBX version X.Y.Z YYYYMMDD

❗ デーモンのように使用したい場合は、`startd.sh` コマンドを実行してください。ただし、`nohup`をインストールされている環境のみで動作します。

❗ DMXエージェントを実行すると、`dmx`、`prx`プロセスが実行されます。

ログの確認

- [whatap.log](#)

エージェントの起動および収集に関するログはすべて、`$WHATAP_HOME /logs/whatap.log`パスで確認できます。
`whatap.log`ファイルのサイズが25MBを超える場合は、`whatap.log.[n]`ファイル名でバックアップします。ファイルをクリーンアップする必要がある場合は、番号(n)が大きい(古い)ファイルから削除してください。

- `prx.log`

DMA(Direct Memory Access)基盤となるデータ収集で発生する主要なアクティビティとイベントを記録するログファイルです。DMXエージェントを再起動すると、ログを記録します。エージェントのインストールパスで `start.sh` または `startd.sh` (デーモン実行)ファイルを実行します。

エージェントプロセスの確認

次のコマンドを通じて、動作中のWhaTap DMXエージェントプロセスとprxプロセスを確認できます。

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

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

モニタリングの確認

サーバーで正常にログが確認取れた後、コンソールに正しく登録されたことを確認するには、[WhaTapモニタリングサービスの初期画面](#)に移動します。確認するプロジェクトを選択し、画面の左側の**管理 > エージェントリスト**メニューを選択します。エージェントリストからデータベースサーバーの名前を確認できます。データベースサーバーの名前を別途設定する場合、[次の文書](#)を参照してください。

エージェントリスト											
Csv Download											
Agent Name, OID											
ステータス	名前 (初期)	ID	開始時間	活性 / 非活性	エージェントバージョン	Agent IP	DB IPアドレス	DB ポート	DB バージョン	CPU コア	
● 活性	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	🗑️ 🗑️
● 活性	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	🗑️ 🗑️
● 活性	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	🗑️ 🗑️
● 活性	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	🗑️ 🗑️

インストール問題の解決

データベースエージェントのインストール時に発生する問題を確認してください。

ファイアウォール設定の確認

ファイアウォールのブロックを解除してください。WhaTapサーバーのTCPアウトバウンドファイアウォールが設定されていると、モニタリング情報をサーバーに転送できません。

ファイアウォールを確認する方法

telnet コマンドを実行し、接続関連情報をが正常に表示します。(telnet[サーバーIP][サーバーポート])

Example

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

❗ WhaTapの収集サーバー情報は、**管理** > **エージェントインストール**メニューから確認できます。

制約事項

複数リージョンを1つのプロジェクトとして選択することはできません。

- [WhaTapモニタリングサービス](#)サイトでプロジェクトを作成する時にリージョンは重複選択できず、複数のリージョンを活用する場合は別途のプロジェクトを作成する必要があります。
- クラウド環境に複数のリージョン(Region)にサーバが存在する場合、ネットワークのlatencyなどパフォーマンス情報収集の制約を回避するために、リージョン(Region) 単位で収集サーバをインストールする必要があります。

- WhaTapは、顧客のニーズを満たすためにリージョン(Region)ごとに収集サーバーを構築します

エージェント名(ONAME) は重複使用できません

WhaTap DBXモニタリングは、エージェントを認識するために、エージェントのIPアドレスとポート(Port)情報を使用します。ユーザー環境でモニタリングするデータベースサーバーが同一IP、同一ポート(Port)を使用している場合、WhaTapサーバーのサーバーインスタンスを区分できません。

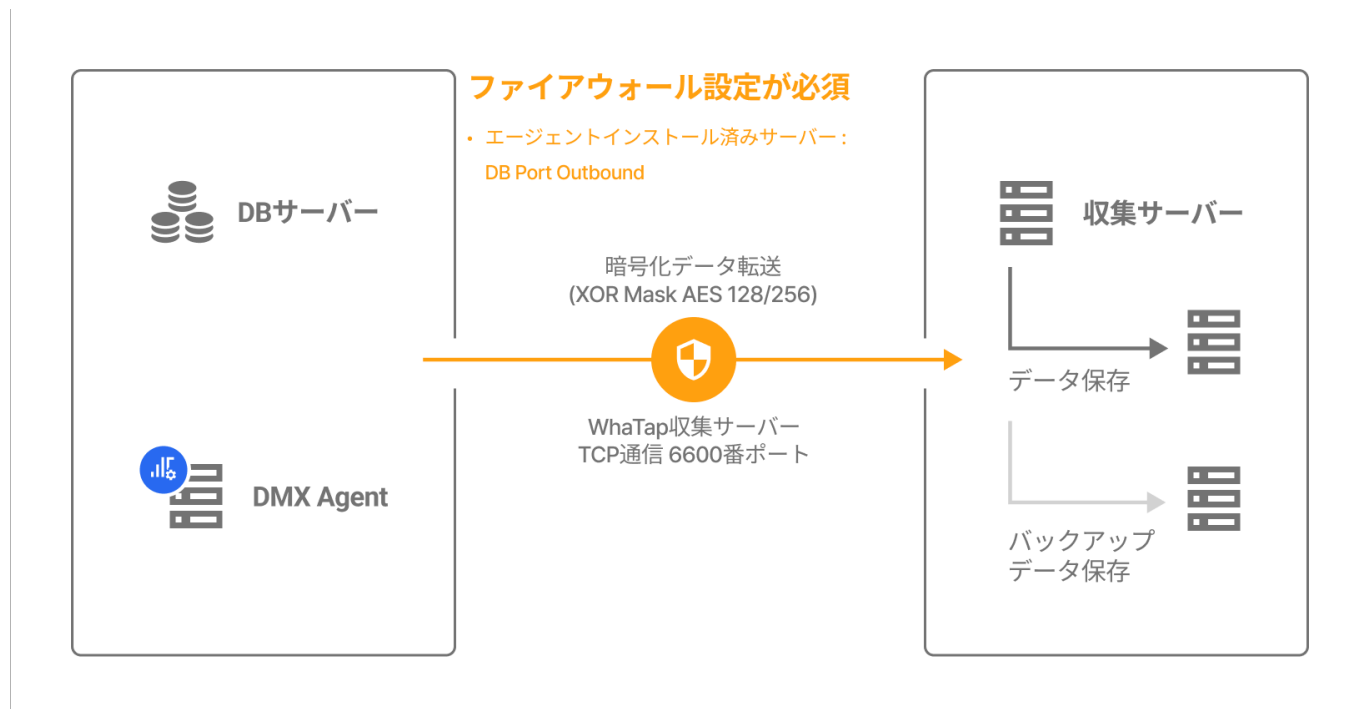
- サーバーインスタンスが自動で拡張される環境でなければ、[次の文書](#)を参照してデータベースを識別するための名称を直接設定すると解決できます。
- モニタリングするデータベースのInternal addressまたはネットワーク仮想化により重複したIPを使用している場合は、[次の文書](#)を参照して、別のデータベース名のパターンを使用してください。

設定

このページでは、WhaTapエージェントの構成について説明し、各種機能の設定方法について解説します。また、モニタリング対象となるDBサーバーのリソースを監視するためのオプション機能についても紹介します。

エージェント構成

以下は、WhaTapエージェントの構成図です。



• 収集サーバー

エージェントが収集したデータベースのパフォーマンスデータを収集および保存、統計情報を抽出し、これをユーザーに効率的な形式で提供します。収集サーバーはリージョン（Region）別に設定できます。リージョンごとに収集サーバーのアドレスが異なるように割り当てます。ユーザーが選択したリージョンによって収集サーバーのアドレスは異なる場合があります。プロジェクトを作成する際、リージョンも設定します。

• DMXエージェント

DMA（ダイレクトメモリアクセス）方式で性能データを収集し、収集サーバーに送信します。モニタリング対象の

データベースサーバーに直接インストールします。

- **ネットワーク**：WhaTapモニタリングエージェントは、収集したモニタリング情報をWhaTap収集サーバーに転送するために外部通信(TCP)6600番ポートを使用します。

prxプロセスのオプション

prxプロセスはdmxプロセスを監視しながら異常にダウンしているか、過剰なCPU、メモリを使用している場合はdmxプロセスを再起動させます。次のオプションを使用して、再起動基準を設定できます。

prx.conf

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

- **cpu_limit**

既定値 10 (単位%)

CPU使用率が指定した値以上であればdmxプロセスを再起動させます。

- **mem_limit**

既定値 500000 (500MB、単位KB)

メモリ使用量が指定した値以上であればdmxプロセスを再起動させます。

DMXエージェント設定

DMXエージェントに必要な設定は、[whatap.conf](#)ファイルに作成します。設定できるオプションについては、次を参照してください。

基本オプション

- **license** String

エージェントのインストール時にサーバーから付与されたプロジェクトアクセスキーを設定します。プロジェクトアクセスキーには、エージェントが属するプロジェクトとセキュリティ通信のためのパスワードキーが含まれています。

- **whatap.server.host** String

既定値 127.0.0.1,127.0.0.1

エージェントが収集したデータを送信するWhaTap収集サーバーのIPアドレスを設定します。収集サーバーの冗長化で2つ以上のIPを所有する場合は、コンマ(,)を区切り文字として使用します。設定したIPアドレスには、収集サーバーproxyデーモンがリスニング状態でサービスされる必要があります。

- **ipckey** String

`ipcs -m` コマンドを使用して取得できます。モニタリングするOracleのKey値(16進数)です。

① `ipcs` コマンドは、プロセス間の通信(IPC)に関する情報を提供します。 `-m` オプションは、共有メモリスourcesを照会します。

- **rac_name** String RAC

Oracleの**Real Application Clusters(RAC)** クラスターグループに束ねられているインスタンスのグループ名を設定するオプションです。このオプションを使用して、複雑なRACクラスターグループ名を簡潔で識別しやすい名前に設定できます。このオプションによって設定されたRACクラスターグループ名は、WhaTapモニタリングサービス画面で確認できます。

例えば、RACクラスターグループ名が'`my_long_rac_cluster_group_name`'のように長い場合、ユーザーは次のように簡潔な名前で設定できます。

`whatap.conf`

```
rac_name=my_rac_cluster_group
```

WhaTapモニタリングサービスで'**my_rac_cluster_group**'と表示され、ユーザーはそのRACクラスターグループをすばやく識別してモニタリングできます。

データベースデータ収集オプション

- **bind_elapse** Int

既定値 3

SQLが設定した時間(秒)より長く実行すると、バインド値を収集するように設定します。設定値が 0 であれば、オプションは機能しません。

`bind_interval` を設定しなくてもSQLの実行時間が設定値より長い場合は、バインド値が収集されます。

- **bind_interval** Int

既定値 0

バインドキャプチャ間隔を秒単位で設定します。既定値(0)を設定するとバインドキャプチャしません。

`v$sql_bind_capture` で設定した間隔単位で収集します。

データが多いかもしれないので、最後に収集した時間以降でありながら `bind_max_count` 分だけ収集します。

- **bind_max_count** Int

既定値 1000

バインドキャプチャ値を収集するときに、いくつまで収集するかを設定します。最後の収集時間以降のバインドキャプチャ値のみを収集できます。

- **codefrom / codeto** String

SQL textで韓国語の文字化けの問題がある場合、文字エンコーディングを変えるオプションです。指定する値は、`iconv -l`のリストから選択できます。

例) `codefrom=EUC-KR, codeto=UTF-8`

- **conn_fail_count** Int

既定値 15

15回連続connection errorが発生した時に通知を送信します。希望する回数を入力して、通知の発生基準を調整できます。

- **event** String

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

RAC環境の時は、gc cr block receive time、gc current block receive timeの2つの指標を基にさらに収集します。

イベント関連の指標を収集するように設定します。 v\$system_event 、 v\$event_name などの EVENT カラムを参照して収集したい指標名を作成してください。 event+= などの方法で作成すると、基本指標とともに追加で収集できます。 空白なくコンマ(,)を区切り文字として利用して複数入力できます。

イベント1つあたり、次の3つの値を収集します。

```
- time waited : 当該イベントで待ち受けた総時間(単位: 1/100秒)。例) db file sequential read time waited

- timeouts : タイムアウトで終わった待機回数。例) db file sequential read total timeouts

- total waits : 当該イベントが発生した総回数。例) db file sequential read total waits
```

- **interval** Int

既定値 3

データ収集サイクルを秒単位で設定します。

① Oracle Proは、メモリアクセスを通じてアクティブセッション及びSQL情報を1/20秒単位で収集します。このように収集されたデータに基づいてstatを生成するため、データの損失が少ないです。 interval オプションによって現在のDB状態を送信する際、stat、event、SQL stat、xview、lock、PQ情報はメモリから直接読み込み、迅速に処理されます。一方、bind情報、DB parameter、plan情報、テーブル、テーブルスペース、Wait classなどはクエリーを通じて収集します。

- **multi_db_stat** Boolean

既定値 true

Multi DBを使用する場合、データベースごとに指標を収集するか、全体インスタンス単位で合算(sum)して収集するかを設定します。

- **ora_hidden_param** Boolean

既定値 false

ヒドゥンパラメータ値を収集するかを設定します。 ヒドゥンパラメータを収集するには、オプション値を true に

設定してください。

- **plan_change** Int

既定値 1

プランの変更履歴を収集するかを設定します。

- 1 : sysを含まない
- 2 : sysを含む

- **plan_elapse** Int

既定値 3

実行時間が5秒以上かかるプランを収集します。

- **segments** Boolean

既定値 true

セグメントサイズの情報を収集します。統計/レポート>テーブルスペースのサイズメニューで該当データベースまたはテーブルスペースを選択すると、サイズ情報を確認できます。

- **segments_exclude_sys** Int

既定値 1

sysに該当するアカウントを除いて、セグメントのサイズ情報を収集します。

- **segments_hour** Int

既定値 5

セグメントサイズの情報を収集する時刻を設定します。既定値は 5 であり、午前5時に収集を始めます。

- **segments_max_count** Int

既定値 50

テーブルスペースごとにサイズの大きいセグメント50個だけを収集して保存します。

- **segments_min_size** Int

既定値 10

セグメントサイズの情報を収集する最小単位の既定値は10MBです。設定した値より小さいセグメントは収集しません。

- **segments_sys** String

既定値:

Details



<summary>計18個のアカウントを表示</summary>

```
`APPQOSSYS`, `AUDSYS`, `CTXSYS`, `DBSFUSER`, `DBSNMP`, `DVSYS`,
`GSMADMIN_INTERNAL`, `LBACSYS`, `MDSYS`, `OJVM SYS`, `OLAPSYS`,
`ORDDATA`, `ORDSYS`, `OUTLN`, `SYS`, `SYSTEM`, `WMSYS`, `XDB`
```

sysに該当するアカウントの一覧です。上記のアカウントに追加する場合は、`segments_sys+='xxx'` で追加できます。

- **stat** String

既定値:

Details



<summary>計20個の指標を表示</summary>

```
`CPU used by this session`, `data blocks consistent reads - undo records applied`, `db block changes`,
`db time`, `enqueue waits`, `execute count`, `logons current`, `opened cursors current`,
`parse count (hard)`, `parse count (total)`, `physical reads`, `physical writes`, `redo entries`,
`rollback changes - undo records applied`, `session logical reads`, `sorts (disk)`, `sorts (memory)`,
`transaction tables consistent reads - undo records applied`, `user commits`, `user rollbacks`
```

統計関連の指標を収集するように設定します。`v$sysstat`、`v$statname` などの `NAME` カラムを参照して収集したい指標名を作成してください。`stat+=` などの方法で作成すると、基本指標とともに追加で収集できます。空白なくコンマ(,)を区切り文字として利用して複数入力できます。

- **tablespace** Int

既定値 1

テーブルスペースの free space を照会する際に使用するビューを選択します。環境によって `dba_free_space` の照会で `query timeout` が発生する場合は、別のビューに変更できます。

- 1 : dba_free_space を照会
- 2 : dba_lmt_free_space を照会
- 3 : usage_metrics を照会

対応バージョン: dmX 1.0.1x 以上

• tablespace_interval Int

既定値 60

テーブルスペース情報の収集周期を分単位で設定します。最小値は 5 であり、5 より小さい値を設定すると 5 分に自動補正され、補正したことを知らせるログが記録されます。60 より大きい値も設定できます。

対応バージョン: dmX 1.0.8c 以上

• tablespace_send_opt Int

既定値 0

テーブルスペース情報の送信方式を設定します。テーブルスペースが多い環境で同一データが重複送信されてエラーが発生する問題を改善できます。

- 0 : 分割送信
- 1 : 一括送信

対応バージョン: dmX 1.0.8c 以上

• topevent Int

既定値 50

イベント関連の主要指標の変化を高い数値順に設定された値だけ収集します。stat_top、event_top カテゴリのデータを使用します。

① このオプションは、値を設定しないと topstat の設定値に従って動作します。

• topstat Int

既定値 50

統計関連の主要指標の変化を高い数値順に設定された値だけ収集します。stat_top、event_top カテゴリのデータを使用します。

- **trace_tkprof_path** String

SQL trace 実行時の tkprof 実行ファイルのパスを指定します。

対応バージョン: dmX 1.0.4y 以上

- **waitclass_opt** Int

既定値 1

Wait Class 指標の収集方式を設定します。

- 1: Memory ベースの収集。他の指標と同じ周期で収集します。
- 1 以外の値: 既存の方式(クエリベース、1 分周期固定)を維持します。

対応バージョン: dmX 1.0.8c 以上

- **xos** Boolean

既定値 true

データベースサーバーのリソースを追加でモニタリングするためのXOS エージェント([xos.conf](#))の活性化オプションです。

- **xview_min** Int

既定値 100 (1秒)

単位はセンチ秒(centisecond, 1/100秒)です。このオプションは、設定した値よりも長い経過時間(elapsed time)を持つジョブのみを処理します。

- -1: このオプションを非活性化します。
- 0: すべての作業を収集します。

ユーザーメトリック収集 (SQLMON)

SQLMON は、ユーザー定義の SQL 結果を定期的に収集し、メトリクスとして保存します。数値を返し、負荷が低いクエリに適しています。

複数行の結果は最大 100 件まで保存し、結果値が int または float の場合は数値として認識します。SELECT 以外に INSERT、UPDATE、DELETE 文も使用できます。

INSERT/UPDATE/DELETE 対応: dmX 1.0.8s 以上・複数行(最大 100 件)保存: dmX 1.0.5g 以上

- **sqlmon** Int

既定値: 1

- 1: dbx と sqlmon を同時に実行
- 2: sqlmon のみ実行

対応バージョン: 2.3.5 以上

設定

1. dbx エージェント配下に `sqlmon` ディレクトリを作成し、収集したいクエリを `<filename>.sql` ファイルとして作成します。

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

2. 収集するクエリを `<filename>.sql` ファイルとして保存します。

`.sql` ファイルの1行目に実行オプションを、2行目以降に SQL 文を記述します。

オプション形式

```
周期(分);動作時間;タグ名
SQL文
```

項目	説明	例
周期	実行周期（分単位）、最小値 1	1 → 1分ごとに実行
動作時間	HHMM-HHMM 形式、カンマ（,）で複数の時間帯を指定可能。0 の場合は常時実行	0900-1100 → 09:00～11:00のみ実行
タグ名	メトリクスに保存するタグ名。省略時はファイル名がタグ名として使用されます	whatap

例：1分周期、09:00～11:00動作、タグ名 whatap

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

例：1分周期、常時動作、タグ名はファイル名を使用

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

- ❗ 動作時間は日付をまたいで設定できます。（例：2300-0100 → 23:00～翌日01:00）
- タグ名を省略する場合、動作時間の後にセミコロン（;）を付けしないでください。
- SQL文の末尾にセミコロン（;）を付けしないでください。

3. `whatap.conf` で SQLMON を有効化します。
4. エージェントを再起動します。
5.  サイトマップ > メトリクス照会 メニューで、`sqlmon` カテゴリを確認します。

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

AES256暗号化の適用

WhaTap DMXエージェントは、収集されたデータを暗号化してサーバーに送信します。データの重要度や設定によって変更できます。基本的にXOR演算とAESアルゴリズムによる暗号化を使用し、平文を128ビット単位に分割して暗号化、復号を行います。ユーザー設定により、256ビットまで拡張できます。

WhaTap DMXエージェントをインストールした経路で `whatap.conf` ファイルに次のようにオプションを追加します。

`whatap.conf`

```
1 cypher_level=256
```

設定を追加してから、エージェントを再起動してください。

エージェント名の設定

自動設定する

エージェント設定([whatap.conf](#))で名前を設定しないと、エージェント名は**DMX-{hostname}**と同じように、ホスト名が自動的に組み合わされて指定されます。

手動設定する

エージェント名を直接設定したい場合は、[whatap.conf](#)ファイルで `whatap.name` または `object_name` オプションに目的の名前を入力します。ユーザーが入力した名前がエージェント名として設定されます。

whatap.conf

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

XOSエージェント設定

`xos.conf`ファイルに次のオプションを設定すると、データベースサーバーのリソースを追加でモニタリングできます。このオプションを利用するには、`whatap.conf`ファイルで `xos` オプションを `true` に設定してください。

- `interconnect_nic` String RAC

設定されたオプション値に対応するネットワークカード情報を収集します。このオプションを設定しないと、interconnect関連の指標を収集しません。

`xos.conf`

```
# example
interconnect_nic=enp0s3
```

`netstat -i` コマンドで確認されたネットワークインターフェースのうちReal Application Cluster(RAC)のinterconnectに使用されているネットワークインターフェースを入力します。設定された値は、そのネットワークカードの性能と活動をモニタリングするために使用されます。

- `cpu_limit` Int

既定値 0

limitの設定値より大きい値を持つプロセス情報を収集します。1/100%単位で1.23%の場合、123 に設定します。

- `mem_limit` Int

既定値 10240

limitの設定値より大きい値を持つプロセス情報を収集します。単位は「KB」で既定値である10240は、10MBになります。(1024 KB = 1 MB)

- `disk_usage_interval` Second

既定値 60

インスタンス別のディスク使用率と使用可能サイズ、全体サイズ情報を表示します。値が 0 の場合、ディスク使用率は収集しません。

- `process` String

プロセスの実行状態をモニタリングできます。オプション値でプロセス名を入力します。コンマ (,) を区切り文字として複数設定することができます。

xos.conf

```
process={process_name}
```

例えば、topとlock.shプロセスをモニタリング対象に設定するには、次のようにオプションを設定してください。

xos.conf

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

- ① process オプションを使用して、データベースの実行中のプロセスが終了される場合にアラート通知を送信できます。詳細については、[次の文書](#)を参照してください。

- **iplist_interval** Second

データベースサーバーのIP一覧を収集できます。秒単位(最低3秒)で時間間隔(interval)を設定すると収集できます。

- **file** String

ログメニュー(ログTail、ログトレンド、ログ検索、ログ設定)を使用するには、file オプションにモニタリングするログファイル名のフルパスを入力します。

xos.conf

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

- ✔ モニタリングするログファイルを追加するには file 、 file1 、 file2 といった方法でオプションを追加します。

xos.conf

```
file={log_file_path1}
file1={log_file_path2}
file2={log_file_path3}
file3={log_file_path4}
```


ユーザーメトリクス収集 (SHMON)

SHMONはユーザー定義のシェルスクリプトの実行結果を定期的に収集し、メトリクスとして保存します。

- **shmon** Int

既定値: 1

対応バージョン: dbx 2.55.01以降 & xos 1.2.0f以降

設定

1. エージェントのインストールパスに `shmon` ディレクトリを作成します。
2. 収集するスクリプトを `.sh` ファイルとして保存します。
3. `xos.conf` でSHMONを有効にします。
4. エージェントを再起動します。
5. メトリクスビューで `shmon` カテゴリを確認します。

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

管理

エージェント削除

エージェントがインストールされたサーバーから削除する

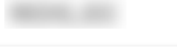
サーバー上で起動されるJavaプロセス中 `whatap.agent.dmx` 文字列を含むプロセスを終了します。エージェントをインストールした経路で`stop.sh`を実行します。その後、エージェントファイルおよびエージェントに登録したUser権限を削除処理してください。

サービス画面から削除する

WhaTapモニタリングサービスからエージェントを削除するには、**エージェントリスト**メニューに移動します。エージェントを削除する前に、エージェントをインストールしたサーバーでエージェントプロセスを終了する必要があります。

1. プロジェクトリストからエージェントを削除するプロジェクトを選択します。
2. 左側のメニューから**管理** > **エージェントリスト**を選択します。
3. エージェントリストで削除するエージェントで右側の  ボタンをクリックします。

- ⓘ • 非活性化状態のエージェントのみ削除できます。

ステータス	名前 (初期)	ID
 活性	 	
 非活性	 	
 活性	 	

- エージェントのバージョンによっては、削除機能をサポートしていない場合があります。最新バージョンのエージェントに更新します。

ダッシュボード

WhaTapエージェントのインストールが完了しましたら、[WhaTapモニタリングサービス](#)にログインしてください。初期画面はプロジェクトリストです。プロジェクトリストで作成したプロジェクトを確認できます。プロジェクトリストで作成したプロジェクトを選択すると、[インスタンスリスト](#)が表示されます。

WhaTapデータベースモニタリングサービスでダッシュボードは、プロジェクトの全体のステータスを一目で確認できる機能です。モニタリングされているリソース全体の使用量を確認でき、リアルタイムダッシュボードでモニタリング対象のリソースをフィルタリングできます。

- 主なパフォーマンス指標とステータスを表示するように構成されています。パフォーマンス指標は5秒間隔で更新されます。
- 1つのインスタンスで複数のDBをフィルタリングできます。
- 収集したデータに基づいて、特定の1日あたりの全体的なトレンドを分析できる機能を提供します。特定の時間に進行中だったセッションを特定し、低速なQueryを分析することができます。

インスタンスリスト

エージェントをインストールしたデータベースサーバーの一覧を確認できます。主なパフォーマンス指標と状態(Status)を表示できるように構成されています。パフォーマンス指標は5秒間隔で更新され、一覧のカラムは設定を使用して目的のカラムに配置できます。詳細については、[次の文書](#)を参照してください。

インスタンスモニタリング

- データベースサーバーの主要メトリクスとアクティブセッションに関する詳細情報をリアルタイムでモニタリングし、問題になる可能性のあるアラートを確認できます。関心のある重要メトリクスを選択して、全体的なシステム状況をモニタリングできます。
- リアルタイムモニタリングは、5秒間隔でパフォーマンス指標をモニタリングします。実行中の**アクティブセッション**、**ロックツリー**、**プロセス情報**を確認できます。

詳細については、[次の文書](#)を参照してください。

マルチインスタンス

プロジェクトに登録されたすべてのインスタンスを5秒間隔でリアルタイムでモニタリングします。特定のインスタンスにフィルタリングしてモニタリングすることもできます。詳細については、[次の文書](#)を参照してください。

RACモニタリング

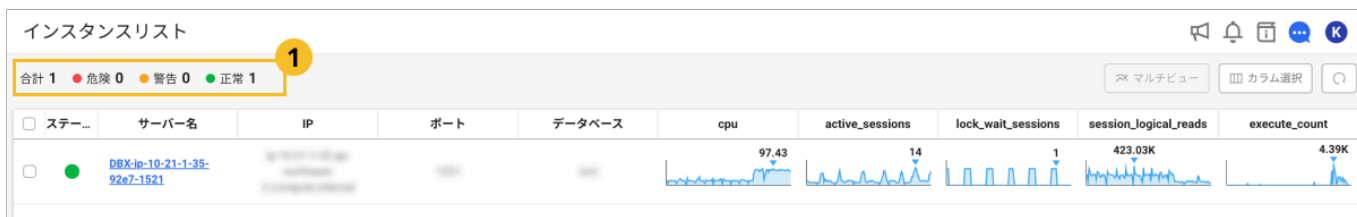
RACモニタリングは、Oracleデータベースでクラスターからなるデータベースを一つの単位にまとめてモニタリングできます。詳細については、[次の文書](#)を参考にしてください。

インスタンスリスト

ホーム画面 > プロジェクト選択 > ダッシュボード > インスタンスリスト

エージェントをインストールしたデータベースサーバーの一覧を表示し、各インスタンスの主要なパフォーマンス指標と状態が把握できます。パフォーマンス指標は、5秒間隔で更新されるし、カラム選択ボタンで表示する指標を選択できます。

基本画面ガイド



- ① 領域は、プロジェクトに設定されているインスタンスの状態を表示します。合計にはテーブルに表示中のインスタンスの数、危険、警告、正常にはステータスカラムの各ステータスの数が表示されます。
 - 正常: 定常状態のエージェントの数です。
 - 警告: アラート通知 > 通知設定メニューで設定したイベントのしきい値が警告レベルを超えたエージェントの数です。
 - 危険: アラート通知 > 通知設定メニューで設定したイベントのしきい値が危険レベルを超えたエージェントの数です。

① アラート通知メニューの詳細については、[次の文書](#)を参照してください。

- ステータスカラムで正常ではない状態のアイコンをクリックするとイベント画面が表示されます。そのインスタンスで発生したイベントの履歴を確認できます。
- 各インスタンスを選択すると、インスタンスモニタリングメニューが表示されます。選択したインスタンスの詳細をリアルタイムでモニタリングできます。

① インスタンスモニタリングメニューの詳細については、[次の文書](#)を参考にしてください。

- 主要なパフォーマンス指標をグラフチャート形式で提供します。最も高い数値をチャートに表示します。チャートにカーソルを合わせると、その場所の数値が表示されます。パフォーマンス指標の詳細については、[次の文書](#)を参照してください。
- 画面右上の ボタンをクリックすると、リアルタイムの通知が確認できます。通知設定の詳細については、[次の文書](#)を参照してください。

① 画面上部にある固定領域の詳細については、[次の文書](#)を参照してください。

インスタンス性能指標の比較

2つ以上のインスタンスを選択して、性能指標を比較できます。性能指標を比較するインスタンスを一覧から選択すると、画面右上のマルチビューボタンが活性化されます。マルチビューボタンを選択すると、マルチインスタンスメニューに移動します。主要な性能指標のデータを比較したグラフチャートとアクティブセッション、ロックツリーに関する詳細情報を確認できます。

① マルチインスタンスメニューの詳細については、[次の文書](#)を参考にしてください。

カラム表示または非表示

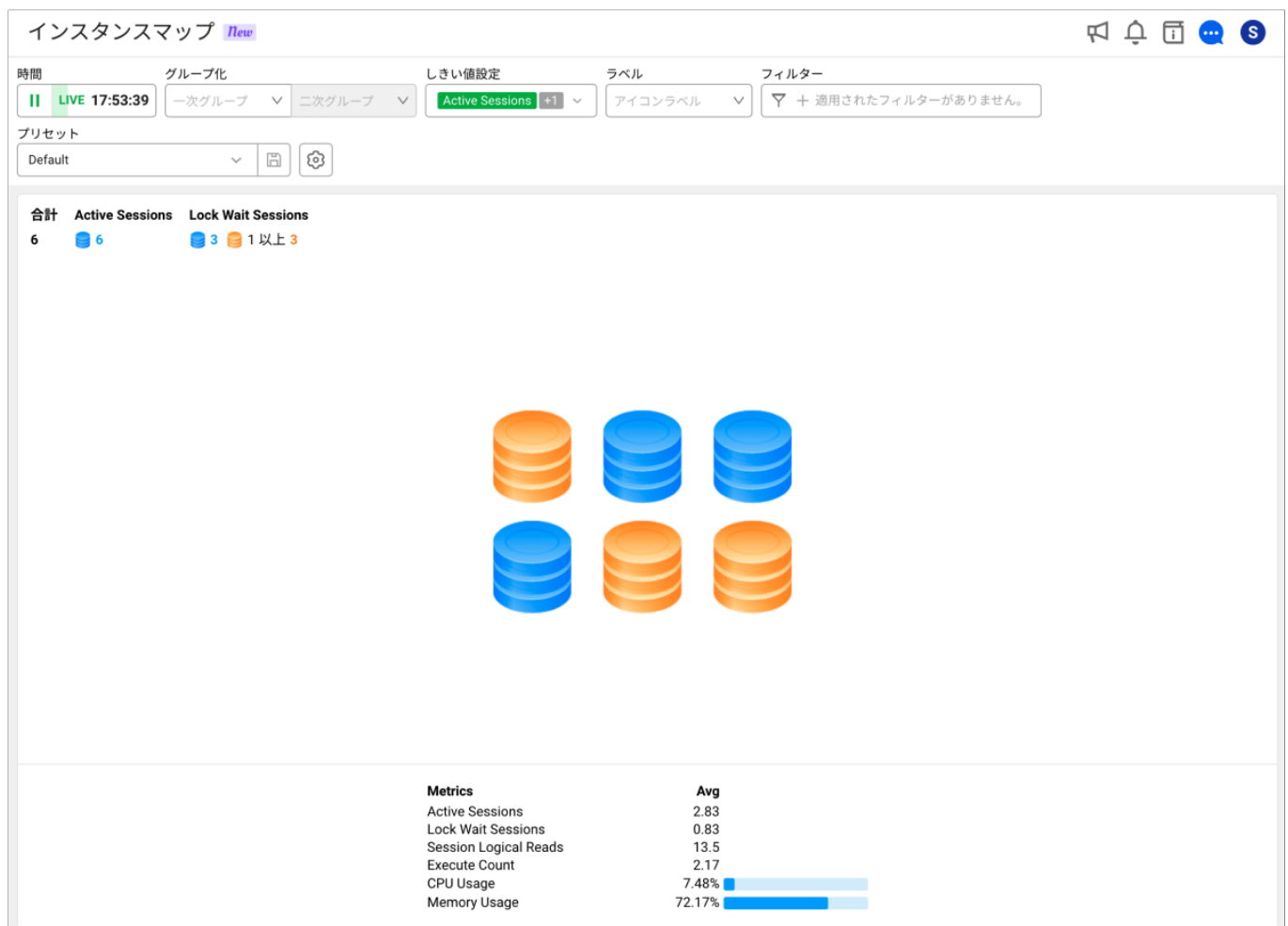
インスタンスリストのパフォーマンス指標カラムを表示または非表示できます。画面右上の**カラム選択**ボタンをクリックします。目的の項目を選択、または解除します。カラムを表示または非表示できます。

インスタンスマップ

ホーム画面>プロジェクト選択>**ダッシュボード**>インスタスマップ

インスタンスマップは、データベースサーバーをモニタリングしているインスタンスの状態を視覚化し、難しく感じられることの多いデータベースモニタリングを簡単に行えるよう支援する機能です。

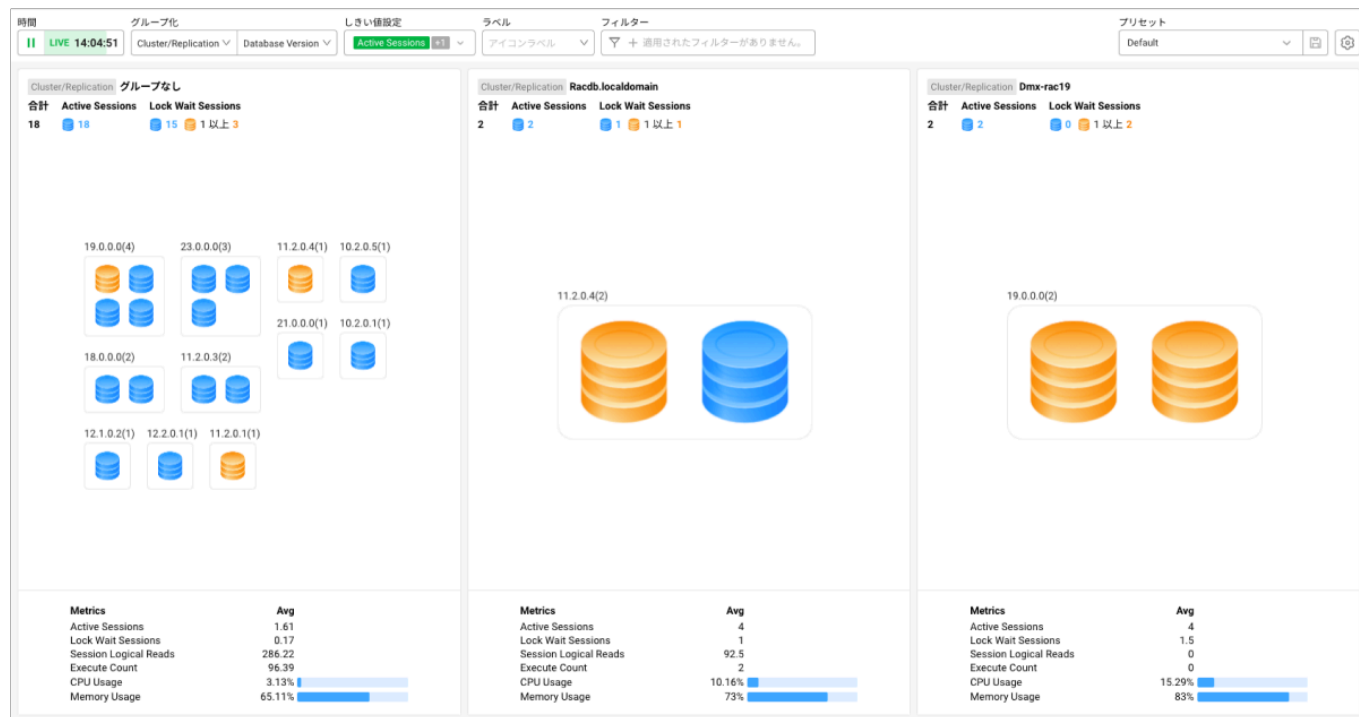
データベースサーバーの主要指標にしきい値（スレッシュホールド）を設定することで、データベースの状態を一目で把握でき、実行中のアクティブセッションもリアルタイムで確認できます。また、インスタンスモニタリング画面やマルチインスタンスモニタリング画面へ遷移して、障害が発生したデータベースの分析を迅速に行えるよう連携されています。



❗ CPU使用率、メモリ使用量など、データベースサーバーのプロセス使用状況はXOSエージェントの実行によりデータ収集が可能です。

基本画面

インスタンスマップは主に **ユーザー設定エリア** とインスタンスの **マップチャートエリア** で構成されています。



ユーザー設定

インスタンスマップ画面上部のユーザー設定エリアでは、グループ化、しきい値設定、ラベル、フィルター、プリセット、⚙️ 追加設定が可能です。

グループ化

インスタンスをさまざまな基準でグループ化したり、ユーザーのニーズに応じてカスタム基準でまとめてモニタリングできます。グループ化機能は最大2階層までサポートされています。2階層目のグループをクリックすると、そのグル

ープに属するインスタンスのデータをまとめて確認できる詳細画面が表示されます。

しきい値の設定

データベースモニタリングの主要指標に対してしきい値を設定することで、インスタンスの状態を色で表現することができます。複数の指標を設定でき、各指標ごとに警告（Warning）および重大（Critical）の範囲を指定できます。正常（Normal）区間は青色で表示します。設定された範囲に応じて、該当状態の色が適用されます。

1. 画面上のしきい値設定 オプションをクリックしてください。
2. 任意の指標を選択し、Warning と Critical の範囲を設定してください。
3. メトリクスを追加する場合は、下段の + 指標追加 をクリックしてください。
4. 各指標のしきい値設定が完了したら、保存 ボタンをクリックしてください。

インスタンス状態の表示優先順位

状態表示の優先順位はCritical区間が一番高いです。複数の指標のうち、いずれか1つでも Critical の範囲に該当する場合、そのインスタンスは赤色で表示されます。以下の例では、2つのインスタンスがともに Critical として表示されます。

ラベル

インスタンスアイコンの内部にはテキストラベルが表示されます。ラベルの種類は、データベース情報および主要指標で構成されています。

フィルター

フィルターを指定することで、特定のインスタンスに集中してモニタリングを行うことができます。

- **フィルターキー:** データベースおよびインスタンス情報がフィルターキーとして提供されます。
 - **収集カラム:** インスタンス名、クラスタ/レプリケーション情報、エージェントバージョン、CPUコア数、データベースの種類、インストール位置、バージョン、IP、Port
- **条件:** フィルターの条件は、 $\geq$ 、 $\leq$ 、 $>$ 、 $<$ などの数値型条件と一致、不一致、包含、不包含などの文字列条件で設定できます。

プリセット

ユーザーは、設定したオプションをプリセットに保存し、必要な時読み込みできます。プリセットはプロジェクト単位で保存されます。これにより、プロジェクトメンバーが同じ設定でインスタンスをモニタリングすることができます。デフォルトプリセット(Default) が提供されており、デフォルトプリセットは削除できません。

✔ 基本プリセット

インスタンスマップのデフォルトプリセットでは、`active_sessions` と `lock_wait_sessions` のしきい値が以下のように設定されています。

- `active_sessions`
 - **Warning:** 値が 20 以上
 - **Critical:** 値が 30 以上
- `lock_wait_sessions`
 - **Warning:** 値が 1 以上
 - **Critical:** 値が 3 以上

追加設定

インスタンスマップ右上の ⚙️ アイコンをクリックして、追加設定が可能です。

- **第1グループなしを表示:** 第1グループを設定した際、該当グループが存在しない'グループなし'を表示するかどうかを設定します。
- **グループ下部メトリクスエリアの表示:** 下部のメトリクス情報の要約を表示するかどうかを設定します

インスタンスマップチャート

インスタンスマップチャートは、3つのエリアで構成されています。

上部エリア

インスタンスの全体数、またはグループ化されている場合はグループごとの総数と、選択されたメトリクスの状態別の件数を表示します。

インスタンスの可視化エリア

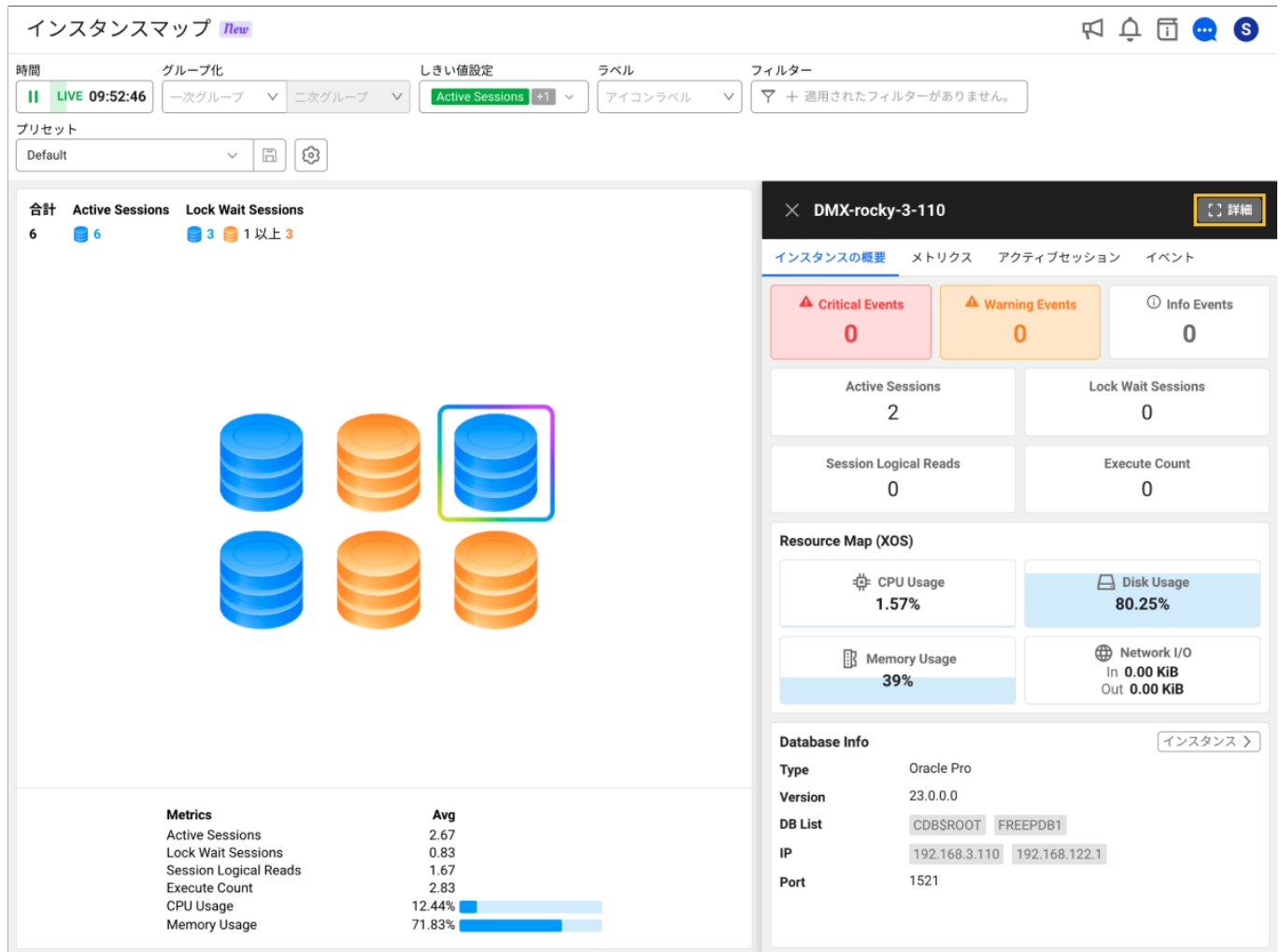
すべてのインスタンスの状態と件数を直感的に把握できるビジュアルダッシュボードを提供します。

下部エリア

主要なメトリクスのインスタンス平均値を表示します。⚙ の **グループ下部メトリクスエリアの表示** オプションを使ってオン/オフを切り替えることができます。

要約の表示と詳細分析

インスタンスマップで、確認したいインスタンスを選択してください。右側に表示されるサマリービュー画面では、該当インスタンスの概要情報を **インスタンス概要**、**メトリクス**、**アクティブセッション**、**イベント**、**ログタブ** で確認できます。サマリービュー画面で **詳細分析** ボタンをクリックすると、詳細分析画面を確認できます。



インスタンス概要

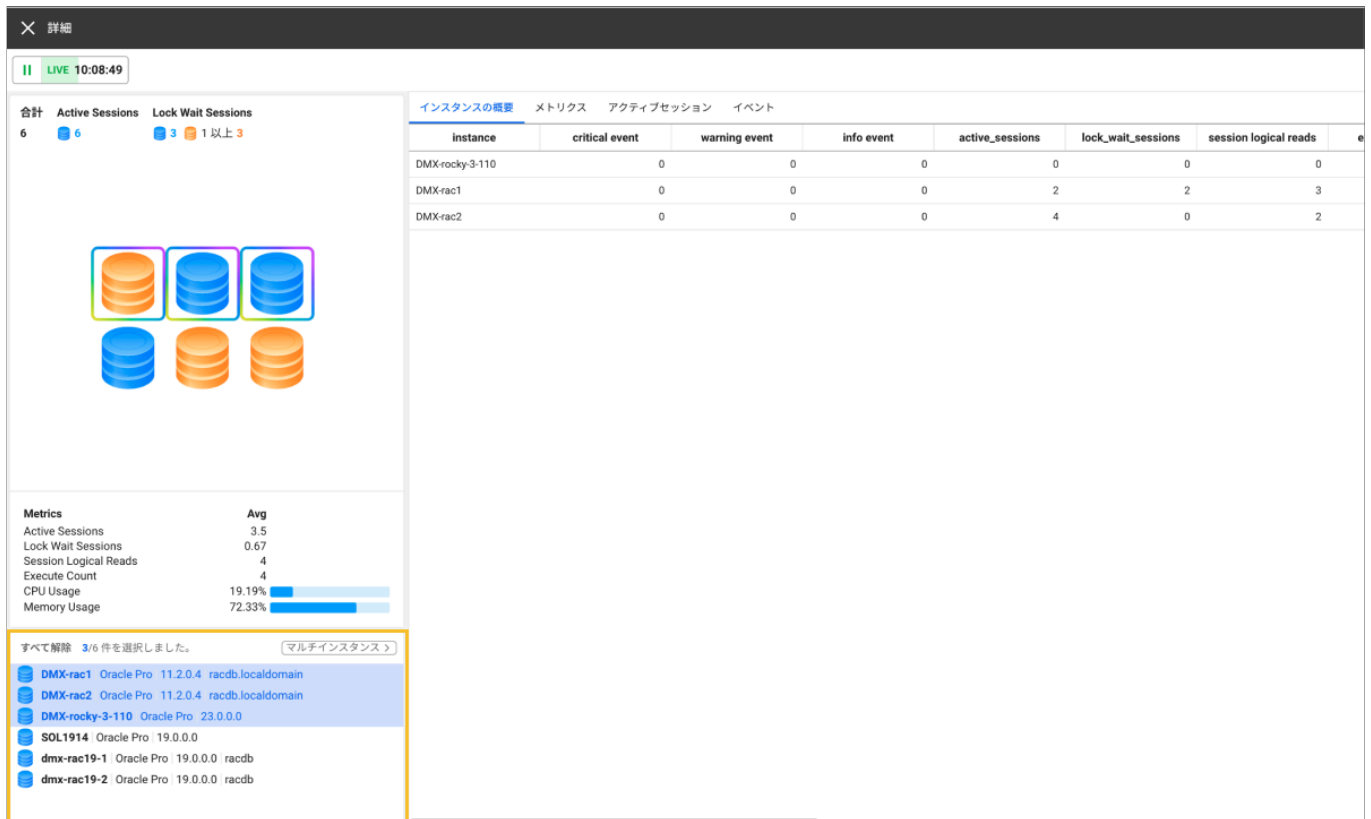
詳細分析ページの **インスタンス概要** タブでは、選択したインスタンスがモニタリングしているデータベースの情報および指標情報、発生したイベントを確認できます。また、インスタンスモニタリングメニューへ移動することもでき、非アクティブなインスタンスの場合は、非アクティブ化された時点のインスタンスモニタリング画面へ移動します。

- **イベントエリア:** ユーザーが設定したアラート基準に基づくイベント発生情報を提供します。各イベントカードをクリックすると、どのイベントが発生したかを確認できます。
- **指標エリア:** 主要なデータベースメトリクスおよびリソース使用状況をリアルタイムで確認できます。リソース使用状況は XOS エージェントを設定することで確認可能です。
 - データベース指標

- Active Sessions: 実行中のクエリセッション数
- Lock Wait Sessions: ロック待機中のセッション数
- Session Logical Reads: メモリから読み取ったデータブロック数 (db block gets + consistent gets)
- Execute Count: 実行された SQL ステートメントの呼び出し回数
- リソース使用状況 (Resource Map (XOS))
 - CPU Usage: CPU 使用率
 - Disk Usage: ディスク使用量
 - Memory Usage: メモリ使用量および使用率
 - Network I/O: ネットワークの送受信量
- データベース情報エリア (Database Info) : データベースの詳細情報を確認できます。 インスタンスモニタリングをクリックすると、選択したインスタンスのモニタリング画面へ移動できます。

インスタンス概要詳細ビュー

詳細ビュー画面の **インスタンス概要** タブでは、複数のインスタンスを選択して、主要メトリクスおよびデータベース情報をテーブル形式でモニタリングできます。 **インスタンス概要** タブの右上にある **詳細ビュー** ボタンをクリックするか、2次グループを設定後、インスタンス視覚化エリアで2次グループ名をクリックすると、詳細ビュー画面に移動します。



- ・詳細ビュー画面では、視覚化エリア内のインスタンスアイコンをクリックするか、画面左下のインスタンスリストをクリックして、インスタンスを選択または選択解除できます。すべて選択 または すべて解除 をクリックすることで、全インスタンスの一括選択または解除が可能です。
- ・複数のインスタンスを選択した場合、マルチインスタンスモニタリング> ボタンをクリックして、選択されたインスタンスを マルチインスタンスモニタリング 画面でモニタリングできます。

メトリクス

メトリクス タブでは、選択されたインスタンスのメトリクストレンドを確認できます。時間セレクターを操作することで、リアルタイムまたは過去のメトリクスの変化を確認できます。

アクティブセッション

アクティブセッション タブでは、リアルタイムで実行中のセッションおよびクエリの内容を確認できます。

❗ テーブルリスト内の文字色は、黒 → オレンジ → 赤 の順にセッションの実行速度が遅いことを示します。

イベント

イベント タブでは、直近1時間に発生したイベントを確認できます。発生したイベント項目をクリックすると、イベントが発生した時点のインスタンスモニタリング画面に移動します。

ログ

ログ タブでは、発生したログの確認およびリアルタイムログの確認が可能です。右側の **詳細ビュー** ボタンをクリックすると、より詳細なログを確認できます。

インスタンスモニタリング

ホーム画面 > プロジェクト選択 > **ダッシュボード** > インスタンスモニタリング

インスタンスモニタリングメニューでは、データベースサーバーのコア指標とアクティブセッションに関する詳細情報をリアルタイムでモニタリングし、問題のあるアラートを特定できます。デフォルトでは、過去10分間の収集データをリアルタイムで表示します。収集データは5秒間隔で更新します。

基本画面ガイド



1 検索時間と検索先を選択します。

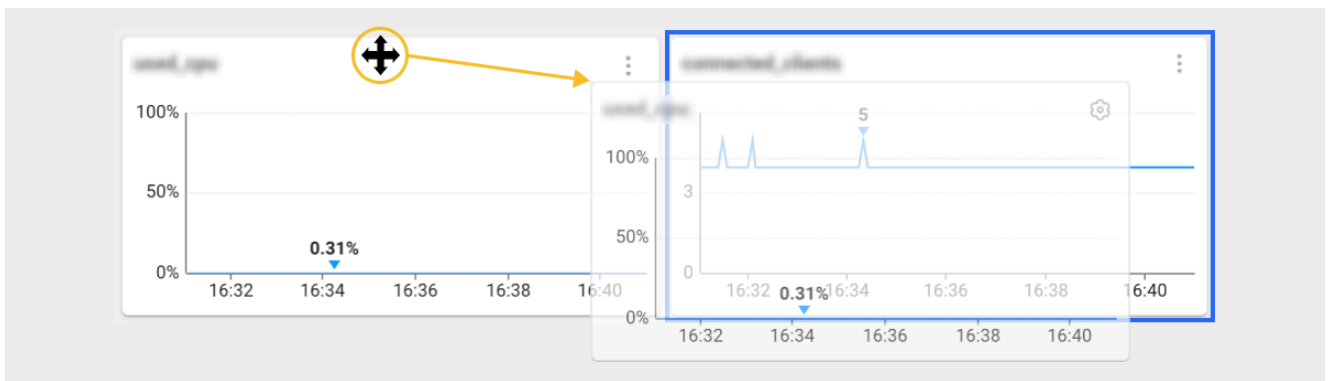
基本的に、直近10分間の収集されたデータをリアルタイムで表示します。時間とインスタンス項目から時間と対象を選択すると、収集データが2 グラフチャートと4 のアクティブセッションテーブルに自動的に反映されます。

- 過去のデータを照会するには、**時間**で緑のボタンをクリックし、希望の照会時間を選択します。過去3週間までのデータを表示できます。
- 任意の日時を照会するには**時間**で **II** ボタンを選択します。日付と時間テキスト領域をクリックして、日付と時間を選択するオプションを表示します。再度、リアルタイムでデータを検索します **▶** ボタンを選択します。
- 他のエージェントのデータを照会するには、**インスタンス**をクリックし、目的のエージェント名を選択します。

2 グラフチャートセクション

3 Default プリセットで保存された16個の指標をグラフチャート形式のウィジェットで表示します。横軸は時間、縦軸は各指標の数値です。各指標の意味を知りたい場合は、名前の横の ⓘ ボタンを選択するか、[次の文書](#)を参照してください。

- 他のウィジェットの位置を変更する場合は、ウィジェットの上部を選択した状態でドラッグします。ただし、ウィジェットのサイズは変更できません。



- 時間の**リアルタイム照会**モードを停止し、ウィジェットでチャートの特定時間をクリックすると、選択した時間に行われたセッション情報を **4 アクティブセッション**テーブルで確認できます。その後、**アクティブセッション**テーブルの上に、5秒単位でデータを照会できるタイムバーが表示されます。

- ⓘ ◦ 選択した時間は、**アクティブセッション**テーブルの右上で確認できます。
- **アクティブセッション**テーブル一覧の文字の色は黒 → オレンジ → 赤の順にセッションの実行速度が遅いことを意味します。

- リアルタイム照会モードを停止した後、ウィジェットの右上の ⓘ ボタンをクリックすると、チャートデータをCSVファイルにダウンロードできます。
- **Top Stat:** ウィジェットに表示されたボタンを選択すると、ユーザー コミットに関する上位統計を提供します。モ

モニタリングするインスタンスでトランザクションコミットが発生したセッション情報と性能に関する主要な指標を表示します。

- 特定sidまたはmoduleでトランザクションが集中しているかを確認できます。
- programカラムを通じて、特定の作業がどのようなプロセスによって実行されたかを把握できます。
- **Top Event:** ウィジェットに表示されたボタンを選択すると、その指標に関する主要イベント待機の情報を提供します。このデータにより、性能低下の原因を特定し、問題が発生するセッションを診断できます。
 - time waited値の高いセッションを確認して、ボトルネック区間を見つけることができます。
 - 特定moduleまたはprogramで待機イベントが集中しているかを確認できます。

⚠ **Top Stat**または**Top Event**ボタンは、提供する指標によっては提供されない場合があります。

4 アクティブセッション / ロックツリー / プロセス情報

リアルタイムで実行中のアクティブセッションおよびロックツリー、プロセス情報を照会できます。ウィジェットのチャートで特定の時間を選択すると、**リアルタイム参照モード**が停止し、選択した時間に実行したセッション情報がテーブルに表示します。テーブルの上には5秒単位でデータを検索できる時間選択ボタンが表示されます。

- **アクティブセッション:** リアルタイムで実行中のセッションとクエリ内容を照会できます。
 - 実行中のセッション項目を選択すると**セッション詳細画面**が表示され、選択したセッションの詳細を確認できます。次の文書を参考にしてください。
 - テーブルの右側にあるquery_textカラムの項目を選択すると、**SQL詳細画面**が表示されます。選択したクエリ(Query)に対するFull TextおよびPlan情報を確認できます。詳細については、次の文書を参照してください。

• ロックツリー

ロックセッションに関する情報やHolderとWaiter情報をリアルタイムで照会できます。テーブルのカラムの詳細については、次の文書を参照してください。

• プロセス情報

データベースサーバーのリソース使用量をリアルタイムで表示できます。

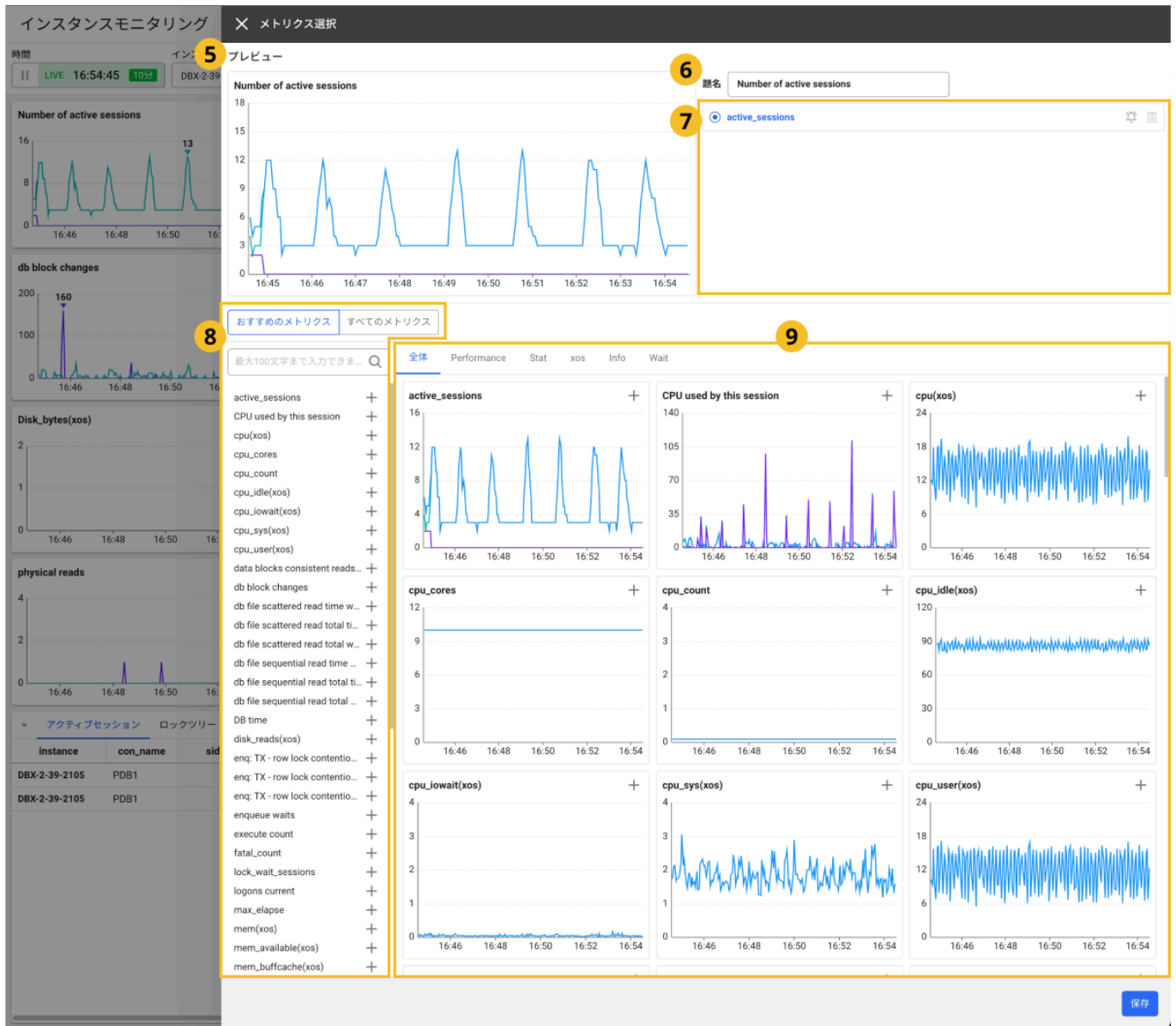
- ✓ 連携プロジェクトを追加すると**アクティブセッション**テーブルで、アクティブセッションと連動したアプリケーションアクティブTXの詳細情報をリアルタイムで確認できます。連携プロジェクトの追加と活用につい

✓ ての詳細は、[次の文書](#)を参照してください。

ⓘ テーブル一覧の文字の色は黒 → オレンジ → 赤順にセッションの実行速度が遅いことを意味します。

メトリクスを選択する

ウィジェットに指標グラフを追加または置換するには、⋮ ボタンを選択します。メトリクス選択画面が表示されます。設定を完了して保存ボタンを選択します。



- **5 プレビュー**：選択した指標のグラフのプレビューを確認できます。
- **6 題名**：ウィジェット名を入力します。
- **7** 現在選択されているウィジェットのチャートで表示されるメトリクス項目が表示されます。 最大4つのメトリクスを選択できます。

① メトリクス項目の右側に🔔ボタンを選択すると、そのメトリクスに基づいてアラート通知を設定できるイ

① **ベント設定** *New* メニューに移動できます。

- **8** **おすすめのメトリクス / すべてのメトリクス**：おすすめのメトリクスを選択すると、データベースモニタリングで頻繁に表示される指標のリストが表示されます。一覧に表示されている指標をウィジェットの指標に変更または追加できます。
- **9** 領域で指標の推移をグラフで確認し、ウィジェットの追加または変更することができます。目的のタブを選択して、メトリクスのリストをカテゴリ別に絞り込むことができます。

指標の追加

ウィジェットに2つ以上の指標を追加できます。最大4つまで追加できます。

1. 指標を追加するウィジェットの右上の **：** ボタンを選択します。
2. **メトリクス選択** ウィンドウが表示されると、**8** メトリクス一覧または **9** グラフ一覧で追加する指標の右側にある **+** ボタンを選択します。
3. 指標をさらに追加する場合は、2の手順を繰り返します。
4. ウィジェットを簡単に識別できるように **6** 題名を変更します。
5. 画面の右下にある **保存** ボタンを選択します。

指標を追加したウィジェットに追加した指標の数に対してチャートが作成されます。ウィジェット名の横にある **①** ボタンを選択すると、追加した指標の名前と内容を確認できます。また、チャートにカーソルを合わせると、ツールチップから数値を確認できます。

❗ **①** ボタンのツールチップ機能は多言語ではサポートされていません。

指標の変更

ウィジェットの指標を別の指標に変更できます。

1. 指標を変更するウィジェットの右上の **：** ボタンを選択します。
2. **メトリクス選択** ウィンドウが表示されると、**7** 領域で変更する指標を選択します。

3. ⑧ メトリクス一覧または ⑨ グラフ一覧から目的の指標を選択します。
4. ⑦ 領域で選択した指標の名前が変更されたことを確認してください。
5. 画面の右下にある**保存**ボタンを選択します。

指標の削除

ウィジェットに複数の指標を設定されている場合は、不要な指標を削除できます。

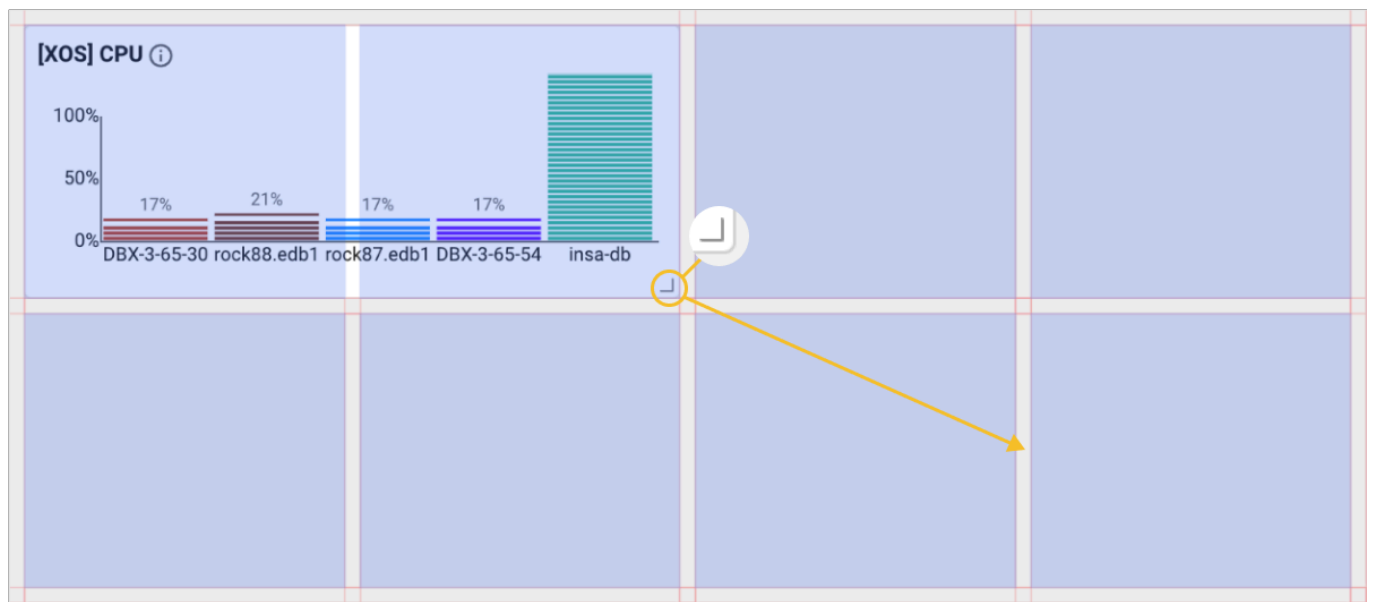
1. 指標を削除するウィジェットの右上の ⋮ ボタンを選択します。
2. **メトリクス選択**ウィンドウが表示されると、⑦ 領域から削除する指標の右側の  ボタンを選択します。
3. 画面の右下にある**保存**ボタンを選択します。

❗ ウィジェットには、少なくとも 1つの指標を選択する必要があります。ウィジェットのすべての指標を削除できません。

ダッシュボードウィジェットの編集

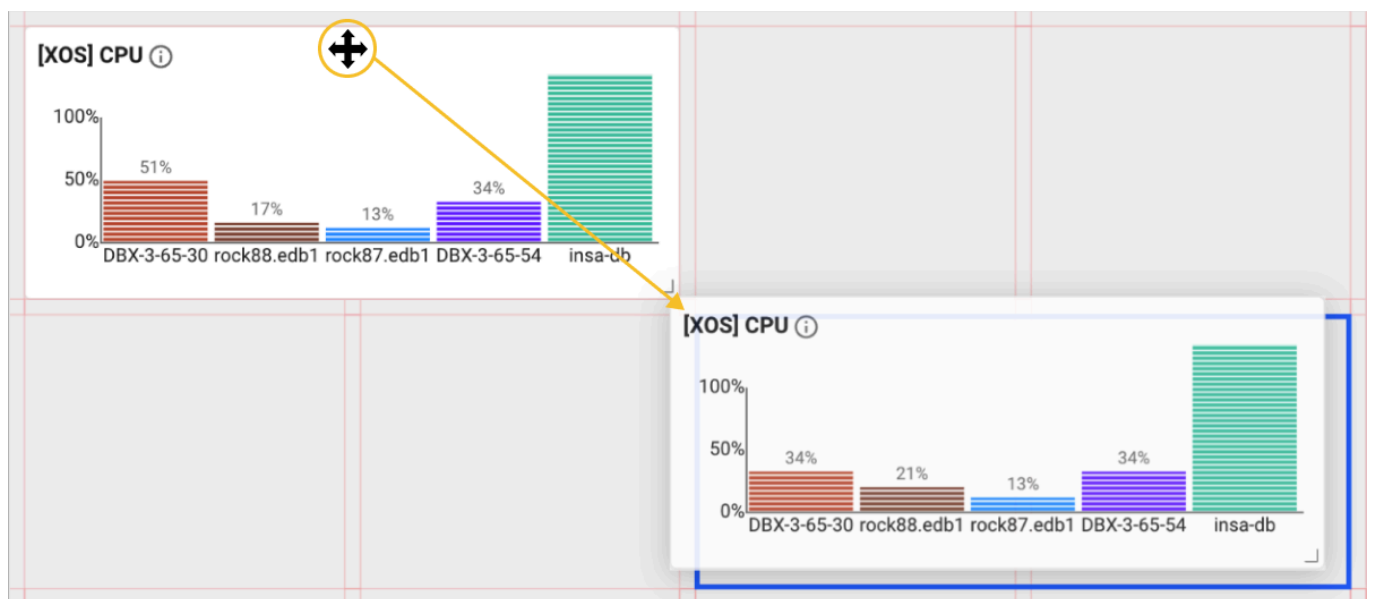
ダッシュボードに配置したウィジェットは、ユーザーが希望するサイズに調整でき、任意の位置に配置できます。るウィジェットは削除したり、再度追加したりすることもできます。

ウィジェットのサイズ変更



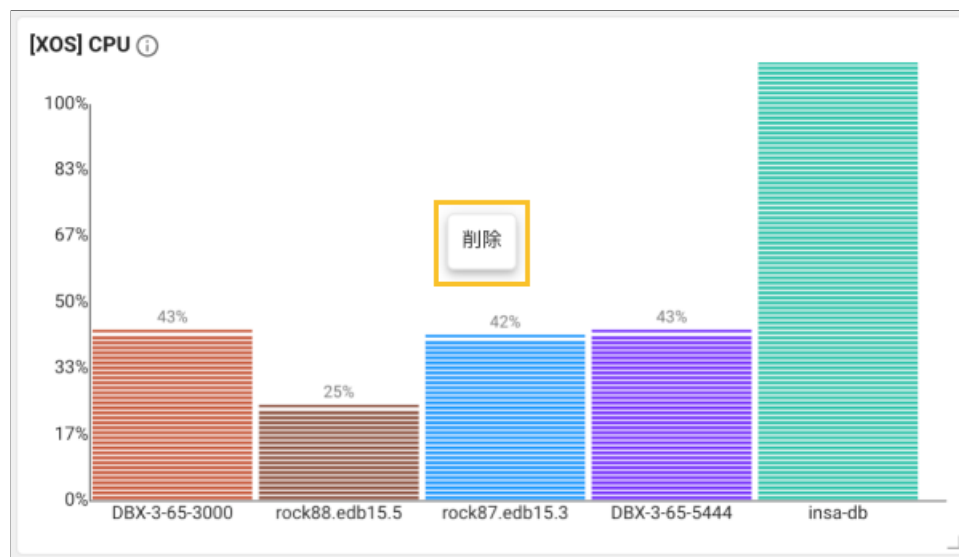
ウィジェットの右下にある「」をマウスでクリックした状態で希望するサイズにドラッグします。幅と高さの比率が均一なグリッドが表示され、グリッド単位でウィジェットのサイズを調整できます。

ウィジェットの移動



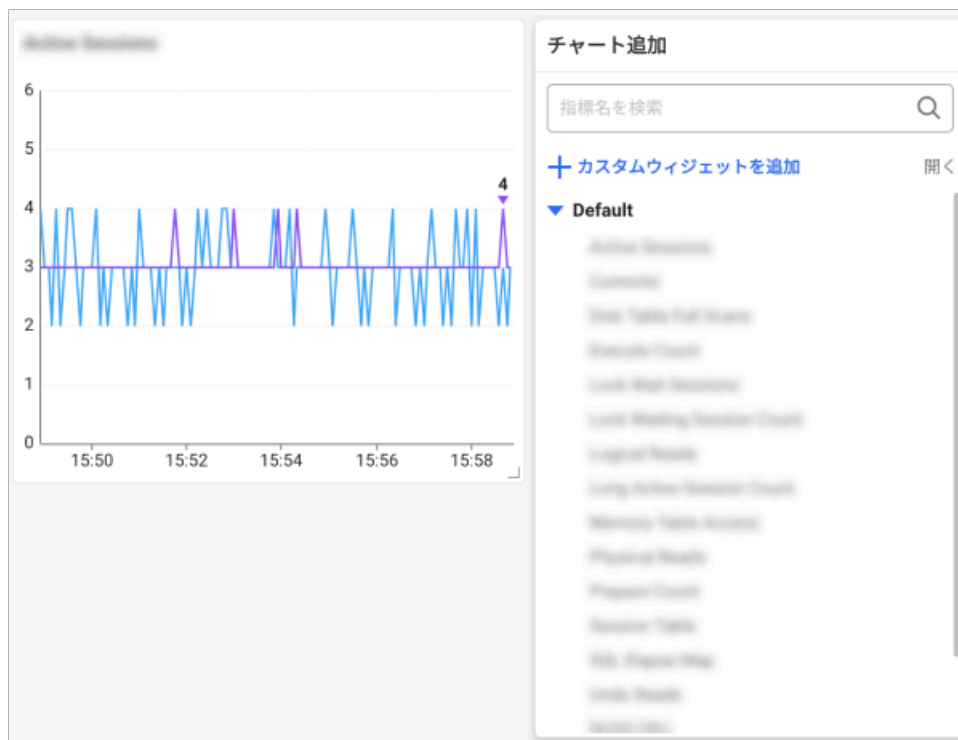
ウィジェットの上部にマウスカーソルを位置すると、カーソルの形が $\oplus$ の形に変更されます。この時、マウスの左ボタンをクリックした状態で希望する位置にドラッグすると、ウィジェットを移動することができます。

ウィジェットを削除する



削除するウィジェットで右クリックします。**削除**ボタンをクリックすると、ウィジェットがダッシュボードから削除されます。

ウィジェットを追加する



ダッシュボードから空白エリアにマウスカーソルを移動し、マウスの右ボタンをクリックします。ポップアップメニューから追加するウィジェットを選択します。希望する場所にウィジェットを配置してサイズを変更します。

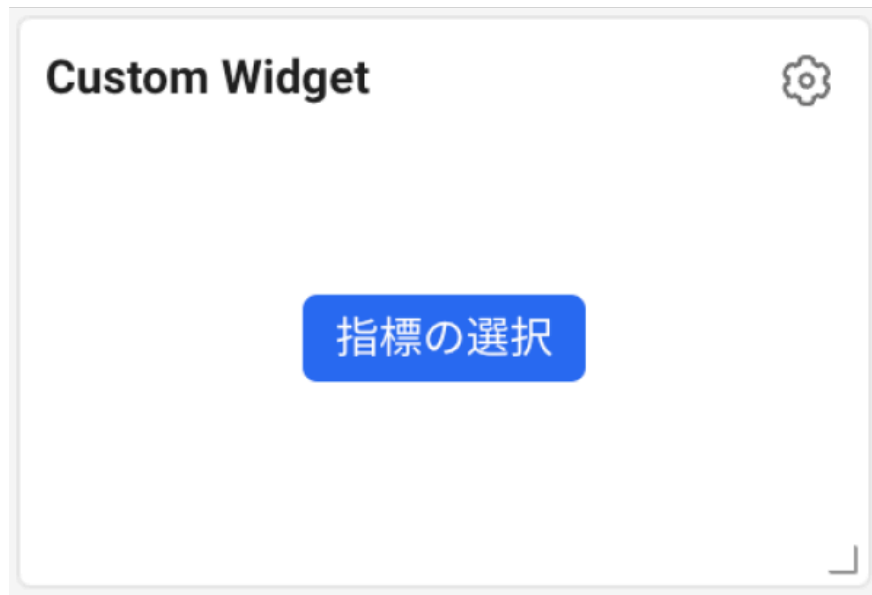
カスタムウィジェット

カスタムウィジェットは、ダッシュボードでデフォルトに提供されるメトリクスの他に、ユーザーがメトリクスを追加できる機能です。デフォルトで提供するメトリクスの他に、ダッシュボードで継続的にモニタリングしたいメトリクスがある場合は、カスタムウィジェット機能を利用できます。

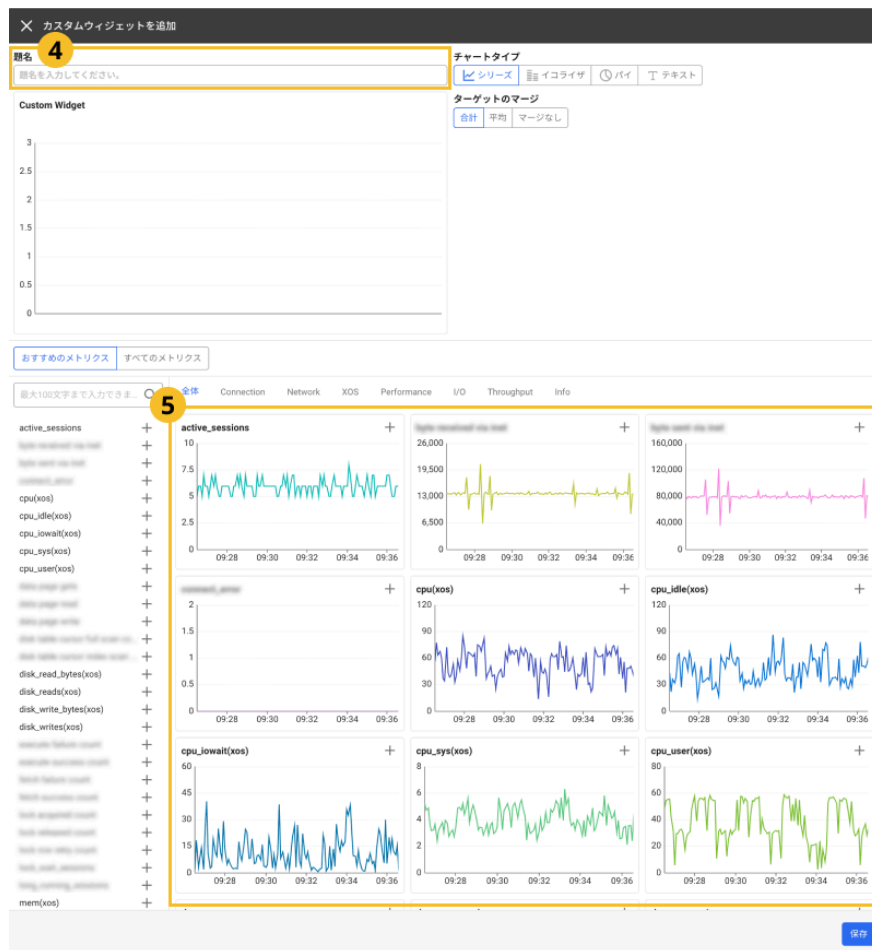
追加する

1. ダッシュボードの空のスペースで、右クリックします。
2. ポップアップメニューで**カスタムウィジェットを追加**を選択します。

3. ダッシュボードに**Custom Widget**が生成されたら、**指標の選択**ボタンを選択します。



4. 画面の右側から **カスタムウィジェット**を追加画面が現れると、**題名項目**にウィジェットの名前を入力します。



5. 画面下のメトリクス一覧から、追加するメトリクスの右上にある＋ボタンを選択します。

6. メトリクスをすべて追加したら**保存**ボタンを選択します。

- ① 最大4つのメトリクスを追加できます。メトリクスを追加する時は、＋ボタンを選択します。
- メトリクスを変更する時は、メトリクス一覧で別のメトリクス項目を選択します。＋ボタンを選択すると、そのメトリクスが追加されます。
- ウィジェットのタイトルを入力しないと、Custom Widgetと保存されます。

チャートタイプを選択する

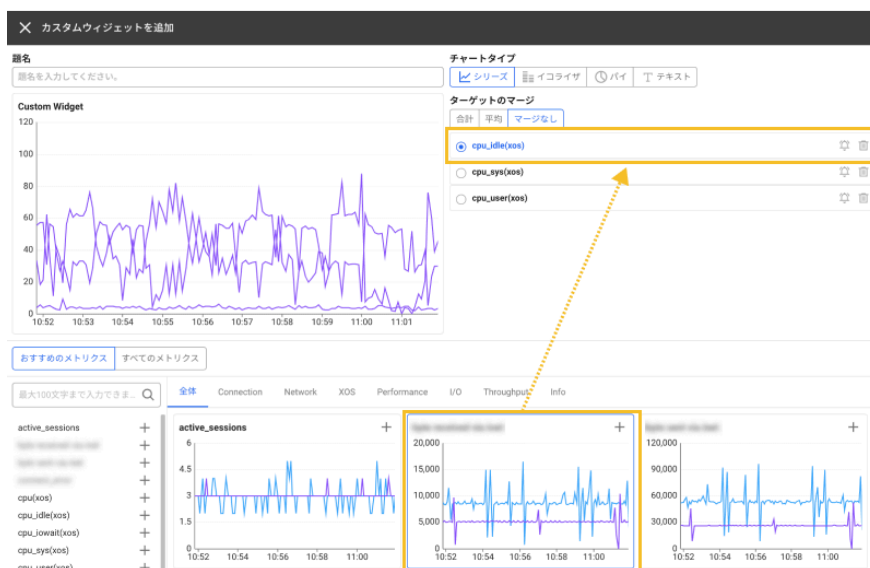
カスタムウィジェットを追加画面のチャートタイプでは、次の形式のチャートを選択できます。

-  シリーズ: 時間の経過によるメトリクスの変化を確認できる時系列形式のチャートです。
-  イコライザ: 複数のメトリクスの性能を比較できる棒グラフ形式のチャートです。様々なメトリクスを1つの画面に視覚的に表現することで、全体的な性能状態を簡単に把握できます。
-  パイ: 複数のメトリクスの全体に対する部門構成比率を確認するのに役立つ円形チャートです。
-  テキスト: メトリクスに対する数値をテキスト形式で表示します。

指標の変更

ユーザー定義ウィジェットに追加したメトリクスを、他のメトリクスに変更または追加できます。

1. ダッシュボードに追加したユーザー定義ウィジェットの右上に  ボタンを選択します。
2. 画面の右側でカスタムウィジェットを追加画面が表示されたら、変更するメトリクスを選択します。



3. 画面下のメトリクスの一覧で変更するメトリクスを選択します。メトリクスを追加する場合、追加するメトリクスの右上に  ボタンを選択します。
4. メトリクスを全て変更したのであれば保存ボタンを選択します。

✓ 追加または変更したメトリクス一覧に表示されたアイコンボタンの機能は、次のとおりです。

-  : 該当メトリクスをカスタムウィジェットから除外できます。
-  : 該当メトリクスに対するアラート通知を設定できます。 ボタンを選択すると、**アラート通知 > イベント設定**メニューに移動します。 **メトリクスイベント**設定の詳細については、[次の文書](#)を参考にしてください。

ウィジェットオプション

ウィジェットに表示されているアイコンボタンの機能は次のとおりです。

-  : 主要ウィジェットに関する機能と情報が確認できます。(多言語に対応予定)
-  /  : チャートの縦軸範囲を拡大または縮小できます。
-  : 該当ウィジェットのデータを広い画面に開いて表示します。
-  : 該当ウィジェットのデータをエージェントごとに区分して照会できる詳細画面が表示されます。

ⓘ ウィジェットによって使用可能なオプションは異なる場合があります。

プリセット

ユーザーが設定したウィジェットの設定とアクティブセッションテーブルの配置状態をプリセットで保存して読み込むことができます。 **Default**プリセットは、変更できません。



プリセット保存

プリセット一覧から保存した項目を選択して、設定を呼び出すことができます。

1. ウィジェットの指標を追加または変更します。
2. 画面の右上にボタンを選択します。
3. **プリセット保存**で名前を変更した上で**保存**ボタンを選択します。

- ❗
- **Default**プリセットは、変更できません。
 - データベース環境によって、一部のプリセットはサポートされない場合があります。
 - ユーザー定義プリセットは、変更した指標グラフだけでなく、アクティブセッションのカラム順序、カラム幅、フィルター適用内容も一緒に適用されます。
 - **プロジェクト修正権限**のあるメンバーのみがプリセットを保存または変更できます。メンバー権限の詳細については、[次の文書](#)を参照してください。

プリセットを作成する

1. ダッシュボードから、希望する形にウィジェットを配置してみてください。サイズを変更して、頻繁に確認するウィジェットのみを配置することもできます。
2. 画面の右上にアイコンをクリックします。
3. 新しいプリセット名を入力します。
4. **保存**ボタンをクリックします。プリセット一覧から新しく保存したプリセットを確認できます。

- ❗
- 新しく作成したプリセットに変更が加えられた場合は、再度プリセットを保存してください。アイコンをクリックし、同じ名前でもプリセットを保存します。既存のプリセットに変更を上書きします。
 - ダッシュボードの変更を保存せずに別のメニューに移動すると、変更は保存されません。
 - プリセットはプロジェクト単位で保存され、他のユーザーと共有できます。

プリセット削除

画面の右上ある**プリセット**を選択すると、プリセット一覧が表示されます。削除するプリセット項目の右側にある **×** ボタンを選択します。

アクティブセッションのセクションを活用

画面の下にあるアクティブセッションセクションからリアルタイムで実行中のセッション情報を確認できます。ここでは、テーブルの右上にあるボタン機能を説明します。

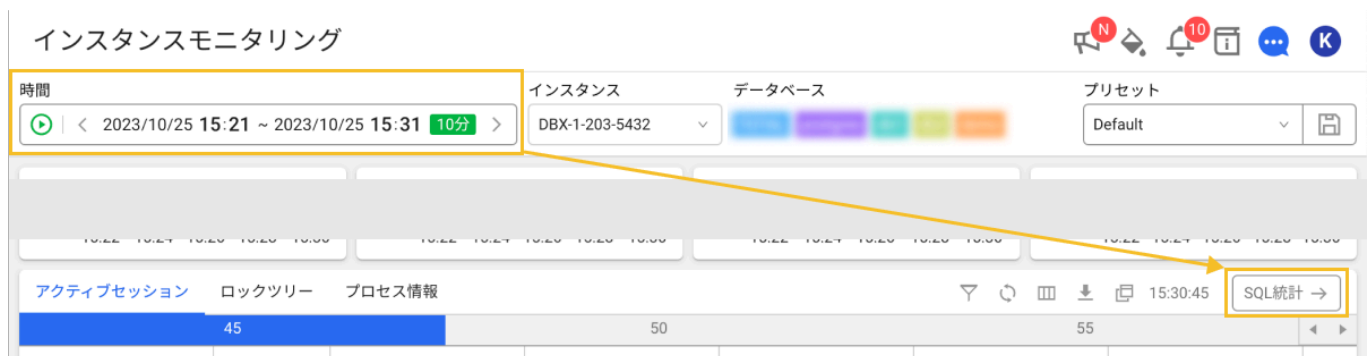
- ：テーブルのカラムヘッダー項目に基づいて一覧をフィルタリングできます。
- ：テーブルのリストを更新できます。
- ：テーブル内のカラムヘッダー項目を表示または非表示にすることができます。
- ：稼働中のセッションを停止できます。

① この機能は、**修正権限のあるメンバー**のみ利用できます。メンバー権限に関する詳細な説明は[次の文書](#)を参照してください。

- ：テーブルの内容をCSVファイルとしてダウンロードできます。
- ：別ウィンドウで開き、より大きな画面でリストを確認できます。

SQL統計を確認する

過去の特定時間の間に発生した主要性能指標の推移とともに、その間のTop SQL一覧が照会できます。

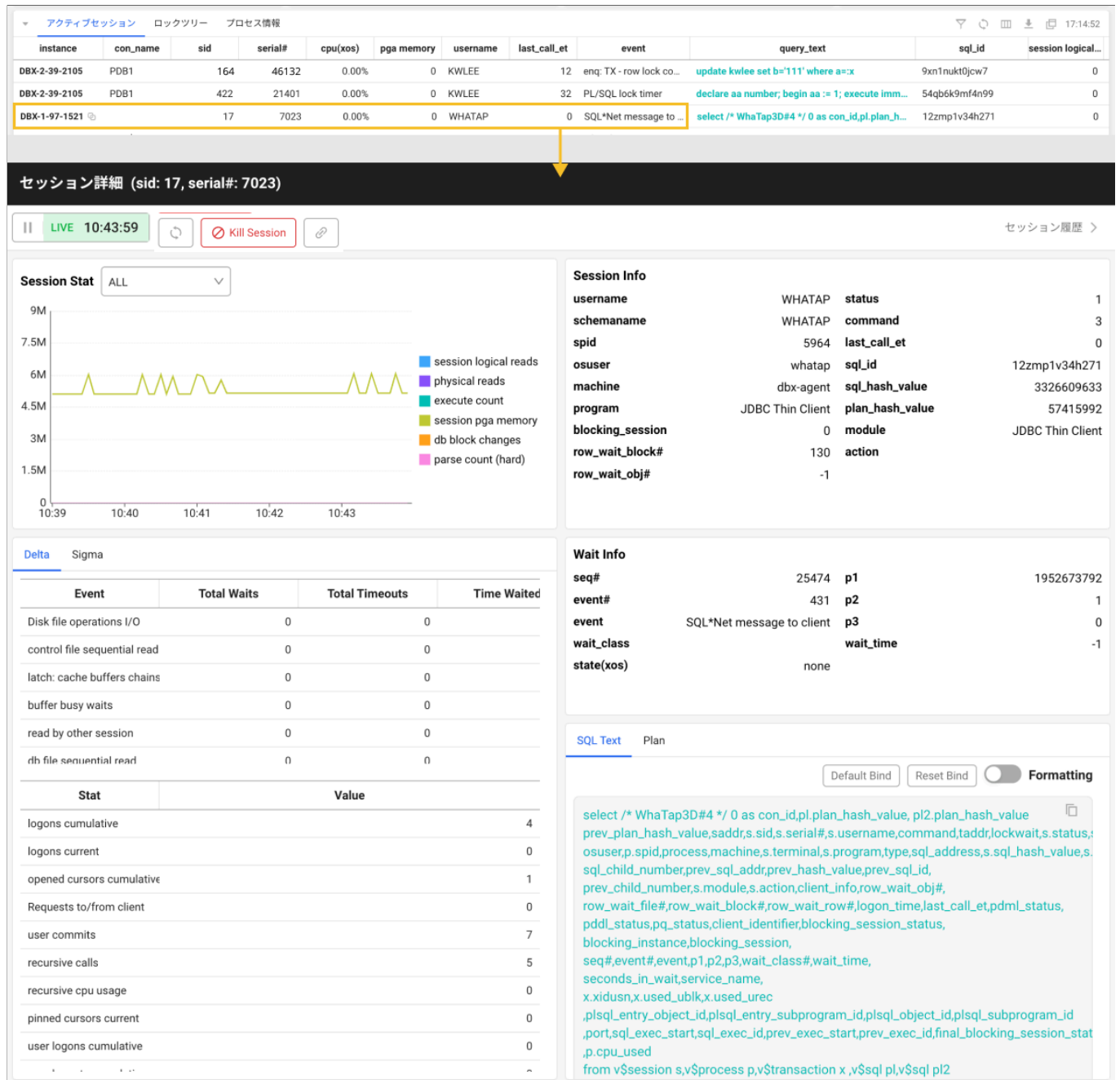


1. 時間から  ボタンを選択して、リアルタイムモードから非リアルタイムモードに切り替えてください。
2. 時間で照会したい時間帯を指定してください。
3. アクティブセッション領域の右上のSQL統計ボタンを選択します。

新しい画面とともに**SQL統計**メニューへ移動します。 **インスタンスメニュー**で設定した時間の間に実行されたTop SQL一覧を確認することができます。 **SQL統計詳細**については、 [次の文書](#)を参照してください。

セッション詳細情報を確認する

アクティブセッションタブでセッション項目を選択すると**セッション詳細**画面が表示されます。 選択したセッションについての詳細を確認できます。

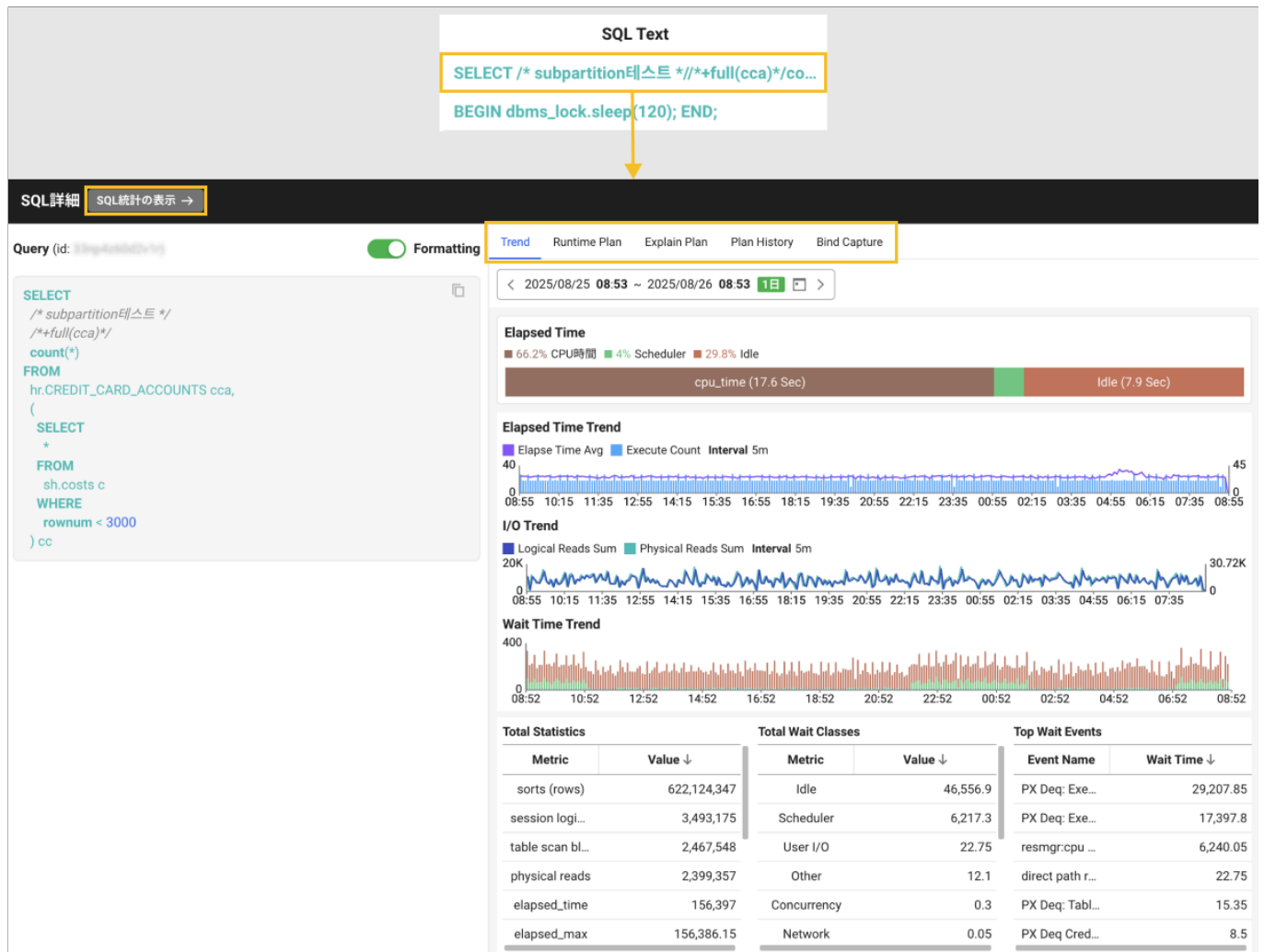


クエリの詳細を見る

現在実行中のセッションまたは過去セッションの SQL クエリ情報を確認するには、**SQL_Text** カラム項目を参照してください。

SQL クエリ文とプラン情報を確認できる **SQL詳細** ウィンドウが表示されます。

SQL統計の表示 → ボタンをクリックすると、該当 SQL クエリ文に関連する統計情報を確認できる **SQL統計** メニューに移動できます。



SQL 詳細 ウィンドウのタブは次のとおりです。

- **Trend:** 特定 SQL の実行時間、I/O、待機イベントを総合的に分析し、パフォーマンスボトルネックを迅速に確認できます。
- **Runtime Plan:** 選択した SQL クエリの実行計画とランタイム情報を提供します。実行回数、平均実行時間、平均物理読み取りなどの詳細情報を確認できます。
- **Explain Plan:** オプティマイザが予測した実行計画を表示します。コスト、操作、オブジェクト名、カーディナリティなどの情報を提供します。

- **Plan History:** データベースで実行された SQL クエリの実行計画に関する履歴を確認できます。
- **Bind Capture:** データベースで実行された SQL クエリに使用されたバインド変数の値を確認できます。これによりクエリ実行の実際の内容を把握できます。

- ❗ • Planツリーで🔍ボタンを選択すると、チューニングガイドを確認できます。チューニングガイドは、Plan Treeノードが TABLE ACCESS (FULL) / INDEX (SKIP SCAN) である場合とノードに CARTESIAN キーワードが含まれている場合、確認できます。

連携プロジェクトを確認する

連携プロジェクト管理メニューで別のアプリケーションプロジェクトを追加すると、アクティブセッション項目に🔗ボタンが追加されます。

🔗ボタンを選択すると表示される**連携分析**画面を使用すると、ユーザーはアクティブセッションと連動したアプリケーションのアクティブTX情報をリアルタイムで確認できます。トランザクションURLおよびデータベース接続情報、DBタイプ、HTTPメソッド、実行時間などの基本情報とともに、トランザクションID、スレッドID、クライアントIP、CPU使用時間などの具体的なメトリクスを確認できます。また、SQLクエリの実行時間、呼び出されたSQLの数、DB接続時間などもモニタリングでき、システムの性能を総合的に分析できます。

特に、スタックトレースによるコールスタック(Call stack) 情報とSQLクエリを一緒に提供し、問題発生時の詳細な原因を追跡して解決するのに役立ちます。これにより、アプリケーションとデータベース間の相互作用を深く分析し、性能のボトルネック現象を迅速に診断できます。

❗ 連携プロジェクトの追加と活用についての詳細は、[次の文書](#)を参照してください。

進行中のセッションの停止

1. テーブルの右上にある  ボタンを選択します。
2. テーブル一覧から停止するセッションを選択します。複数選択できます。
3. 停止ボタンを選択します。
4. セッション中止画面が表示されると、パスワード(**Parameter Key**)を入力します。パスワードは、エージェントのインストールディレクトリにある [paramkey.txt] で確認できます。
5. 適用ボタンを選択します。選択したセッションが停止します。

- ❗
- DBXエージェントのインストールパスの [paramkey.txt](#) ファイルからパスワードを確認できます。
 - この機能は、**修正権限**のあるメンバーのみ利用できます。メンバー権限の詳細については、[次の文書](#)を参照してください

グラフチャートセクションを拡大する

現在見ている画面が小さくてウィジェットを確認しづらい場合は、より広い画面でグラフチャートウィジェットを確認することができます。画面下段に**アクティブセッション**セクションの ▾ ボタンを選択します。**アクティブセッション**セクションは縮小され、グラフチャートウィジェットセクションは拡大します。

グラフチャートウィジェットセクションを縮小するには、画面下部に**アクティブセッション**セクションの ▲ ボタンを選択します。

テーブルカラムの設定

テーブルヘッダーカラムを非表示にしたり、希望する項目を追加できます。カラムの順序を変更することもできます。 ボタンをクリックします。

- ❗
- 設定を完了した後は、**確認**ボタンを選択すると、設定事項がテーブルに反映されます。

- ③ 検索バーにテキストを入力し、カラム項目を検索できます。入力したテキストと一致するカラム項目が表示されます。
- 画像は、製品またはプロジェクト、メニューによって異なる場合があります。

カラムを追加する

① 一覧でテーブルヘッダーカラムに追加する項目を選択します。すべての項目を追加するには、**すべて選択**を選択します。

カラムを削除する

① 一覧から削除するカラム項目のチェックボックスのチェックを解除します。または② 一覧から削除する項目の右側の×ボタンを選択します。

カラムの順序の変更

② 一覧で順序を変更するアイテムをドラッグして、希望する位置に移動できます。

設定事項の初期化

変更事項をすべてキャンセルし、初期化するには、🔄 **初期化**ボタンを選択します。

テーブルデータのフィルタリング



1. テーブルの右上にある▽ボタンを選択します。
2. テーブルのカラムヘッダー項目と条件を選択します。
3. 条件を入力してください。テキストボックスに条件を入力します。
4. 保存ボタンを選択します。

カラム情報の案内

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

カラム情報に関する詳しい内容は、[次のリンク](#)を参照してください。

項目	説明
username	DBユーザー名
command	COMMAND_TYPE と同一、SQLのcommand number
status	セッションの状態
schemaname	スキーマ、usernameと同一
osuser	セッションに接続したOSのユーザー名
spid	OSプロセスの識別子
process	セッションに接続したクライアントプロセスID
logon_time	セッションが作成された時点
last_call_et	セッションがアクティブになっている場合、アクティブになってから経過した時間(秒)
sql_text	query、実行されているstatement、リテラル文字列を除外したSQL文
sql_param	query から分離したリテラル値
prev_sql_text	以前に実行したquery

項目	説明
prev_sql_param	以前に実行したquery param
program	クライアントで実行したプログラムの情報
type	session type - user background recursive (whatap.conf)で設定可能
seq#	現在または最後の待機を固有に識別する番号(待機するたびに増加)
event	セッションがキューに登録されている場合は、セッションが待機しているリソースまたはイベントの数
p1	最初の待機イベントのパラメータ(小数点)
p2	2番目の待機イベントのパラメータ(小数点)
p3	3番目の待機イベントのパラメータ(小数点)
wait_class	待機中のイベントのクラス名
wait_time	セッションが現在待機中の場合、値は0です。0より大きい場合は、最後の待機時間(milliseconds)
seconds_in_wait	セッションが現在待機中の場合、セッションが待機のために待機した時間を意味します。セッションが待機中でない場合、最後の待機開始からの時間を意味します。
state	待機状態、WAITING, WAITED UNKNOWN TIME, WAITED SHORT TIME, WAITED KNOWN TIME
machine	セッションに接続したクライアントサーバー名(hostname)
port	クライアントポート番号
terminal	端末名
module	クライアントで実行したプログラムでユーザが設定したモジュール(module)名

項目	説明
アクション	現在実行されているモジュールのユーザーが設定したアクション名
client_info	ユーザーが設定したクライアント情報
client_identifier	ユーザーが設定したクライアントの識別子
session logical reads	バッファキャッシュ内でデータを読み込んだ回数
physical reads	物理ディスクからデータを読み出した回数
excute count	SQLの実行数
parse count (hard)	構文分析の呼び出し合計数(実際の構文分析)
parse count (total)	構文分析呼び出しの総数(hard、soft、describe)
opened cursors current	現在開いているcursorの数
db block changes	データベース内のすべてのブロックの変更回数
session pga memory	セッションの現在のPGAサイズ(size)
undo_segid	undo情報id
undo_blk	undoブロックの数
undo_rec	undoレコードの数
sql_address	SQL文(statement)を識別する情報
sql_hash_value	SQL文(statement)を識別する情報
sql_id	現在実行中のSQL文(statement)のSQL識別子
sql_child_number	現在実行中のSQL文(statement)の下位番号

項目	説明
sql_exec_start	当該セッションで現在実行中のSQL実行は、開始点にあります。
sql_exec_id	SQL実行識別子
prev_sql_addr	最後に実行されたSQL文(statement)を識別する情報
prev_child_number	最後に実行されたSQL文(statement)の下位番号
prev_exec_start	最後に実行されたSQL文(statement)のSQL実行の開始時点
prev_exec_id	最後に実行されたSQL文(statement)のSQL実行識別id
plsql_entry_object_id	スタックの一番上にあるPL/SQL下位プログラムのオブジェクトid
plsql_entry_subprogram_id	スタックで一番上にあるPL/SQL下位プログラムのサブプログラムid
plsql_object_id	現在実行中のPL/SQLオブジェクトのサブプログラムid
taddr	トランザクションのアドレス(address)
lockwait	lock wait address、セッションが待機中のロックアドレス
row_wait_obj	ROW_WAIT_ROW# で指定された行を含むテーブルのオブジェクトID
row_wait_file	ROW_WAIT_ROW# で指定された行を含むデータファイルの識別子
row_wait_block	ROW_WAIT_ROW# で指定された行を含むブロックの識別子
row_wait_row	現在ロックされている行
pdml_status	値がENABLEDの場合、セッションがPARALLEL DML有効モードです。値がDISABLEの場合、セッションでPARALLEL DML有効モードに対応していません。値がFORCEDの場合、セッションがPARALLEL DMLを強制的に使用するように変更されたものです。
pq_status	値がENABLEDの場合、セッションがPARALLEL QUERY有効モードです。値が

項目	説明
	DISABLE の場合、セッションで PARALLEL QUERY 有効モードに対応していません。値が FORCED の場合、セッションが PARALLEL QUERY を強制的に使用するように変更されたものです。
blocking_session_status	ブロックしているセッションの存在有無に関する詳細を提供
blocking_instance	ブロックしているセッションのインスタンス識別子
blocking_session	ブロックしているセッションのセッション識別子
final_blocking_session_status	最終ブロックセッションの存在有無に関する詳細を提供
final_blocking_instance	最終ブロックしているセッションのインスタンス識別子
final_blocking_session	最終ブロックしているセッションのセッション識別子
service_name	セッションのサービス名やDB名と類似
saddr	セッションアドレス
con_id	データに関連するコンテナのid(マルチデータベースをサポートする場合のみ)

ⓘ WhaTapは、クライアントに関する情報をデフォルトで保存します。

項目	説明
Instance	インスタンス（エージェント）名
sid	セッションID
serial#	セッションの一意な識別子と共に使用するセッションシリアル番号
holder_type	ロックを保持しているセッションのロックタイプ（例：TX、TMなど）

項目	説明
holder_mode	保持中のロックモード（例：0～6、6はexclusive）
waiter_type	ロックを待機しているセッションのロックタイプ
waiter_mode	現在要求中のロックモード
id1	ロック識別子1（ロック対象オブジェクトを識別するために使用、例：トランザクションIDなど）
id2	ロックの詳細情報またはバージョン情報を表します

次の項目は、XOSエージェントをインストールした場合に収集される指標です。

項目	説明	単位
pid	プロセスID	-
cpuusage	プロセスが使用するCPU使用率	%
cputime	プロセスがこれまでに使用した累積CPU時間	秒
elapsed	プロセスが実行されてからの経過時間	秒
vsize	Virtual memory size. 全体の仮想メモリ使用量（RAM + Swap + 予約）	KB
rss	Resident Set Size. 現在のプロセスが占有している実際の物理メモリサイズ	KB
pss	Proportional Set Size. 共有メモリを含む実際の物理メモリ使用量	KB
state	プロセスの現在の状態コード（例：R、S、D、T、Z など）	-
ioread	累積読み込みバイト数（read_bytes）	バイト（B）
iowrite	累積書き込みバイト数（write_bytes）	バイト（B）
uid	このプロセスを実行したユーザーID	-

項目	説明	単位
cmd	実行されたコマンドの短縮形式	-
longcmd	実行されたコマンドの完全な形式（フルパスおよび引数を含む）	-

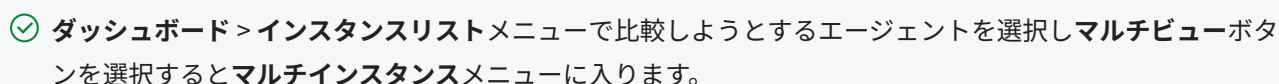
マルチインスタンス

ホーム画面 > プロジェクト選択 > ダッシュボード > マルチインスタンス

インスタンスモニタリングメニューでは、1つのエージェント(インスタンス)だけをモニタリングすると、マルチインスタンスは、複数のエージェント(インスタンス)を同時に監視して管理できる機能を提供します。

各インスタンスで発生する主な性能指標を、一目で比較して分析できます。実行中の各エージェントの数、すべてのアクティブセッションの合計、トランザクション数などのデータを集計して、効果的にモニタリングできます。ユーザーは、グラフチャートを通じて時間によるメトリクス値の変化と各インスタンスの高い性能メトリクスを簡単に確認できます。

また、最も長く続くアクティブセッションに関する情報を確認でき、ユーザーはシステム性能のボトルネック現象を診断し、解決できる根拠資料として活用できます。



照会時間設定

ダッシュボードではリアルタイムモニタリング機能を基本的に提供しますが、画面上部のタイムセクターを利用して過去時間のデータを照会できます。タイムセクターで || ボタンを選択して時間を設定します。ユーザーが設定した時間に基づいて、ダッシュボードに配置したウィジェットのデータを更新します。

エージェント確認

エージェント接続状態を確認する



画面の左上のタイムピッカーの右側には、プロジェクトに関連したエージェントの状態を確認できる情報が表示されます。この機能で、モニタリング対象サーバーの動作有無がすぐに確認できます。

- **Total**: プロジェクトに関連付けられているすべてのエージェントの数
- **Active**: 活性化されたエージェントの数
- **Inactive**: 非活性化状態のエージェント数
- : 非活性化されたエージェントを表示または非表示します。

エージェント別モニタリング



基本的に、ダッシュボードにはすべてのエージェントから収集した指標をチャートとして表示しますが、エージェントを選択してデータを表示することもできます。タイムピッカーの下にあるエージェントを1つ以上選択します。選択したエージェントの指標でダッシュボードに配置されたウィジェットのデータを更新します。

- ✔ エージェントを1つ以上選択した状態で再度すべてのエージェントを選択するには、選択を解除するか、[Total](#)をクリックします。

- ❗ プロジェクトに接続されたエージェントの数が多い場合は、エージェント名を短く設定することをお勧めします。エージェント名の設定に関する詳細は、[次の文書](#)を参照してください。

ウィジェット編集

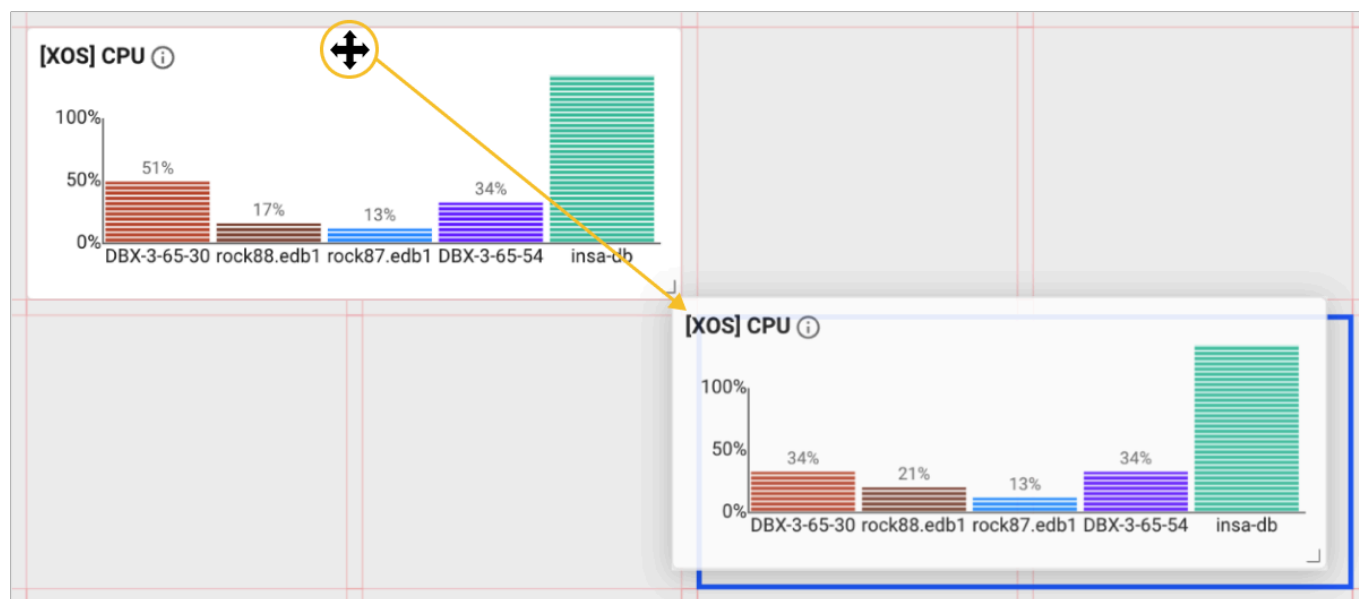
ダッシュボードに配置したウィジェットは、ユーザーが希望するサイズに調整でき、任意の位置に配置できます。ウィジェットは削除したり、再度追加したりすることもできます。

ウィジェットのサイズ変更



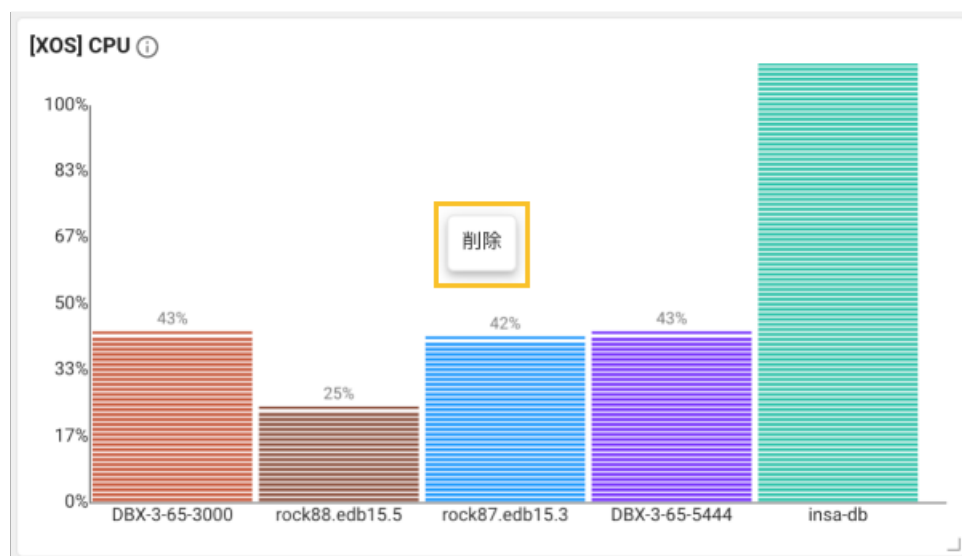
ウィジェットの右下にある「L」をマウスでクリックした状態で希望するサイズにドラッグします。幅と高さの比率が均一なグリッドが表示され、グリッド単位でウィジェットのサイズを調整できます。

ウィジェットの移動



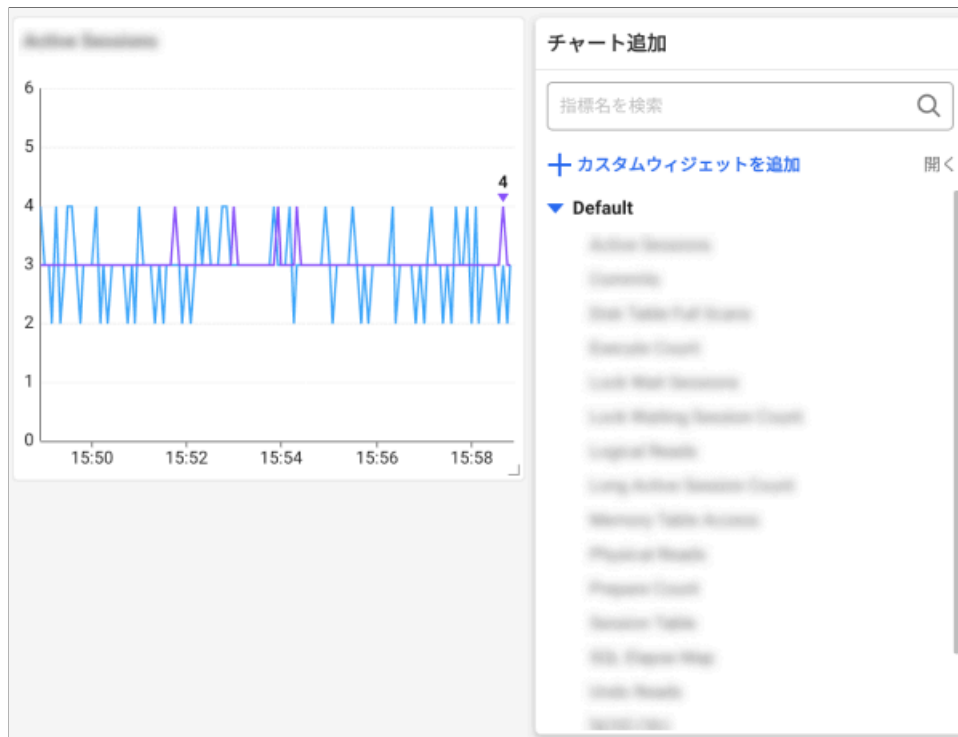
ウィジェットの上部にマウスカーソルを位置すると、カーソルの形が $\oplus$ の形に変更されます。この時、マウスの左ボタンをクリックした状態で希望する位置にドラッグすると、ウィジェットを移動することができます。

ウィジェットを削除する



削除するウィジェットで右クリックします。削除ボタンをクリックすると、ウィジェットがダッシュボードから削除されます。

ウィジェットを追加する



ダッシュボードから空白エリアにマウスカーソルを移動し、マウスの右ボタンをクリックします。ポップアップメニューから追加するウィジェットを選択します。希望する場所にウィジェットを配置してサイズを変更します。

- ❗ • ダッシュボードに配置できるウィジェットの詳細については、[次の文書](#)を参考にしてください。
- ダッシュボードで基本的に提供されるメトリクスの他に、ユーザーが希望するメトリクスを追加できます。ユーザー定義ウィジェットの詳細については、[次の文書](#)を参照してください。

カスタムウィジェット

カスタムウィジェットは、ダッシュボードでデフォルトに提供されるメトリクスの他に、ユーザーがメトリクスを追加

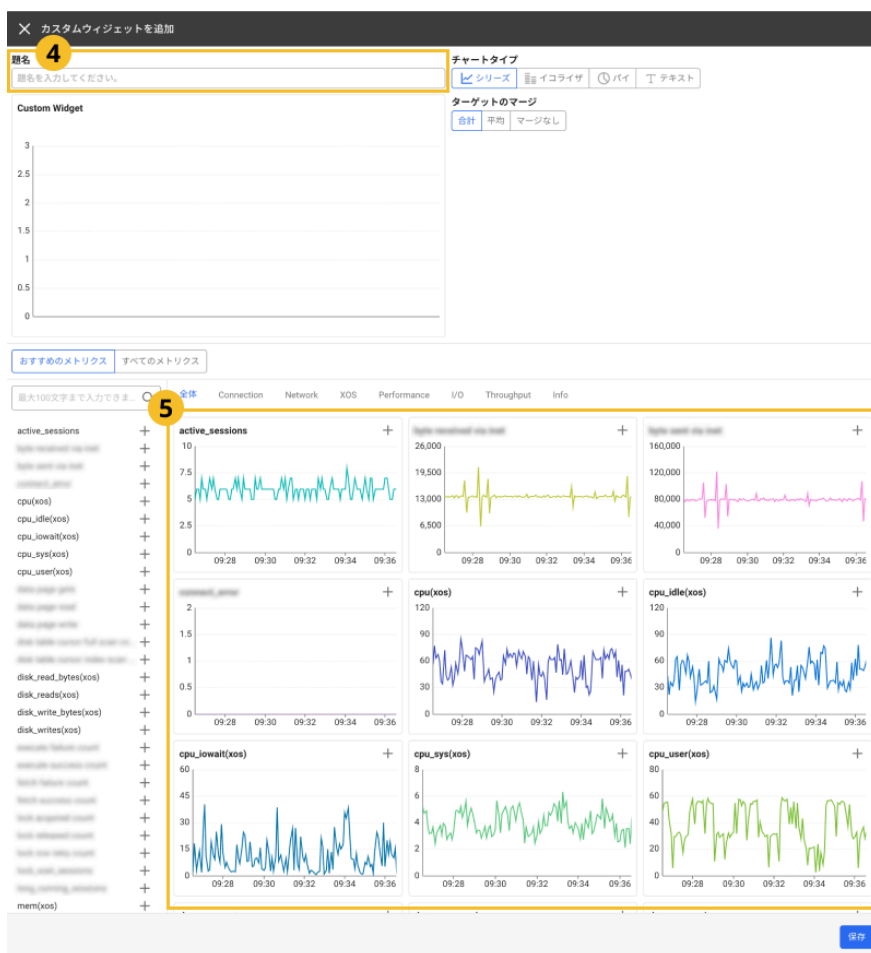
できる機能です。デフォルトで提供するメトリクスの他に、ダッシュボードで継続的にモニタリングしたいメトリクスがある場合は、カスタムウィジェット機能を利用できます。

追加する

1. ダッシュボードの空のスペースで、右クリックします。
2. ポップアップメニューで**カスタムウィジェットを追加**を選択します。
3. ダッシュボードに**Custom Widget**が生成されたら、**指標の選択**ボタンを選択します。



4. 画面の右側から **カスタムウィジェットを追加**画面が現れると、**題名項目**にウィジェットの名前を入力します。



5. 画面下のメトリクス一覧から、追加するメトリクスの右上にある＋ボタンを選択します。

6. メトリクスをすべて追加したら**保存**ボタンを選択します。

- ① 最大4つのメトリクスを追加できます。メトリクスを追加する時は、＋ボタンを選択します。
- メトリクスを変更する時は、メトリクス一覧で別のメトリクス項目を選択します。＋ボタンを選択すると、そのメトリクスが追加されます。
- ウィジェットのタイトルを入力しないと、Custom Widgetと保存されます。

チャートタイプを選択する

カスタムウィジェットを追加画面のチャートタイプでは、次の形式のチャートを選択できます。

-  **シリーズ**: 時間の経過によるメトリクスの変化を確認できる時系列形式のチャートです。
-  **イコライザ**: 複数のメトリクスの性能を比較できる棒グラフ形式のチャートです。様々なメトリクスを1つの画面に視覚的に表現することで、全体的な性能状態を簡単に把握できます。
-  **パイ**: 複数のメトリクスの全体に対する部門構成比率を確認するのに役立つ円形チャートです。
-  **テキスト**: メトリクスに対する数値をテキスト形式で表示します。

データマージ方式を選択する

カスタムウィジェットを追加画面の対象のマージでは、エージェントから収集されたメトリクスを表示する方法を選択できます。

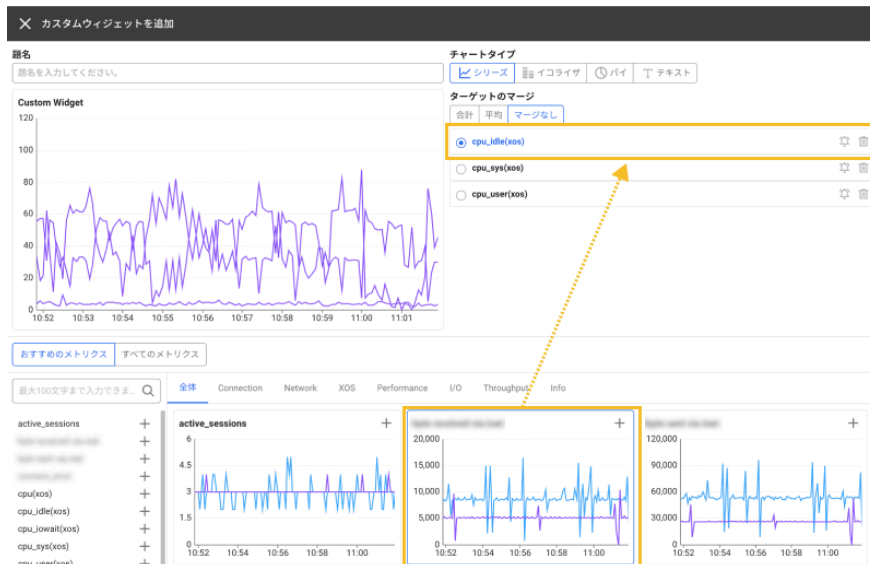
- **合計**: 複数のエージェントから収集されたメトリクス値を合計して、1つの値として表示します。このオプションは、複数のエージェントのデータを統合して全体的な合計を表示する場合に役立ちます。
- **平均**: 複数のエージェントから収集されたメトリクス値を平均して、1つの値として表示します。このオプションは、エージェント全体の平均性能や状態を把握するのに役立ちます。
- **最大値**: 複数のエージェントから収集された指標値の中から最大の値を選択して、一つの値として表示します。このオプションは、エージェントの中で最もパフォーマンスが消耗した状況や最悪の状態を把握するのに役立ちます。
- **マージなし**: 各エージェントのメトリクスを個別に表示します。

 チャートのタイプによって選択できるマージ方式は異なる場合があります。

指標の変更

ユーザー定義ウィジェットに追加したメトリクスを、他のメトリクスに変更または追加できます。

1. ダッシュボードに追加したユーザー定義ウィジェットの右上に  ボタンを選択します。
2. 画面の右側で **カスタムウィジェットを追加画面**が表示されたら、変更するメトリクスを選択します。



3. 画面下のメトリクスの一覧で変更するメトリクスを選択します。メトリクスを追加する場合、追加するメトリクスの右上に+ ボタンを選択します。
4. メトリクスを全て変更したのであれば保存ボタンを選択します。

✓ 追加または変更したメトリクス一覧に表示されたアイコンボタンの機能は、次のとおりです。

- : 該当メトリクスをカスタムウィジェットから除外できます。
- : 該当メトリクスに対するアラート通知を設定できます。ボタンを選択すると、**アラート通知 > イベント設定**メニューに移動します。メトリクスイベント設定の詳細については、[次の文書](#)を参考にしてください。

ウィジェットオプション

ウィジェットに表示されているアイコンボタンの機能は次のとおりです。

- : 主要ウィジェットに関する機能と情報が確認できます。(多言語に対応予定)
- / : チャートの縦軸範囲を拡大または縮小できます。
- : 該当ウィジェットのデータを広い画面に開いて表示します。
- : 該当ウィジェットのデータをエージェントごとに区分して照会できる詳細画面が表示されます。

ⓘ ウィジェットによって使用可能なオプションは異なる場合があります。

プリセット設定

ダッシュボードからユーザーが設定したウィジェットの設定とレイアウト状態を保存して読み込むことができます。ウィジェットのサイズを調整し、希望する場所に配置して新しいプリセットを作成できます。



- **Default** : 主要DBメトリクスとXOSメトリクスで構成されたプリセットです。

- ⓘ
- プリセットに保存できるダッシュボードの設定は次のとおりです。
 - ウィジェットのサイズ、位置
 - ユーザー定義ウィジェットに定義された設定
 - **Session Table**ウィジェットに定義されたテーブルの幅とフィルター、選択したカラム値
 - 基本プリセット(デフォルト)は変更できません。
 - データベース環境によって、一部のプリセットはサポートされない場合があります。
 - **プロジェクト修正権限**のあるメンバーのみがプリセットを保存または変更できます。メンバー権限の詳細については、[次の文書](#)を参照してください。

新しいプリセットを作成する

1. ダッシュボードから、希望する形にウィジェットを配置してみてください。サイズを変更して、頻繁に確認するウィジェットのみを配置することもできます。
2. 画面の右上に  ボタンを選択します。
3. 新しいプリセット名を入力します。



エージェント選択履歴と一緒に保存するには、**エージェント選択履歴を含む**を選択します。

4. 保存ボタンを選択します。

プリセット一覧から新しく保存したプリセットを確認できます。

- ① 新しく作成したプリセットに変更が加えられた場合は、再度プリセットを保存してください。📁ボタンを選択し、同じ名前でプリセットを保存します。既存のプリセットに変更を上書きします。
- ダッシュボードの変更を保存せずに別のメニューに移動すると、変更は保存されません。
- プリセットはプロジェクト単位で保存され、他のユーザーと共有できます。

プリセットを削除する

使用していないプリセットは、プリセット一覧から削除できます。プリセット一覧から削除したい項目の右側に **×** ボタンをクリックします。

- ① **Default** プリセットは削除できません。

主なウィジェットを調べる

ダッシュボードに配置できるウィジェットの種類は次のとおりです。

XOSウィジェット

- [XOS] CPU

エージェント別のCPU使用率をリアルタイムで確認できます。

- [XOS] Memory

エージェント別のメモリ使用率をリアルタイムで確認できます。

- [XOS] Disk Usage

エージェント別のディスク使用率と使用可能サイズ、全体サイズ情報を表示します。

- : **ディスク使用量の詳細**ポップアップが表示されます。時間帯別のディスク使用量を表示するチャートを確認できます。インスタンスごとに確認でき、**Size**、**Free**、**Used**ボタンを選択すると、全体のサイズ、残りのサイズ、使用中のサイズが確認できます。
- : ウィジェットに表示されたテーブルをより広い画面で確認できます。

✔ データが表示されない場合は、`xos.conf`ファイルに次の設定を追加します。

`xos.conf`

```
# disk usage 0ならばoff、単位: 秒
disk_usage_interval=60
```

Databaseウィジェット

提供するウィジェットを通じて、データベース性能分析の主要メトリクスとして、CPU、メモリ、ディスクI/Oなど多様なシステムリソースの状態を確認できます。

- Active Sessions

時間に応じたアクティブセッションの数値を時系列チャートで視覚的に表示します。データベースで実行されているセッションの数を時間ごとに追跡し、グラフ化したウィジェットです。セッションの使用パターンと負荷を視覚的に把握するのに役立ちます。

- Lock Wait Sessions

時間に応じたロック待機セッションの数値を時系列グラフで視覚的に表示します。データベースでロック待機状態にあるセッションの数を時間ごとに追跡し、リアルタイムグラフで表示したウィジェットです。

• Long Active Session Count

データベースで実行される活性セッションの長期実行に関する情報を提供します。セッションの実行時間に応じてグループ化され、時間単位で集計された情報をチャートで表示します。長時間実行されるセッションを識別してデータベースの性能問題やリソース漏れを早期に感知したり、セッション実行時間による分類を通じてどのセッションが長期間にわたり活性状態に維持されているかを視覚的に把握できます。

3秒未満は青、3秒以上10秒未満は緑、10秒以上15秒未満はオレンジ、15秒以上は赤で表示します。

• Long Waiting Session Count

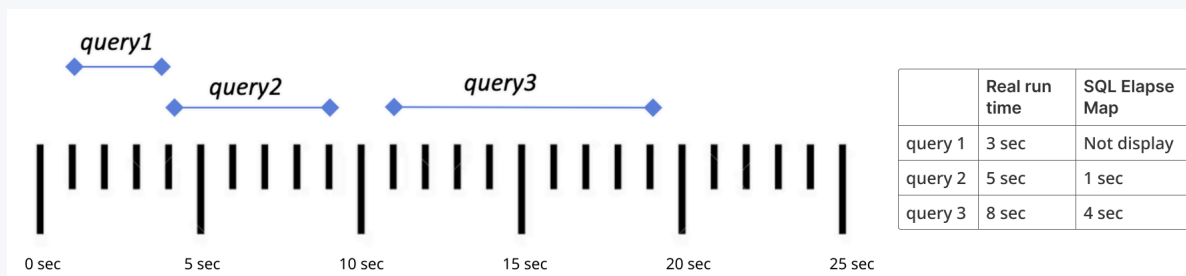
データベースでロックを保持または待機するセッションの長期待機時間に関する情報を提供します。ロック情報と活性セッション情報を組み合わせて、待機中のセッションの長期待機時間を追跡できます。長時間ロックを保持または待機するセッションをモニタリングします。どのセッションが長時間ロックを保持または待機しているかを視覚的に把握できます。

5秒未満は青、5秒以上10秒未満は緑、10秒以上60秒未満はオレンジ、60秒以上は赤で表示します。

• SQL Elapse Map

実行中のクエリの経過時間をリアルタイムチャートで表示します。チャートの特定の領域をドラッグするとQuery List一覧を確認できます。へまたはvボタンを選択して、縦軸の範囲を拡大または縮小できます。

- ① クエリの実行時間は、アクティブセッションを収集して計算します。その結果、アクティブセッションの収集サイクルによって誤差範囲が発生する可能性があります。次の図を参照してください。収集サイクルの間に終了したSQL実行時間は計算されません。収集サイクルが短いほど誤差範囲は小さくなりますが、DBの負荷を考慮して一般的に5秒に設定されています。長時間にわたって実行されるクエリをモニタリングする場合に役立ちます。SQL統計メニューのデータは同じです。



• Wait Event

各バックエンドプロセスが、現在待機中の待機イベント名とそのイベントの発生頻度をリアルタイムのグラフで表示します。主にApplication、Configuration、Administrative、Networkなどに関連する項目の待機イベントをモニタリングします。各待機イベントの発生頻度を視覚的に確認でき、グラフの上にマウスオーバーすると、ツールチップが表示され、各待機イベントに関する詳細情報を提供します。

- **Physical Reads**

データベースで発生する物理的な読み取り操作(physical reads)をモニタリングするウィジェットです。主にデータベースがディスクからデータを読み込む作業に関する統計を提供し、データベースのI/O性能をモニタリングします。

- **Excute Count**

データベースで実行されたクエリまたはプロシージャの実行回数をモニタリングするウィジェットです。データベースによって実行されたクエリやプロシージャの実行回数をリアルタイムで追跡し、ユーザーに提供します。

- **Session Logical Reads**

データベースセッションで実行された論理的な読み取り操作(logical reads)をモニタリングするウィジェットです。データベースセッションで実行された論理的な読み取り操作の回数をリアルタイムで追跡し、ユーザーに提供します。

Session Tableウィジェット

画面下段の[Session Table](#)ウィジェットでは、リアルタイムで実行されているアクティブセッションとロックツリー情報を照会できます。

- **アクティブセッション**：リアルタイムで実行中のセッションとクエリ内容を照会できます。

- 実行中のセッション項目を選択すると[セッション詳細画面](#)が表示され、選択したセッションの詳細を確認できます。[次の文書](#)を参考にしてください。
- テーブルの右側にある[query_text](#)カラムの項目を選択すると、[SQL詳細画面](#)が表示されます。選択したクエリ(Query)に対するFull TextおよびPlan情報を確認できます。詳細については、[次の文書](#)を参照してください。

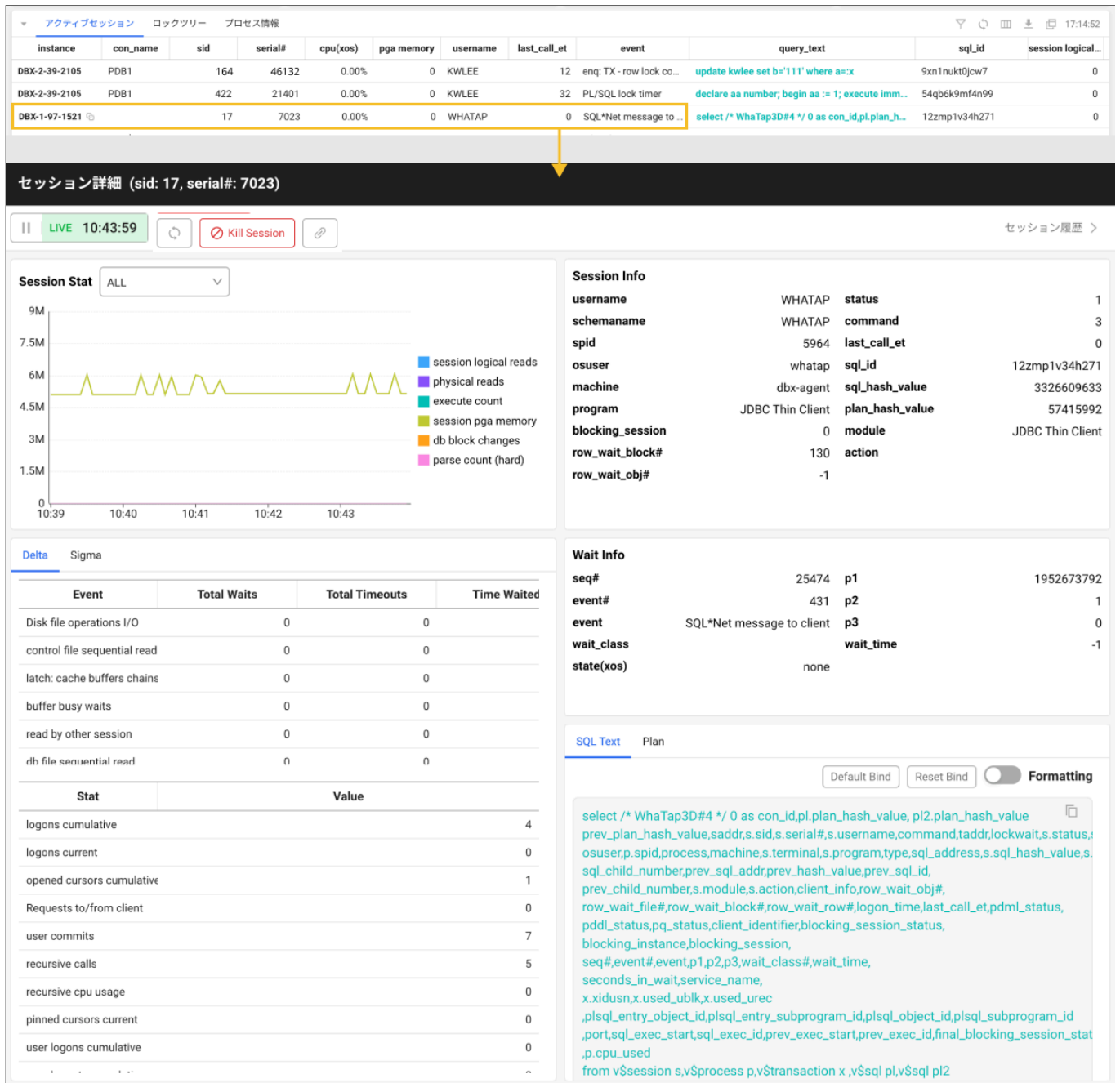
- **ロックツリー**

ロックセッションに関する情報やHolderとWaiter情報をリアルタイムで照会できます。テーブルのカラムの詳細については、[次の文書](#)を参照してください。

❗ テーブル一覧の文字の色は黒 → オレンジ → 赤順にセッションの実行速度が遅いことを意味します。

セッション詳細情報を確認する

アクティブセッションタブでセッション項目を選択するとセッション詳細画面が表示されます。選択したセッションについての詳細を確認できます。

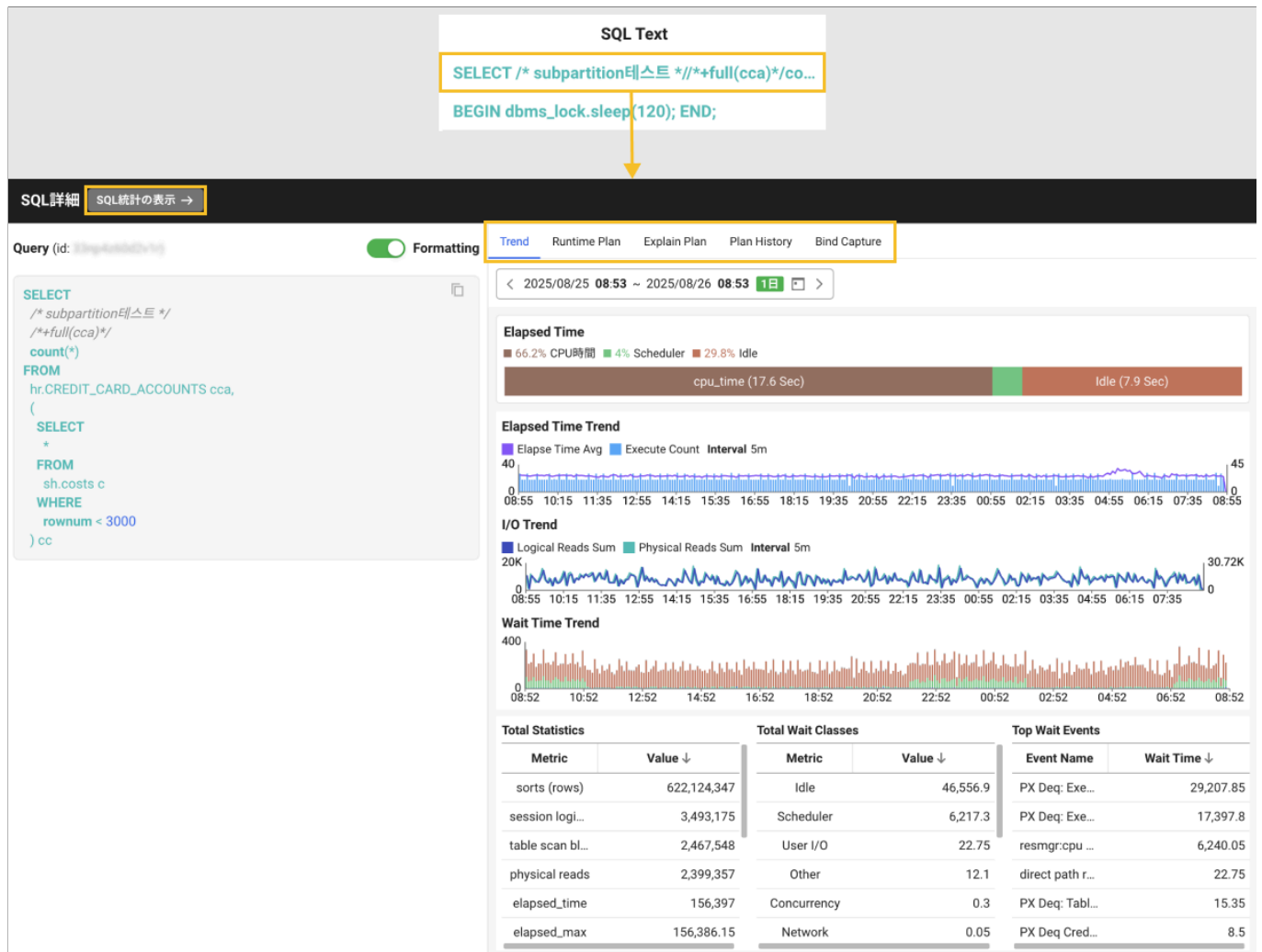


クエリの詳細を見る

検索結果リストで query カラム項目を選択すると **SQL 詳細** ウィンドウが表示されます。

SQL統計の表示 → ボタンをクリックすると、該当 SQL クエリに関連する統計情報を確認できる **SQL統計** メニューに

移動します。



SQL 詳細 ウィンドウのタブは以下の通りです。

- **Trend:** 特定 SQL の実行時間、I/O、待機イベントを総合的に分析し、パフォーマンスのボトルネックを迅速に確認できます。
- **Runtime Plan:** 選択した SQL クエリの実行計画とランタイム情報を表示します。実行回数、平均実行時間、平均物理読み取りなど詳細情報を提供します。
- **Explain Plan:** オプティマイザが予測した実行計画を表示します。コスト、操作、オブジェクト名、カーディナリティなどの情報を提供します。
- **Plan History:** データベースで実行された SQL クエリの実行計画履歴を確認できます。

- **Bind Capture:** データベースで実行された SQL クエリに使用されたバインド変数の値を確認できます。これによりクエリ実行の実際の内容を把握できます。

❗ Planツリーで🗨️ボタンを選択すると、チューニングガイドを確認できます。チューニングガイドは、Plan Treeノードが TABLE ACCESS (FULL) / INDEX (SKIP SCAN) である場合とノードに CARTESIAN キーワードが含まれている場合、確認できます。

テーブルデータのフィルタリング



1. テーブルの右上にある▽ボタンを選択します。
2. テーブルのカラムヘッダー項目と条件を選択します。
3. 条件を入力してください。テキストボックスに条件を入力します。
4. 保存ボタンを選択します。

テーブルカラムの設定

テーブルヘッダーカラムを非表示にしたり、希望する項目を追加できます。カラムの順序を変更することもできます。☐ボタンをクリックします。

- ❗ 設定を完了した後は、**確認**ボタンを選択すると、設定事項がテーブルに反映されます。
- ③ 検索バーにテキストを入力し、カラム項目を検索できます。入力したテキストと一致するカラム項目が表示されます。



- 画像は、製品またはプロジェクト、メニューによって異なる場合があります。

カラムを追加する

① 一覧でテーブルヘッダーカラムに追加する項目を選択します。すべての項目を追加するには、**すべて選択**を選択します。

カラムを削除する

① 一覧から削除するカラム項目のチェックボックスのチェックを解除します。または② 一覧から削除する項目の右側の×ボタンを選択します。

カラムの順序の変更

② 一覧で順序を変更するアイテムをドラッグして、希望する位置に移動できます。

設定事項の初期化

変更事項をすべてキャンセルし、初期化するには、 **初期化**ボタンを選択します。

カラム情報の案内

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

カラム情報に関する詳しい内容は、[次のリンク](#)を参照してください。

項目	説明
username	DBユーザー名
command	COMMAND_TYPE と同一、SQLのcommand number
status	セッションの状態
schemaname	スキーマ、usernameと同一

項目	説明
osuser	セッションに接続したOSのユーザー名
spid	OSプロセスの識別子
process	セッションに接続したクライアントプロセスID
logon_time	セッションが作成された時点
last_call_et	セッションがアクティブになっている場合、アクティブになってから経過した時間(秒)
sql_text	query、実行されているstatement、リテラル文字列を除外したSQL文
sql_param	query から分離したリテラル値
prev_sql_text	以前に実行したquery
prev_sql_param	以前に実行したquery param
program	クライアントで実行したプログラムの情報
type	session type - user background recursive (whatap.conf)で設定可能
seq#	現在または最後の待機を固有に識別する番号(待機するたびに増加)
event	セッションがキューに登録されている場合は、セッションが待機しているリソースまたはイベントの数
p1	最初の待機イベントのパラメータ(小数点)
p2	2番目の待機イベントのパラメータ(小数点)
p3	3番目の待機イベントのパラメータ(小数点)
wait_class	待機中のイベントのクラス名

項目	説明
wait_time	セッションが現在待機中の場合、値は0です。0より大きい場合は、最後の待機時間(milliseconds)
seconds_in_wait	セッションが現在待機中の場合、セッションが待機のために待機した時間を意味します。セッションが待機中でない場合、最後の待機開始からの時間を意味します。
state	待機状態、WAITING, WAITED UNKNOWN TIME, WAITED SHORT TIME, WAITED KNOWN TIME
machine	セッションに接続したクライアントサーバー名(hostname)
port	クライアントポート番号
terminal	端末名
module	クライアントで実行したプログラムでユーザが設定したモジュール(module)名
アクション	現在実行されているモジュールのユーザが設定したアクション名
client_info	ユーザが設定したクライアント情報
client_identifier	ユーザが設定したクライアントの識別子
session logical reads	バッファキャッシュ内でデータを読み込んだ回数
physical reads	物理ディスクからデータを読み出した回数
excute count	SQLの実行数
parse count (hard)	構文分析の呼び出し合計数(実際の構文分析)
parse count (total)	構文分析呼び出しの総数(hard、soft、describe)
opened cursors current	現在開いているcursorの数
db block changes	データベース内のすべてのブロックの変更回数

項目	説明
session pga memory	セッションの現在のPGAサイズ(size)
undo_segid	undo情報id
undo_blk	undoブロックの数
undo_rec	undoレコードの数
sql_address	SQL文(statement)を識別する情報
sql_hash_value	SQL文(statement)を識別する情報
sql_id	現在実行中のSQL文(statement)のSQL識別子
sql_child_number	現在実行中のSQL文(statement)の下位番号
sql_exec_start	当該セッションで現在実行中のSQL実行は、開始点にあります。
sql_exec_id	SQL実行識別子
prev_sql_addr	最後に実行されたSQL文(statement)を識別する情報
prev_child_number	最後に実行されたSQL文(statement)の下位番号
prev_exec_start	最後に実行されたSQL文(statement)のSQL実行の開始時点
prev_exec_id	最後に実行されたSQL文(statement)のSQL実行識別id
plsql_entry_object_id	スタックの一番上にあるPL/SQL下位プログラムのオブジェクトid
plsql_entry_subprogram_id	スタックで一番上にあるPL/SQL下位プログラムのサブプログラムid
plsql_object_id	現在実行中のPL/SQLオブジェクトのサブプログラムid
taddr	トランザクションのアドレス(address)

項目	説明
lockwait	lock wait address、セッションが待機中のロックアドレス
row_wait_obj	ROW_WAIT_ROW# で指定された行を含むテーブルのオブジェクトID
row_wait_file	ROW_WAIT_ROW# で指定された行を含むデータファイルの識別子
row_wait_block	ROW_WAIT_ROW# で指定された行を含むブロックの識別子
row_wait_row	現在ロックされている行
pdml_status	値が ENABLED の場合、セッションが PARALLEL DML 有効モードです。値が DISABLE の場合、セッションで PARALLEL DML 有効モードに対応していません。値が FORCED の場合、セッションが PARALLEL DML を強制的に使用するように変更されたものです。
pq_status	値が ENABLED の場合、セッションが PARALLEL QUERY 有効モードです。値が DISABLE の場合、セッションで PARALLEL QUERY 有効モードに対応していません。値が FORCED の場合、セッションが PARALLEL QUERY を強制的に使用するように変更されたものです。
blocking_session_status	ブロックしているセッションの存在有無に関する詳細を提供
blocking_instance	ブロックしているセッションのインスタンス識別子
blocking_session	ブロックしているセッションのセッション識別子
final_blocking_session_status	最終ブロックセッションの存在有無に関する詳細を提供
final_blocking_instance	最終ブロックしているセッションのインスタンス識別子
final_blocking_session	最終ブロックしているセッションのセッション識別子
service_name	セッションのサービス名やDB名と類似
saddr	セッションアドレス

項目	説明
con_id	データに関連するコンテナのid(マルチデータベースをサポートする場合のみ)

ⓘ WhaTapは、クライアントに関する情報をデフォルトで保存します。

項目	説明
Instance	インスタンス（エージェント）名
sid	セッションID
serial#	セッションの一意な識別子と共に使用するセッションシリアル番号
holder_type	ロックを保持しているセッションのロックタイプ（例：TX、TMなど）
holder_mode	保持中のロックモード（例：0～6、6はexclusive）
waiter_type	ロックを待機しているセッションのロックタイプ
waiter_mode	現在要求中のロックモード
id1	ロック識別子1（ロック対象オブジェクトを識別するために使用、例：トランザクションIDなど）
id2	ロックの詳細情報またはバージョン情報を表します

次の項目は、XOSエージェントをインストールした場合に収集される指標です。

項目	説明	単位
pid	プロセスID	-
cpuusage	プロセスが使用するCPU使用率	%
cputime	プロセスがこれまでに使用した累積CPU時間	秒

項目	説明	単位
elapsed	プロセスが実行されてからの経過時間	秒
vsize	Virtual memory size. 全体の仮想メモリ使用量（RAM + Swap + 予約）	KB
rss	Resident Set Size. 現在のプロセスが占有している実際の物理メモリサイズ	KB
pss	Proportional Set Size. 共有メモリを含む実際の物理メモリ使用量	KB
state	プロセスの現在の状態コード（例：R、S、D、T、Z など）	-
ioread	累積読み込みバイト数（read_bytes）	バイト（B）
iowrite	累積書き込みバイト数（write_bytes）	バイト（B）
uid	このプロセスを実行したユーザーID	-
cmd	実行されたコマンドの短縮形式	-
longcmd	実行されたコマンドの完全な形式（フルパスおよび引数を含む）	-

照会リストをダウンロードする

現在照会中のSession TableリストをCSVファイルでダウンロードできます。Session Tableセクションの右上の↓ボタンを選択します。

ダウンロードしたファイル名の形式は次のとおりです。

- アクティブセッション: activeSession_HH_MM_SS.csv
- ロックツリー: lockTree_HH_MM_SS.csv

別ウィンドウで見る

Session Tableリストを別ウィンドウで開き、より広い画面で確認できます。Session Tableセクションの右上の🖥️ボタンを選択します。

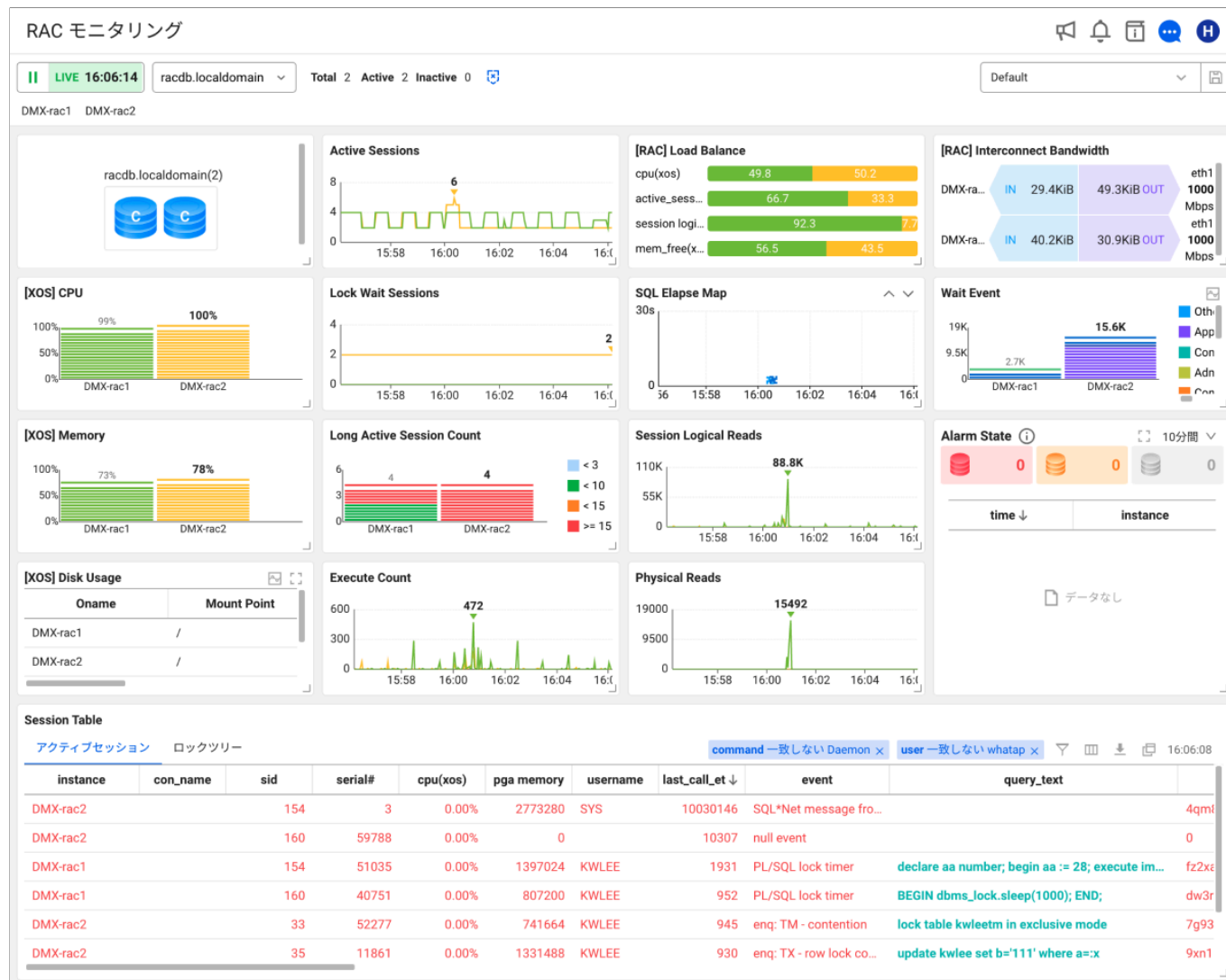
RACモニタリング

ホーム画面>プロジェクト選択> **ダッシュボード** > **RACモニタリング**

RACモニタリングは、Oracleデータベースでクラスターからなるデータベースを一つの単位にまとめてモニタリングできるように構成されています。 **RACモニタリング**メニューで提供されるDefaultプリセットは、RACモニタリングに最適化されています。該当プリセットを保存、削除、変更するために修正権限が必要です。

各インスタンスで発生する主な性能指標を、一目で比較して分析できます。実行中の各エージェントの数、すべてのアクティブセッションの合計、トランザクション数などのデータを集計して、効果的にモニタリングできます。ユーザーは、グラフチャートを通じて時間によるメトリクス値の変化と各インスタンスの高い性能メトリクスを簡単に確認できます。

また、最も長く続くアクティブセッションに関する情報を確認でき、ユーザーはシステム性能のボトルネック現象を診断し、解決できる根拠資料として活用できます。



✓ インスタンスモニタリングとの違い

RACモニタリング画面は既存のマルチインスタンスモニタリングでRACプリセットとクラスターを選択できるインスタンスセクターとして提供されましたが、ダッシュボードに分離されました。

- 使用できるウィジェットに違いがあります。
 - [RAC] Load Balance
 - [RAC] Interconnect Bandwidth
 - [RAC] Main Waits
- DefaultプリセットがRACモニタリングに最適化されました。



- ・ インスタンスSelectではなくクラスターSelectに変更されました。

照会時間設定

ダッシュボードではリアルタイムモニタリング機能を基本的に提供しますが、画面上部のタイムセレクターを利用して過去の時間のデータを照会できます。タイムセレクターで || ボタンを選択して時間を設定します。ユーザーが設定した時間に基づいて、ダッシュボードに配置したウィジェットのデータを更新します。

エージェント確認

エージェント接続状態を確認する

|| LIVE 10:08:02

エージェント ▼

Total 6 Active 6 Inactive 0

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

画面の左上のタイムピッカーの右側には、プロジェクトに関連したエージェントの状態を確認できる情報が表示されます。この機能で、モニタリング対象サーバーの動作有無がすぐに確認できます。

- ・ **Total**: プロジェクトに関連付けられているすべてのエージェントの数
- ・ **Active**: 活性化されたエージェントの数
- ・ **Inactive**: 非活性化状態のエージェント数
- ・ : 非活性化されたエージェントを表示または非表示します。

エージェント別モニタリング

|| LIVE 10:08:02

エージェント ▼

Total 6 Active 6 Inactive 0

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

基本的に、ダッシュボードにはすべてのエージェントから収集した指標をチャートとして表示しますが、エージェント

を選択してデータを表示することもできます。タイムピッカーの下にあるエージェントを1つ以上選択します。選択したエージェントの指標でダッシュボードに配置されたウィジェットのデータを更新します。

- ✓ エージェントを1つ以上選択した状態で再度すべてのエージェントを選択するには、選択を解除するか、[Total](#)をクリックします。

- ! プロジェクトに接続されたエージェントの数が多い場合は、エージェント名を短く設定することをお勧めします。エージェント名の設定に関する詳細は、[次の文書](#)を参照してください。

ウィジェット編集

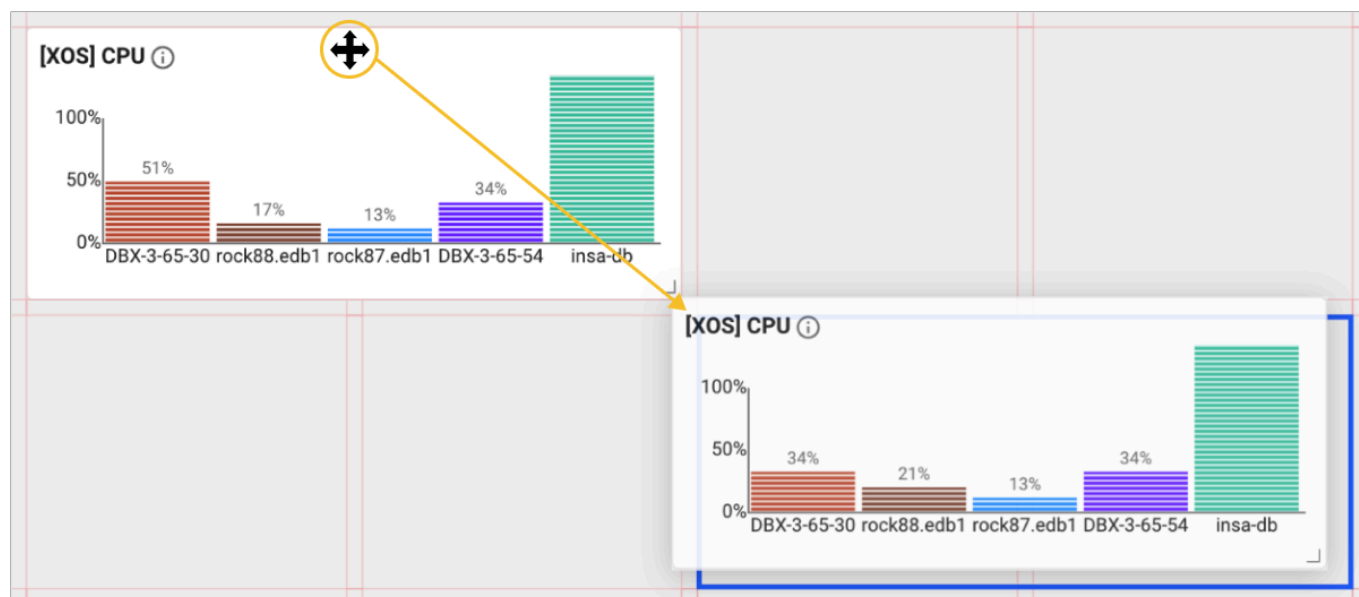
ダッシュボードに配置したウィジェットは、ユーザーが希望するサイズに調整でき、任意の位置に配置できます。ウィジェットは削除したり、再度追加したりすることもできます。

ウィジェットのサイズ変更



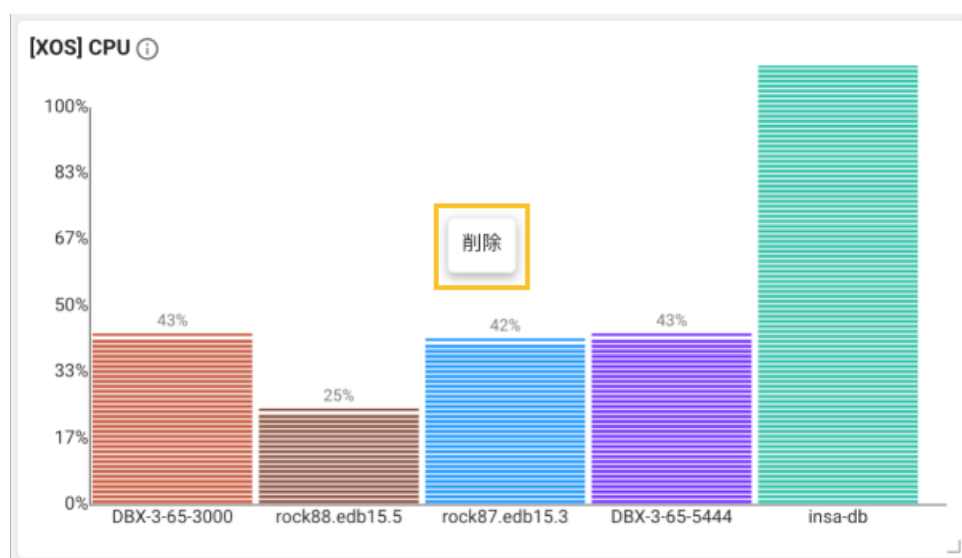
ウィジェットの右下にある「L」をマウスでクリックした状態で希望するサイズにドラッグします。幅と高さの比率が均一なグリッドが表示され、グリッド単位でウィジェットのサイズを調整できます。

ウィジェットの移動



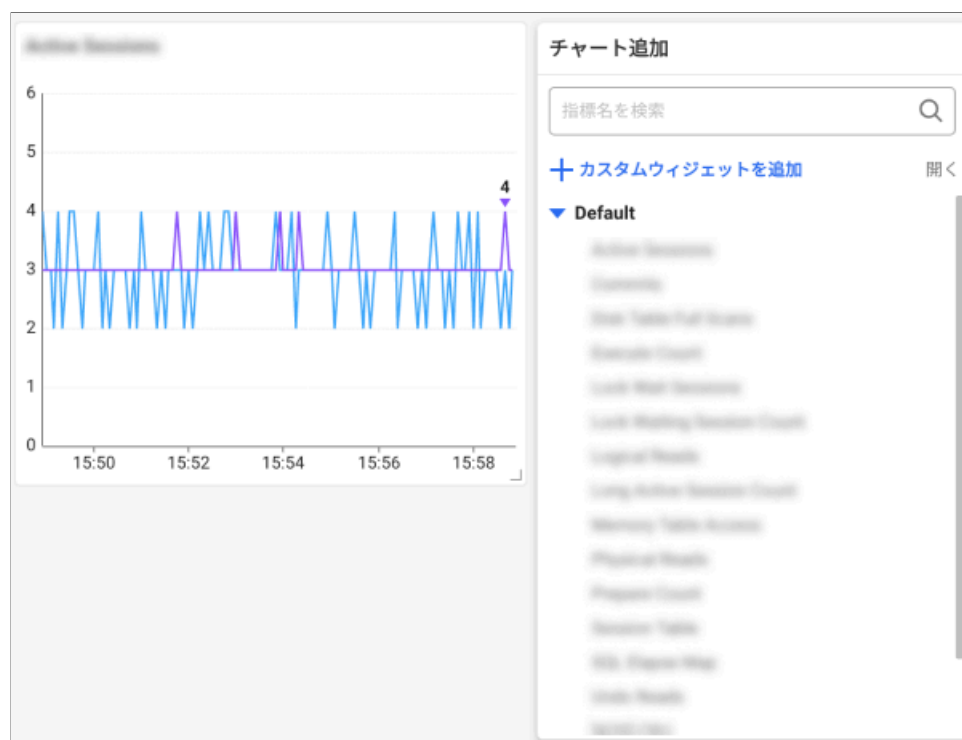
ウィジェットの上部にマウスカーソルを位置すると、カーソルの形が $\oplus$ の形に変更されます。この時、マウスの左ボタンをクリックした状態で希望する位置にドラッグすると、ウィジェットを移動することができます。

ウィジェットを削除する



削除するウィジェットで右クリックします。削除ボタンをクリックすると、ウィジェットがダッシュボードから削除されます。

ウィジェットを追加する



ダッシュボードから空白エリアにマウスカーソルを移動し、マウスの右ボタンをクリックします。ポップアップメニューから追加するウィジェットを選択します。希望する場所にウィジェットを配置してサイズを変更します。

- ❗ • ダッシュボードに配置できるウィジェットの詳細については、[次の文書](#)を参考にしてください。
- ダッシュボードで基本的に提供されるメトリクスの他に、ユーザーが希望するメトリクスを追加できます。ユーザー定義ウィジェットの詳細については、[次の文書](#)を参照してください。

カスタムウィジェット

カスタムウィジェットは、ダッシュボードでデフォルトに提供されるメトリクスの他に、ユーザーがメトリクスを追加

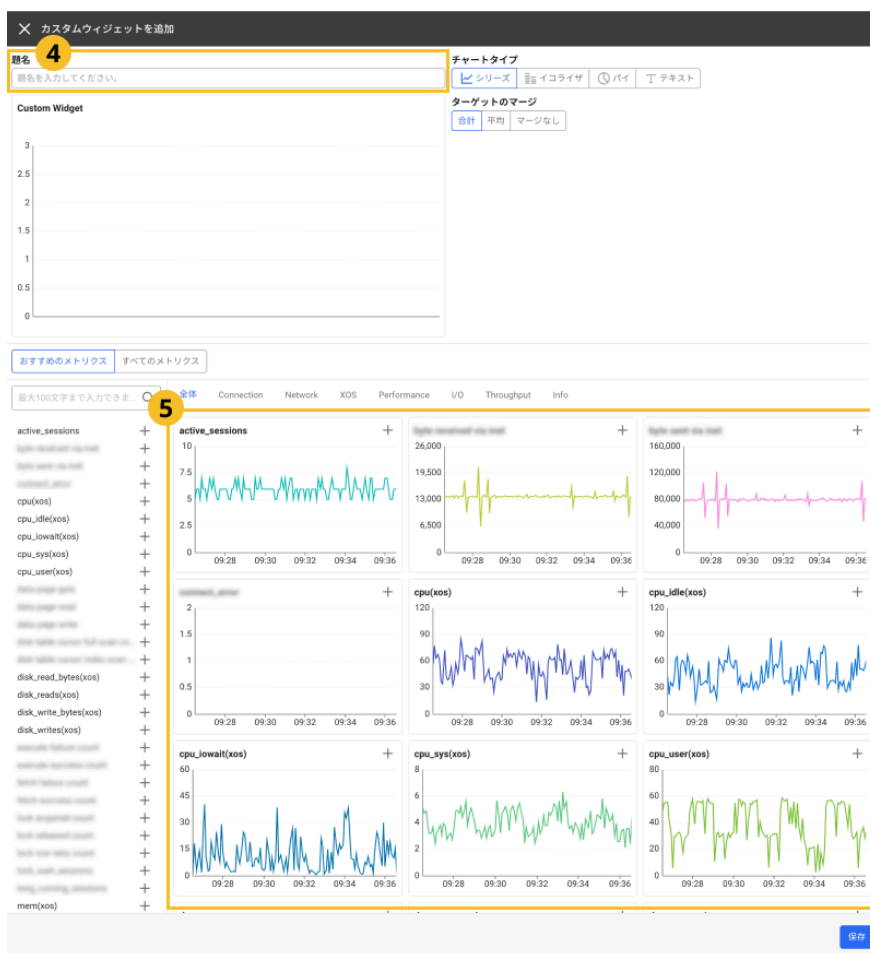
できる機能です。デフォルトで提供するメトリクスの他に、ダッシュボードで継続的にモニタリングしたいメトリクスがある場合は、カスタムウィジェット機能を利用できます。

追加する

1. ダッシュボードの空のスペースで、右クリックします。
2. ポップアップメニューで**カスタムウィジェットを追加**を選択します。
3. ダッシュボードに**Custom Widget**が生成されたら、**指標の選択**ボタンを選択します。



4. 画面の右側から **カスタムウィジェットを追加**画面が現れると、**題名項目**にウィジェットの名前を入力します。



5. 画面下のメトリクス一覧から、追加するメトリクスの右上にある＋ボタンを選択します。

6. メトリクスをすべて追加したら**保存**ボタンを選択します。

- ① 最大4つのメトリクスを追加できます。メトリクスを追加する時は、＋ボタンを選択します。
- メトリクスを変更する時は、メトリクス一覧で別のメトリクス項目を選択します。＋ボタンを選択すると、そのメトリクスが追加されます。
- ウィジェットのタイトルを入力しないと、Custom Widgetと保存されます。

チャートタイプを選択する

カスタムウィジェットを追加画面のチャートタイプでは、次の形式のチャートを選択できます。

-  **シリーズ**: 時間の経過によるメトリクスの変化を確認できる時系列形式のチャートです。
-  **イコライザ**: 複数のメトリクスの性能を比較できる棒グラフ形式のチャートです。様々なメトリクスを1つの画面に視覚的に表現することで、全体的な性能状態を簡単に把握できます。
-  **パイ**: 複数のメトリクスの全体に対する部門構成比率を確認するのに役立つ円形チャートです。
-  **テキスト**: メトリクスに対する数値をテキスト形式で表示します。

データマージ方式を選択する

カスタムウィジェットを追加画面の対象のマージでは、エージェントから収集されたメトリクスを表示する方法を選択できます。

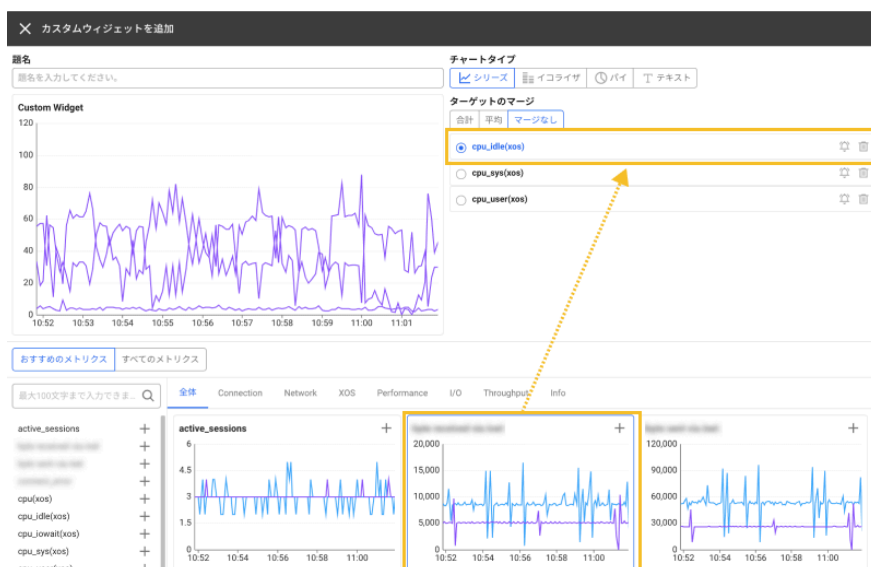
- **合計**: 複数のエージェントから収集されたメトリクス値を合計して、1つの値として表示します。このオプションは、複数のエージェントのデータを統合して全体的な合計を表示する場合に役立ちます。
- **平均**: 複数のエージェントから収集されたメトリクス値を平均して、1つの値として表示します。このオプションは、エージェント全体の平均性能や状態を把握するのに役立ちます。
- **最大値**: 複数のエージェントから収集された指標値の中から最大の値を選択して、一つの値として表示します。このオプションは、エージェントの中で最もパフォーマンスが消耗した状況や最悪の状態を把握するのに役立ちます。
- **マージなし**: 各エージェントのメトリクスを個別に表示します。

 チャートのタイプによって選択できるマージ方式は異なる場合があります。

指標の変更

ユーザー定義ウィジェットに追加したメトリクスを、他のメトリクスに変更または追加できます。

1. ダッシュボードに追加したユーザー定義ウィジェットの右上に ボタンを選択します。
2. 画面の右側で**カスタムウィジェットを追加画面**が表示されたら、変更するメトリクスを選択します。



3. 画面下のメトリクスの一覧で変更するメトリクスを選択します。メトリクスを追加する場合、追加するメトリクスの右上に+ ボタンを選択します。
4. メトリクスを全て変更したのであれば保存ボタンを選択します。

✓ 追加または変更したメトリクス一覧に表示されたアイコンボタンの機能は、次のとおりです。

- 🗑️ : 該当メトリクスをカスタムウィジェットから除外できます。
- 🔔 : 該当メトリクスに対するアラート通知を設定できます。ボタンを選択すると、**アラート通知 > イベント設定**メニューに移動します。メトリクスイベント設定の詳細については、[次の文書](#)を参考にしてください。

ウィジェットオプション

ウィジェットに表示されているアイコンボタンの機能は次のとおりです。

- ⓘ : 主要ウィジェットに関する機能と情報が確認できます。(多言語に対応予定)
- ⤴ / ⤵ : チャートの縦軸範囲を拡大または縮小できます。
- 📄 : 該当ウィジェットのデータを広い画面に開いて表示します。
- 🔍 : 該当ウィジェットのデータをエージェントごとに区分して照会できる詳細画面が表示されます。

❗ ウィジェットによって使用可能なオプションは異なる場合があります。

プリセットを設定する

ダッシュボードからユーザーが設定したウィジェットの設定とレイアウト状態を保存して読み込むことができます。ウィジェットのサイズを調整し、希望する場所に配置して新しいプリセットを作成できます。



- **Default:** RACモニタリング画面の主要DB指標とXOS指標で構成されたプリセットです。RACデータベースで実行中のインスタンスに基づいて指標を確認できます。

- ❗
- プリセットに保存できるダッシュボードの設定は次のとおりです。
 - ウィジェットのサイズ、位置
 - ユーザー定義ウィジェットに定義された設定
 - **Session Table**ウィジェットに定義されたテーブルの幅とフィルター、選択したカラム値
 - 基本プリセット(デフォルト)は変更できません。
 - データベース環境によって、一部のプリセットはサポートされない場合があります。
 - **プロジェクト修正権限**のあるメンバーのみがプリセットを保存または変更できます。メンバー権限の詳細については、[次の文書](#)を参照してください。

新しいプリセットを作成する

1. ダッシュボードから、希望する形にウィジェットを配置してみてください。サイズを変更して、頻繁に確認するウィジェットのみを配置することもできます。
2. 画面の右上にボタンを選択します。
3. 新しいプリセット名を入力します。

Default

プリセット保存

現在設定された指標とレイアウトを保存します。

Default

☐ エージェント選択履歴を含む

キャンセル 保存

エージェント選択履歴と一緒に保存するには、**エージェント選択履歴を含む**を選択します。

4. 保存ボタンを選択します。プリセット一覧から新しく保存したプリセットを確認できます。

- ① 新しく作成したプリセットに変更が加えられた場合は、再度プリセットを保存してください。📁ボタンを選択し、同じ名前でプリセットを保存します。既存のプリセットに変更を上書きします。
- ダッシュボードの変更を保存せずに別のメニューに移動すると、変更は保存されません。
- プリセットはプロジェクト単位で保存され、他のユーザーと共有できます。

プリセットを削除する

使用していないプリセットは、プリセット一覧から削除できます。プリセット一覧から削除したい項目の右側に **×** ボタンをクリックします。

- ① **Default** プリセットは削除できません。

主なウィジェットを調べる

ダッシュボードに配置できるウィジェットの種類は次のとおりです。

XOSウィジェット

- [XOS] CPU

エージェント別のCPU使用率をリアルタイムで確認できます。

- [XOS] Memory

エージェント別のメモリ使用率をリアルタイムで確認できます。

- [XOS] Disk Usage

エージェント別のディスク使用率と使用可能サイズ、全体サイズ情報を表示します。

- : **ディスク使用量の詳細**ポップアップが表示されます。時間帯別のディスク使用量を表示するチャートを確認できます。 インスタンスごとに確認でき、**Size**、**Free**、**Used**ボタンを選択すると、全体のサイズ、残りのサイズ、使用中のサイズが確認できます。
- : ウィジェットに表示されたテーブルをより広い画面で確認できます。

✔ データが表示されない場合は、`xos.conf`ファイルに次の設定を追加します。

```
xos.conf
```

```
# disk usage 0ならばoff、単位: 秒
disk_usage_interval=60
```

Oracle RACウィジェット

- [RAC] Load Balance

Oracle RAC(Real Application Clusters) データベースで実行されているインスタンスのリソースをリアルタイムでモニタリングできます。 ユーザーは、データベースの負荷及び性能指標を視覚的にモニタリングできます。 CPU性能とアクティブセッション、物理的な読み取り操作、使用可能なメモリの数値を確認できます。

- [RAC] Interconnect Bandwidth

Oracle RAC(Real Application Clusters)で使用されるインターコネクト(Interconnect)の帯域幅をモニタリングするウィジェットです。 データベースノード間の通信帯域幅をリアルタイムで表示することで、クラスター内のデータベース、インスタンス間のネットワークトラフィックの状態をモニタリングできます。

- [RAC] Main Waits

Global Lockステータスと各イベントの待機回数を把握できます。 このウィジェットは、データベースの特定の力

カテゴリから収集した指標を通じて、各イベントの総待機回数を照会し、各エージェント(インスタンス)の最新データを抽出します。これにより、特定の時間に発生したデータベースの待機イベントの状態をすばやく把握できます。

Databaseウィジェット

提供するウィジェットを通じて、データベース性能分析の主要メトリクスとして、CPU、メモリ、ディスクI/Oなど多様なシステムリソースの状態を確認できます。

- **Active Sessions**

時間に応じたアクティブセッションの数値を時系列チャートで視覚的に表示します。データベースで実行されているセッションの数を時間ごとに追跡し、グラフ化したウィジェットです。セッションの使用パターンと負荷を視覚的に把握するのに役立ちます。

- **Lock Wait Sessions**

時間に応じたロック待機セッションの数値を時系列グラフで視覚的に表示します。データベースでロック待機状態にあるセッションの数を時間ごとに追跡し、リアルタイムグラフで表示したウィジェットです。

- **Long Active Session Count**

データベースで実行される活性セッションの長期実行に関する情報を提供します。セッションの実行時間に応じてグループ化され、時間単位で集計された情報をチャートで表示します。長時間実行されるセッションを識別してデータベースの性能問題やリソース漏れを早期に感知したり、セッション実行時間による分類を通じてどのセッションが長期間にわたり活性状態に維持されているかを視覚的に把握できます。

3秒未満は青、3秒以上10秒未満は緑、10秒以上15秒未満はオレンジ、15秒以上は赤で表示します。

- **Long Waiting Session Count**

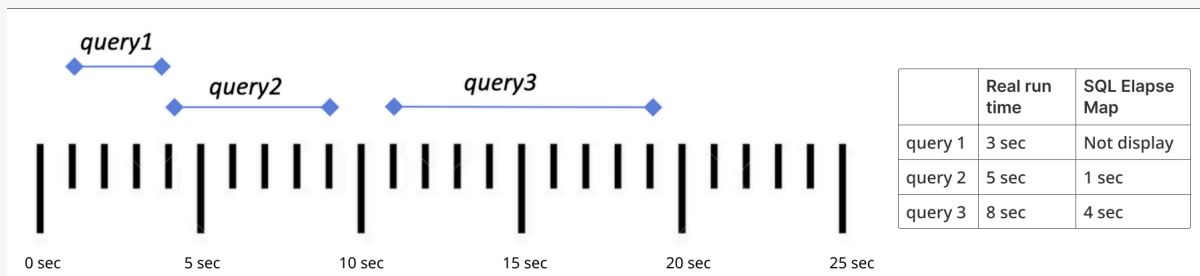
データベースでロックを保持または待機するセッションの長期待機時間に関する情報を提供します。ロック情報と活性セッション情報を組み合わせて、待機中のセッションの長期待機時間を追跡できます。長時間ロックを保持または待機するセッションをモニタリングします。どのセッションが長時間ロックを保持または待機しているかを視覚的に把握できます。

5秒未満は青、5秒以上10秒未満は緑、10秒以上60秒未満はオレンジ、60秒以上は赤で表示します。

- **SQL Elapse Map**

実行中のクエリの経過時間をリアルタイムチャートで表示します。チャートの特定の領域をドラッグするとQuery List一覧を確認できます。へまたはvボタンを選択して、縦軸の範囲を拡大または縮小できます。

- ① クエリの実行時間は、アクティブセッションを収集して計算します。その結果、アクティブセッションの収集サイクルによって誤差範囲が発生する可能性があります。次の図を参照してください。収集サイクルの間に終了したSQL実行時間は計算されません。収集サイクルが短いほど誤差範囲は小さくなりますが、DBの負荷を考慮して一般的に5秒に設定されています。長時間にわたって実行されるクエリをモニタリングする場合に役立ちます。SQL統計メニューのデータは同じです。



• Wait Event

各バックエンドプロセスが、現在待機中の待機イベント名とそのイベントの発生頻度をリアルタイムのグラフで表示します。主にApplication、Configuration、Administrative、Networkなどに関連する項目の待機イベントをモニタリングします。各待機イベントの発生頻度を視覚的に確認でき、グラフの上にマウスオーバーすると、ツールチップが表示され、各待機イベントに関する詳細情報を提供します。

• Physical Reads

データベースで発生する物理的な読み取り操作(physical reads)をモニタリングするウィジェットです。主にデータベースがディスクからデータを読み込む作業に関する統計を提供し、データベースのI/O性能をモニタリングします。

• Excute Count

データベースで実行されたクエリまたはプロシージャの実行回数をモニタリングするウィジェットです。データベースによって実行されたクエリやプロシージャの実行回数をリアルタイムで追跡し、ユーザーに提供します。

• Session Logical Reads

データベースセッションで実行された論理的な読み取り操作(logical reads)をモニタリングするウィジェットです。データベースセッションで実行された論理的な読み取り操作の回数をリアルタイムで追跡し、ユーザーに提供します。

Session Tableウィジェット

画面下段のSession Tableウィジェットでは、リアルタイムで実行されているアクティブセッションとロックツリー情報を照会できます。

- **アクティブセッション**：リアルタイムで実行中のセッションとクエリ内容を照会できます。
 - 実行中のセッション項目を選択すると**セッション詳細画面**が表示され、選択したセッションの詳細を確認できます。[次の文書](#)を参考にしてください。
 - テーブルの右側にある**query_text**カラムの項目を選択すると、**SQL詳細画面**が表示されます。選択したクエリ(Query)に対するFull TextおよびPlan情報を確認できます。詳細については、[次の文書](#)を参照してください。

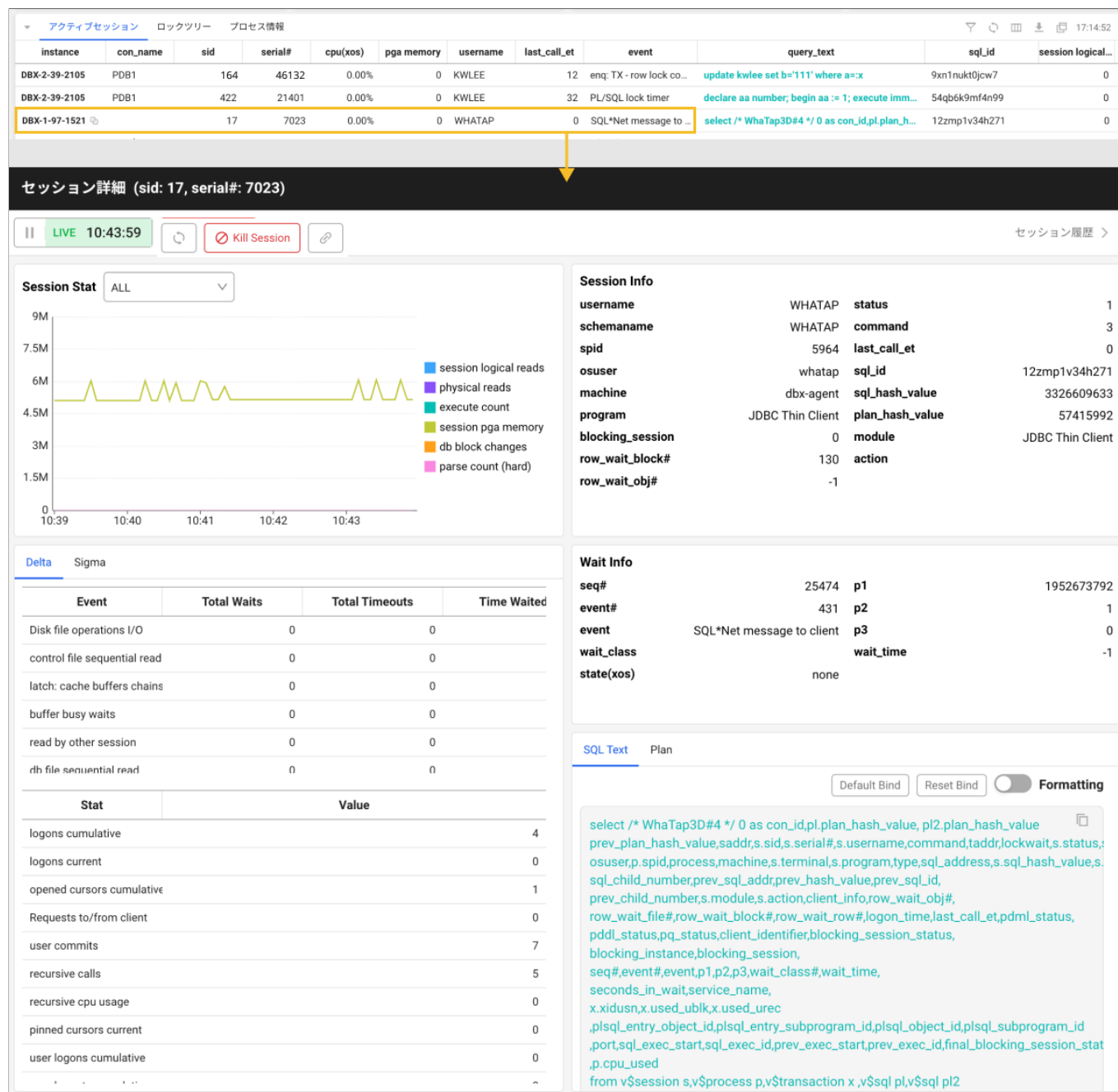
- **ロックツリー**

ロックセッションに関する情報やHolderとWaiter情報をリアルタイムで照会できます。テーブルのカラムの詳細については、[次の文書](#)を参照してください。

① テーブル一覧の文字の色は黒 → オレンジ → 赤順にセッションの実行速度が遅いことを意味します。

セッション詳細情報を確認する

アクティブセッションタブでセッション項目を選択すると**セッション詳細画面**が表示されます。選択したセッションについての詳細を確認できます。



- セッションに関する情報は、既定値でリアルタイムで照会されています。タイムセレクターの || ボタンを選択すると、リアルタイム照会を中止し、中止した時間ベースのデータを確認できます。リアルタイム照会を中止した状態でデータを更新するには、🔄 ボタンを選択します。
- Kill Session** : 選択したセッションを停止できます。セッション中止画面が表示されたらパスワード(Parameter

Key)を入力し、適用ボタンを選択します。

❗ この機能は、リアルタイム照会モードでのみ利用できます。

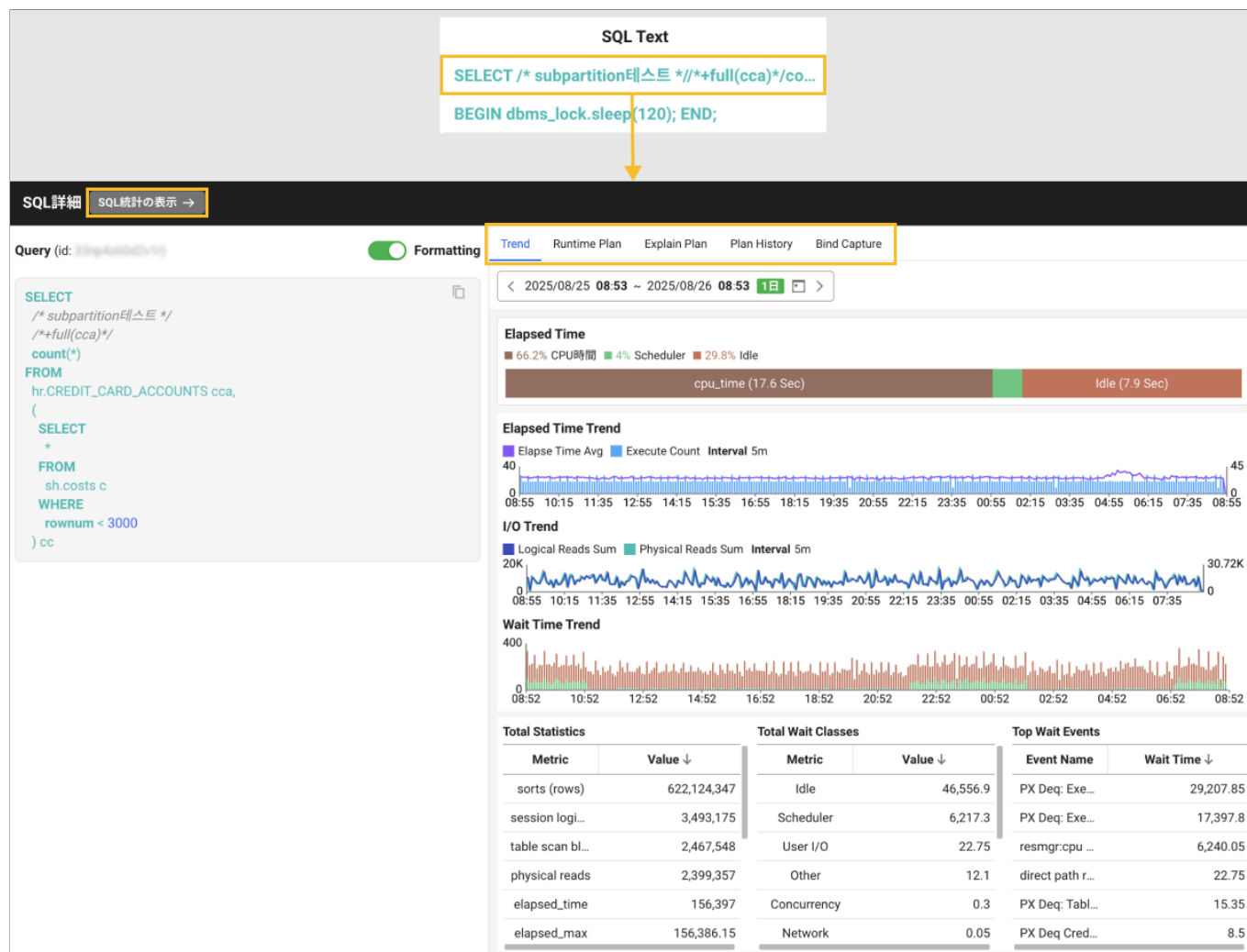
- **セッション履歴**：選択したセッション履歴が確認できる**セッション履歴**メニューに移動できます。

❗ セッション履歴メニューの詳細については、[次の文書](#)を参照してください。

- **Session Stat**: 選択したセッションの様々な性能メトリクスと統計情報を提供します。セッションで実行された作業のタイプと回数、リソース使用量などが含まれます。
- **Delta / Sigma**: 選択したセッションの変化率と統計的分析情報を提供します。セッションのアクティビティに対する変化量を確認できます。
- **Session Info**: 選択したセッションの基本情報とステータスを含む詳細を確認できます。
- **Wait Info**: 選択したセッションの待機イベントと待ち時間の情報を提供します。特定の作業を実行するために、データベースシステムで待機する時に発生する待機イベントに関する情報も確認できます。
- **SQL Text / Plan**：選択したセッションのSQLクエリ文とPlan情報を確認できます。詳細については[次の文書](#)を参照してください。

SQL 詳細情報の確認

SQL統計の表示 → ボタンをクリックすると、選択した SQL クエリに関連する統計情報を確認できる **SQL統計** に移動します。



- **Runtime Plan:** 選択されたSQLクエリの実行計画とランタイム情報を提供します。 実行回数、平均実行時間、平均物理的読み取りなどの詳細を提供します。
- **Explain Plan:** オプティマイザーが予測した実行計画を示します。 費用、作業、オブジェクト名、カーディナリティなどの情報を提供します。
- **Plan History:** データベースで実行されたSQLクエリの実行計画に対する履歴を確認できます。
- **Bind Capture:** データベースで実行されたSQLクエリに使用されたバインド変数の値を確認できます。 これにより、クエリ実行の実際の内容を確認できます。

① リアルタイムで実行されたbind値ではなく、データベースにキャプチャされた値(v\$sql_bind_capture)

① です。件数が多いと、100個まで表示します。

AI チューニングガイド

AI チューニングガイドは、SQL クエリ、プラン、統計情報を分析してパフォーマンス問題を診断し、最適化の提案を行う機能です。

開発者や DBA がボトルネックの原因を迅速に特定し、効率的な SQL によりパフォーマンスを改善できるよう支援します。

⚠ 使用条件および注意事項

PostgreSQL、MySQL、SQL Server ではプランの取得が必須です。

プランを取得しない場合、AI チューニングガイドアイコンは （無効）として表示され、機能を使用できません。

- AI が生成する結果は自動分析に基づいており、精度が 100% ではないことにご留意ください。

- 分析および診断する SQL をクリックし、**SQL詳細** 画面に移動します。
- SQL詳細** 画面の右下にある AI チューニングガイドアイコン  をクリックして AI 分析を開始します。
- AI 分析結果を確認します。

結果項目	説明
クエリプランおよび概要	クエリの目的と実行概要を表示します。 実行回数、累積実行時間、データベース全体の負荷比率を分析し、当該 SQL がシステムパフォーマンスに与える影響を評価します。
パフォーマンス分析	分析結果を総合し、パフォーマンススコアと診断結果を提供します。 CPU 使用率、ディスク使用率、キャッシュヒット率、待機時間など、クエリ実行過程における詳細なリソース使用量を分析し、ボトルネックの発生区間を視覚的に表示します。
発見された主な問題	主な問題点を要約して表示します。

結果項目	説明
最適化推奨事項	検出された問題に基づき、最適化されたクエリを提案します。

デフォルトプランを照会する

デフォルトプランユーザーファイル作成時、エージェントが事前定義されたDB情報を使用してプランを自動的に照会するように設定できます。デフォルトプランの照会項目のチェックボックスを選択した後、次のように予め入力したDB接続情報を通じてプラン情報を照会します。

• Linux Windows

1. エージェント設定(whatap.conf)ファイルに plan_db オプションを追加し、DB名を入力します。

whatap.conf

```
plan_db=(dbname)
```

2. DB設定後 planuid.sh スクリプトを使用してユーザーとパスワードを設定します。

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

❗ ユーザー(user)とパスワード(pwd)に特殊文字が含まれている場合、各特殊文字の前にバックslash(\)を追加します。

例、 user\$example → user\example

1. エージェント設定(whatap.conf)ファイルに plan_db オプションを追加し、DB名を入力します。

whatap.conf

```
plan_db=(dbname)
```

2. DB設定後 planuid.bat スクリプトを使用してユーザーとパスワードを設定します。

planuid.bat user pwd

テーブルデータのフィルタリング



1. テーブルの右上にある🔍ボタンを選択します。
2. テーブルのカラムヘッダー項目と条件を選択します。
3. 条件を入力してください。テキストボックスに条件を入力します。
4. 保存ボタンを選択します。

テーブルカラムの設定

テーブルヘッダーカラムを非表示にしたり、希望する項目を追加できます。カラムの順序を変更することもできます。
🔍ボタンをクリックします。

- ① • 設定を完了した後は、**確認**ボタンを選択すると、設定事項がテーブルに反映されます。
- ③ • 検索バーにテキストを入力し、カラム項目を検索できます。入力したテキストと一致するカラム項目が表示されます。
- 画像は、製品またはプロジェクト、メニューによって異なる場合があります。

カラムを追加する

①一覧でテーブルヘッダーカラムに追加する項目を選択します。すべての項目を追加するには、**すべて選択**を選択します。

カラムを削除する

①一覧から削除するカラム項目のチェックボックスのチェックを解除します。または②一覧から削除する項目の右側の×ボタンを選択します。

カラムの順序の変更

②一覧で順序を変更するアイテムをドラッグして、希望する位置に移動できます。

設定事項の初期化

変更事項をすべてキャンセルし、初期化するには、🔄 **初期化**ボタンを選択します。

カラム情報の案内

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

カラム情報に関する詳しい内容は、[次のリンク](#)を参照してください。

項目	説明
username	DBユーザー名
command	COMMAND_TYPE と同一、SQLのcommand number
status	セッションの状態
schemaname	スキーマ、usernameと同一
osuser	セッションに接続したOSのユーザー名

項目	説明
spid	OSプロセスの識別子
process	セッションに接続したクライアントプロセスID
logon_time	セッションが作成された時点
last_call_et	セッションがアクティブになっている場合、アクティブになってから経過した時間(秒)
sql_text	query、実行されているstatement、リテラル文字列を除外したSQL文
sql_param	query から分離したリテラル値
prev_sql_text	以前に実行したquery
prev_sql_param	以前に実行したquery param
program	クライアントで実行したプログラムの情報
type	session type - user background recursive (whatap.conf)で設定可能
seq#	現在または最後の待機を固有に識別する番号(待機するたびに増加)
event	セッションがキューに登録されている場合は、セッションが待機しているリソースまたはイベントの数
p1	最初の待機イベントのパラメータ(小数点)
p2	2番目の待機イベントのパラメータ(小数点)
p3	3番目の待機イベントのパラメータ(小数点)
wait_class	待機中のイベントのクラス名
wait_time	セッションが現在待機中の場合、値は0です。0より大きい場合は、最後の待機時間(milliseconds)

項目	説明
seconds_in_wait	セッションが現在待機中の場合、セッションが待機のために待機した時間を意味します。セッションが待機中でない場合、最後の待機開始からの時間を意味します。
state	待機状態、WAITING、WAITED UNKNOWN TIME、WAITED SHORT TIME、WAITED KNOWN TIME
machine	セッションに接続したクライアントサーバー名(hostname)
port	クライアントポート番号
terminal	端末名
module	クライアントで実行したプログラムでユーザが設定したモジュール(module)名
アクション	現在実行されているモジュールのユーザが設定したアクション名
client_info	ユーザが設定したクライアント情報
client_identifier	ユーザが設定したクライアントの識別子
session logical reads	バッファキャッシュ内でデータを読み込んだ回数
physical reads	物理ディスクからデータを読み出した回数
excute count	SQLの実行数
parse count (hard)	構文分析の呼び出し合計数(実際の構文分析)
parse count (total)	構文分析呼び出しの総数(hard、soft、describe)
opened cursors current	現在開いているcursorの数
db block changes	データベース内のすべてのブロックの変更回数
session pga memory	セッションの現在のPGAサイズ(size)

項目	説明
undo_segid	undo情報id
undo_blk	undoブロックの数
undo_rec	undoレコードの数
sql_address	SQL文(statement)を識別する情報
sql_hash_value	SQL文(statement)を識別する情報
sql_id	現在実行中のSQL文(statement)のSQL識別子
sql_child_number	現在実行中のSQL文(statement)の下位番号
sql_exec_start	当該セッションで現在実行中のSQL実行は、開始点にあります。
sql_exec_id	SQL実行識別子
prev_sql_addr	最後に実行されたSQL文(statement)を識別する情報
prev_child_number	最後に実行されたSQL文(statement)の下位番号
prev_exec_start	最後に実行されたSQL文(statement)のSQL実行の開始時点
prev_exec_id	最後に実行されたSQL文(statement)のSQL実行識別id
plsql_entry_object_id	スタックの一番上にあるPL/SQL下位プログラムのオブジェクトid
plsql_entry_subprogram_id	スタックで一番上にあるPL/SQL下位プログラムのサブプログラムid
plsql_object_id	現在実行中のPL/SQLオブジェクトのサブプログラムid
taddr	トランザクションのアドレス(address)
lockwait	lock wait address、セッションが待機中のロックアドレス

項目	説明
row_wait_obj	ROW_WAIT_ROW# で指定された行を含むテーブルのオブジェクトID
row_wait_file	ROW_WAIT_ROW# で指定された行を含むデータファイルの識別子
row_wait_block	ROW_WAIT_ROW# で指定された行を含むブロックの識別子
row_wait_row	現在ロックされている行
pdml_status	値が ENABLED の場合、セッションが PARALLEL DML 有効モードです。値が DISABLE の場合、セッションで PARALLEL DML 有効モードに対応していません。値が FORCED の場合、セッションが PARALLEL DML を強制的に使用するように変更されたものです。
pq_status	値が ENABLED の場合、セッションが PARALLEL QUERY 有効モードです。値が DISABLE の場合、セッションで PARALLEL QUERY 有効モードに対応していません。値が FORCED の場合、セッションが PARALLEL QUERY を強制的に使用するように変更されたものです。
blocking_session_status	ブロックしているセッションの存在有無に関する詳細を提供
blocking_instance	ブロックしているセッションのインスタンス識別子
blocking_session	ブロックしているセッションのセッション識別子
final_blocking_session_status	最終ブロックセッションの存在有無に関する詳細を提供
final_blocking_instance	最終ブロックしているセッションのインスタンス識別子
final_blocking_session	最終ブロックしているセッションのセッション識別子
service_name	セッションのサービス名やDB名と類似
saddr	セッションアドレス
con_id	データに関連するコンテナのid(マルチデータベースをサポートする場合のみ)

ⓘ WhaTapは、クライアントに関する情報をデフォルトで保存します。

項目	説明
Instance	インスタンス（エージェント）名
sid	セッションID
serial#	セッションの一意的識別子と共に使用するセッションシリアル番号
holder_type	ロックを保持しているセッションのロックタイプ（例：TX、TMなど）
holder_mode	保持中のロックモード（例：0～6、6はexclusive）
waiter_type	ロックを待機しているセッションのロックタイプ
waiter_mode	現在要求中のロックモード
id1	ロック識別子1（ロック対象オブジェクトを識別するために使用、例：トランザクションIDなど）
id2	ロックの詳細情報またはバージョン情報を表します

次の項目は、XOSエージェントをインストールした場合に収集される指標です。

項目	説明	単位
pid	プロセスID	-
cpuusage	プロセスが使用するCPU使用率	%
cputime	プロセスがこれまでに使用した累積CPU時間	秒
elapse	プロセスが実行されてからの経過時間	秒
vsize	Virtual memory size. 全体の仮想メモリ使用量（RAM + Swap + 予約）	KB
rss	Resident Set Size. 現在のプロセスが占有している実際の物理メモリサイズ	KB

項目	説明	単位
pss	Proportional Set Size. 共有メモリを含む実際の物理メモリ使用量	KB
state	プロセスの現在の状態コード（例：R、S、D、T、Z など）	-
ioread	累積読み込みバイト数（read_bytes）	バイト（B）
iowrite	累積書き込みバイト数（write_bytes）	バイト（B）
uid	このプロセスを実行したユーザーID	-
cmd	実行されたコマンドの短縮形式	-
longcmd	実行されたコマンドの完全な形式（フルパスおよび引数を含む）	-

照会リストをダウンロードする

現在照会中のSession TableリストをCSVファイルでダウンロードできます。Session Tableセクションの右上の  ボタンを選択します。

ダウンロードしたファイル名の形式は次のとおりです。

- アクティブセッション: activeSession_HH_MM_SS.csv
- ロックツリー: lockTree_HH_MM_SS.csv

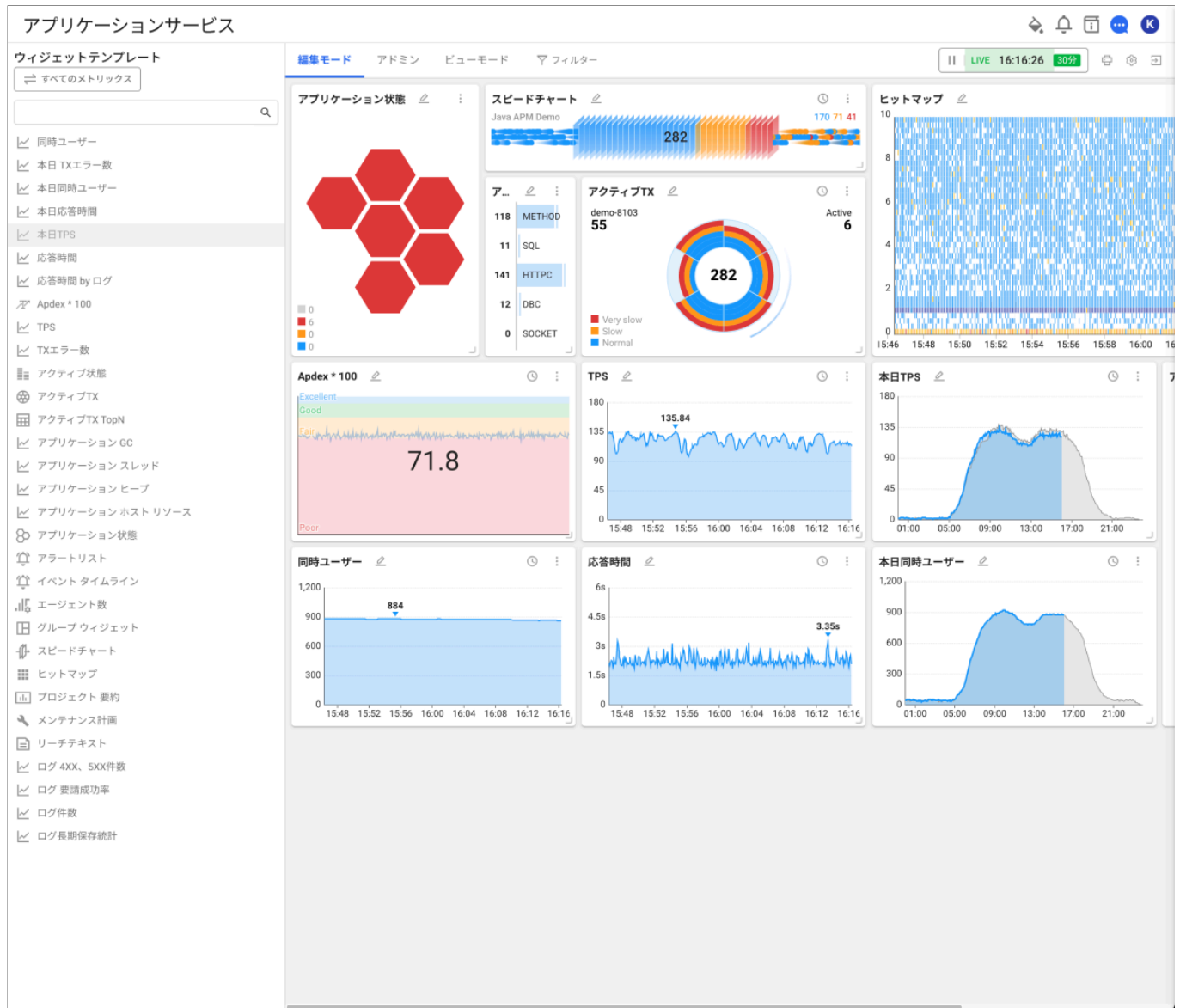
別ウィンドウで見る

Session Tableリストを別ウィンドウで開き、より広い画面で確認できます。Session Tableセクションの右上の  ボタンを選択します。

Flexボード

Flexボードはユーザー定義型の統合ダッシュボードです。アプリケーション、サーバー、データベース、コンテナなどのプロジェクトのデータを使用して、リアルタイムのダッシュボードが作成できます。

事前構成されたテンプレートで初期設定を迅速に完了し、希望する形式のダッシュボードを作成できます。データウィジェットを追加したりプロパティを修正したりして、必要な形式で表示することができ、フィルタリング機能によって監視対象を絞り込むこともできます。時間範囲を指定すれば特定の時点のデータを確認でき、補助チャートを活用することで多様な方法で分析できます。ダッシュボードはお気に入りに登録して簡単にアクセスでき、カスタマイズしたダッシュボードは他のアカウントにコピーして再利用することも可能です。



Flexボードは、次のアクセスパスによってさまざまな用途があります。

- ホーム画面 > 統合Flexボード

- ウィジェットを作成すると、表示可能なすべてのプロジェクトを選択オプションとして提供します。
- ユーザーアカウントにダッシュボードが保存され、他のユーザーにコピーする機能を使用して共有できます。
- 個人アカウントダッシュボードで、権限による影響はありませんが、読み取り専用として共有されたダッシュボードの場合は修正できません。

- ホーム画面 > プロジェクト選択 > **ダッシュボード** > **Flexボード**
 - ウィジェットを作成すると対応するプロジェクトの情報が自動入力されます。
 - プロジェクトメンバーで作成したFlexボードは、自動共有されます。
 - プロジェクト**修正権限**、または**プロジェクトのFlexボードの修正権限**、**Site Admin**権限を持つユーザーが修正できます。

✔ Flexボードの編集権限を持つユーザーは、以下の機能を利用できます。

- ダッシュボードをJSONファイルとしてエクスポート/インポート
- ダッシュボード内のデータリクエストおよびレスポンス内容を確認
- ウィジェット設定オプションをJSON形式で参照および修正

❗ プロジェクト内の**Flexボード**メニューでは、ダッシュボードの変更権限のあるユーザーのみが**編集モード**および**アドミンモード**、**フィルター**機能にアクセスできます。

- アクセス可能なメンバー権限
 - プロジェクト**修正権限**
 - **プロジェクトのFlexボードの修正権限**
 - **Site Admin**権限
- **ビューモード**および**フィルター**機能の詳細については、[次の文書](#)を参照してください。
- メンバー権限の詳細については、[次の文書](#)を参照してください。

Flexボードの使用方法

- ✔ ダッシュボードはメトリクスウィジェットを含むさまざまなウィジェットテンプレートを活用して構成できます。
- 各テンプレートの特徴と使用方法については、[ウィジェットテンプレート](#) ドキュメントで詳しく確認できます。

事前テンプレートを適用する

事前構成テンプレートで初期設定を素早く進め、必要に応じて修正して活用してください。

1. **Flexボード** メニューを選択します。
2. 左側の **テンプレート** 一覧から使用するテンプレートを選択します。
3. **Flexボード管理** ウィンドウが表示されたら、**名前変更** に希望の名前を入力します。
 - **レイアウト** でウィジェットを選択し、ドラッグして希望の位置へ移動できます。
4. **保存** ボタンをクリックします。選択したテンプレートを基準に Flex ボードを作成します。
 - 作成したボードは **Flexボード** のボード一覧で確認できます。

Flex ボードを作成する

Flex ボードを作成し、ウィジェットを配置して自分だけのダッシュボードを作れます。

1. **Flexボード** で右上の **ダッシュボード生成** ボタンをクリックします。
2. **ダッシュボード生成** ウィンドウが表示されたら、入力ボックスにダッシュボードの **名前** を入力します。
3. ウィジェットの配置方式を選択し、**ダッシュボード生成** ボタンをクリックします。
 - **固定レイアウトFlexボード**: ピクセル基準の自由配置
 - **レスポンシブFlexボード**: ブラウザサイズ基準のグリッドレスポンシブレイアウト

(グリッドを設定してから **ダッシュボード生成** ボタンをクリック)

4. **ウィジェットテンプレート** 選択画面で、ダッシュボードに配置するウィジェットを選択します。

- メトリクスウィジェットを追加する場合は、**ウィジェットテンプレート** の右側にある **すべてのメトリクス** ボタンをクリックします。

- ❗ **ウィジェットテンプレート**: 一般的な監視状況で重要となる指標を要約し、事前定義されたウィジェット一覧から選択できます。
- **すべてのメトリクス**: ユーザーのプロジェクトで収集中のすべてのメトリクスデータを基準にウィジェットを作成します。メトリクスウィジェットの詳細は[次のドキュメント](#)を参照してください。

5. 配置したウィジェットの位置を移動したりサイズを調整します。

- ウィジェット上部で十字カーソルが表示されたら、クリックしたままドラッグして位置を移動します。
- ウィジェット右下で矢印カーソルが表示されたら、クリックしたままドラッグしてサイズを調整します。

6. 追加するウィジェットの配置が完了したら、**ビューモード** を選択してレイアウトを確認します。レイアウトを再編集する場合は、**編集モード** を選択してウィジェット配置を変更します。

7. すべての工程が完了したら、右上の (**リストへ**) アイコンをクリックします。

- **Flexボード** の **ダッシュボードリスト** で作成したボードを確認できます。作成したボードを選択して新しいFlexボードを確認してください。

- ❗ お客様のご要望を反映してダッシュボードテンプレートやウィジェットの追加を検討しています。必要なテンプレートまたはウィジェットがあれば support@whatap.io までご連絡ください。

Flex ボードを編集する

- ❗ **Flex ボード編集権限**



- **ホーム > 統合Flexボード:** 個人アカウントのダッシュボードで権限の影響はありませんが、閲覧専用で共有されたダッシュボードは編集できません。
- **ホーム > プロジェクト選択 > ダッシュボード > Flexボード:** プロジェクトの編集権限、プロジェクトのフレックスボード編集権限、または Site Admin 権限のユーザーが編集できます。

メンバー権限の詳細は[次のドキュメント](#)を参照してください。

1. **Flexボードのダッシュボードリスト**で、 編集アイコンをクリックします。画面右側に **Flex ボード管理** ウィンドウが表示されます。
2. **Flex ボード管理** ウィンドウで、名前、プロジェクト、レイアウト、widget.json、option.json を設定します。
 - **名前変更:** ダッシュボード名を編集します。
 - **プロジェクト:** ダッシュボードのウィジェットがデータを取得するプロジェクトを選択します。



プロジェクト オプションは **ホーム > 統合Flexボード** メニューでのみ設定できます。

統合Flexボード から入った場合は、ウィジェットごとにデータ取得元のプロジェクトを選択できます。

1. ウィジェット右上の ボタンをクリックし、目的のプロジェクトを検索して単一または複数選択します。
2. プロジェクトを選択したら **適用** ボタンをクリックします。

• レイアウト

- ウィジェットをドラッグして位置を移動します。
- ウィジェット右下をマウスでクリックしたままドラッグしてサイズを調整します。
- ウィジェットを削除するには、ウィジェット右上の アイコンをクリックします。
- **ウィジェット.json:** ダッシュボードのウィジェット設定を JSON 形式で読み込みます。 アイコンをクリックすると JSON をコピーします。
- **オプション.json:** ダッシュボードのオプション設定を JSON 形式で読み込みます。 アイコンをクリックすると JSON をコピーします。

Flex ボードを削除する

1. Flexボード>ダッシュボードリスト で、削除する項目の  アイコンをクリックします。
2. 確認メッセージで **削除** ボタンをクリックします。

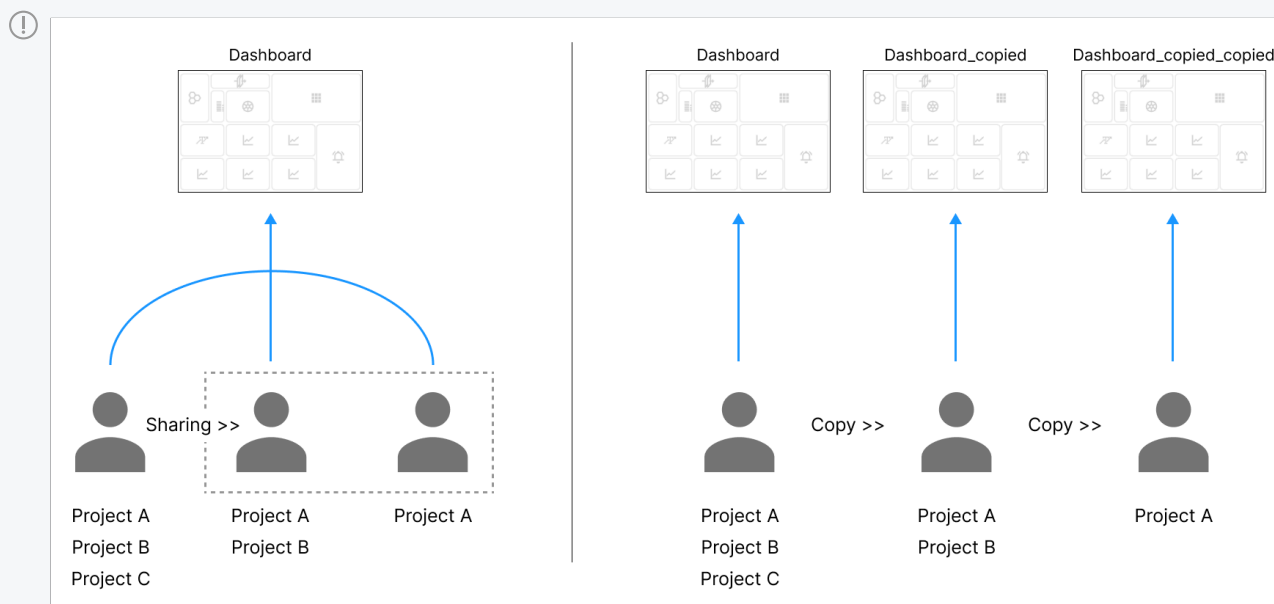
 ダッシュボードの削除は所有者と共有を受けたユーザーで動作が異なります。

- ダッシュボード所有者がダッシュボードを削除すると、共有されたすべてのユーザーアカウントでもそのダッシュボードが削除されます。
- 共有を受けたユーザーがダッシュボードを削除すると、そのユーザーの **ダッシュボードリスト** のみから削除されます。元のダッシュボードは維持されます。

Flex ボードを共有する

 Flex ボードの共有機能は **ホーム>統合Flexボード** メニューでのみ使用できます。
プロジェクトの **Flexボード** メニューでは提供されません。

 共有とコピーの違い



- 共有: 共有されたメンバーは同じダッシュボードを閲覧します。
- コピー: ダッシュボードを他のメンバーまたは自分にコピーすると、コピーされた別個のダッシュボードを閲覧します。
- 読取り専用 で共有/コピーされたダッシュボードは編集できません。編集モード で共有/コピーされた場合のみ編集できます。
- 読取り専用 で共有を受けたメンバーも、そのダッシュボードを 閲覧専用で再共有 したり コピー できます。
- ダッシュボードに含まれるプロジェクトのうち **少なくとも1つ** のプロジェクトに所属しているメンバーにのみ共有できます。共有またはコピー先のメンバーがプロジェクトに所属しているか確認してください。



統合Flexボード で作成した Flex ボードを他のメンバーと共有できます。所有者が共有されたダッシュボードを編集すると、共有を受けたすべてのメンバーに同じ変更が反映されます。

1. 統合Flexボード の **ダッシュボードリスト** で、共有するダッシュボードの  アイコンをクリックします。
2. **ダッシュボード共有** ウィンドウが表示されたら、共有するメンバーを1名または複数名選択するか、アカウントを直接入力します。
 - **読取り専用** を選択すると、共有を受けたメンバーはダッシュボードを編集できません。

- 共有対象は、ダッシュボードに含まれるプロジェクトのうち少なくとも1つのプロジェクトに所属している必要があります。

3. 共有 ボタンをクリックします。

- 共有された項目には **共有** タグが表示されます。**共有** タグにマウスを載せると、共有したメンバー情報を確認できます。
- 閲覧専用で共有された項目には **読取り専用** タグが表示されます。

ダッシュボードを共有で受け取った場合	ダッシュボードを共有した場合（ダッシュボード
 ダッシュボード所有者 owner@whatap.io 更新日時: 2023/06/14 10:22:05 共有 RUM ページ 5 ダッシュボード共有メンバーリスト 更新日時: 2023/06/14 10:19:52 共有 読取り専用 更新日時: 2023/06/05 16:41:36	 ダッシュボード所有者 demoowner@whatap.io 更新日時: 2023/07/06 16:25:06 共有 demoowner@whatap.io ダッシュボード共有メンバーリスト 更新日時: 2023/06/14 10:22:05 共有 更新日時: 2023/06/14 10:19:52

✔ ダッシュボード共有の条件

- ダッシュボード共有機能は **ホーム > 統合Flexボード** のみで提供します。
- だれでも自分のダッシュボードを共有でき、共有で受けたダッシュボードを再共有やコピーできます。
- 読取り専用** で受けたダッシュボードは編集できませんが、**編集モード** で受けた場合は編集できます。
- 所有者** がダッシュボードを削除すると、共有を受けたすべてのアカウントで該当ダッシュボードが削除されます。
- 共有を受けたユーザー** が削除した場合は、そのユーザーの **ダッシュボードリスト** のみから削除され、元のダッシュボードは維持されます。
- 共有で受けたダッシュボードに含まれるプロジェクトのうち、**ビュー** 権限がないプロジェクトはデータを参照できません。

Flex ボードをコピーする

ユーザーは Flex ボードをコピーして他のメンバーに渡せます。

1. 統合Flexボード のダッシュボードリスト で、 アイコンをクリックします。
2. コピーオプションを選択します。
 - **自分にコピー:** 自分のアカウントにコピーします。
 - a. `_copied` 接尾辞が付いた項目が **ダッシュボードリスト** に追加されます。
 - **他人にコピー:** 他のメンバーのアカウントにコピーします。
 - a. **+** **アカウント** をクリックします。
 - b. コピー先メンバーのメールを選択し、**ダッシュボードコピー** ボタンをクリックします。
 - c. そのメンバーの **ダッシュボードリスト** に `_copied` 接尾辞が付いた項目が追加されます。

JSON ファイルで共有する

Flex ボードの設定を JSON ファイルとして他のメンバーに渡したり、他のメンバーの設定を取り込みます。

エクスポート

1. 統合Flexボード またはプロジェクトの **Flexボード** メニューに移動します。
2. **ダッシュボードリスト** で  ボタンをクリックします。
3. エクスポートオプションを選択します。
 - **プロジェクト情報を含む:** ダッシュボードに設定されたプロジェクト情報を JSON ファイルに含めて共有
 - **プロジェクト情報を含まない:** ダッシュボードに設定されたプロジェクト情報を JSON ファイルに含めずに共有
4. JSON ファイルをダウンロードし、共有先のメンバーに渡します。

インポート

1. 統合Flexボード またはプロジェクトの Flexボード メニューに移動します。
2. 画面右上の  インポート ボタンをクリックします。
3. ダウンロードした JSON ファイルを選択します。

❗ この機能は、プロジェクトの 編集 権限を持つメンバーのみ利用できます。

Flex ボードの画面モード

Flex ボードの画面モードに応じて提供機能が異なります。

❗ アクセス可能なメンバー権限

プロジェクト内の Flexボード メニューでは、編集権限のあるユーザーのみが編集モード、アドミン、フィルター 機能にアクセスできます。メンバー権限の詳細は[次のドキュメント](#)を参照してください。

- プロジェクト編集 権限
- プロジェクトのFlexボードの修正 権限
- Site Admin 権限

モード種類

Flexボード は画面モードにより設定できる機能が異なります。

- **編集モード:** ウィジェットテンプレート一覧からレイアウトにウィジェットを配置し、ウィジェットの編集、削除、位置調整ができます。
- **アドミン:** ウィジェットの詳細プロパティを JSON 形式で管理できます。レイアウトに配置したウィジェット右上の  ボタンを選択します。
- **ビューモード:** ウィジェットの配置、サイズ調整、詳細オプションなどの設定を完了後、ダッシュボードのように各ウィジェットのデータをリアルタイムで監視できます。

共通機能

- データフィルタリング
- リアルタイムデータの参照範囲設定
- 印刷モード
- Flex ボード管理
- データ結合オプション

データフィルタリング

データフィルタリングで監視対象を絞り、効率的な Flex ボードを構成できます。

1. 画面上部の **▽ フィルター** タブをクリックします。
2. 画面右に表示される **フィルター** ウィンドウでフィルタリング対象を選択します。すべて選択するには **すべて選択** を選択します。
 - **エージェント オプション**: エージェント種別またはアプリケーション一覧を選択、**検索** 機能あり
 - **メトリクス オプション**: プロジェクトまたはエージェント種別を選択、**+ And** 条件の追加や、**含む**、**除外** による文字列ベースの自動選択が可能
3. フィルタリング対象を選択したら、**適用** ボタンをクリックします。
4. よく使う条件は **フィルター保存** ボタンで保存します。
 - a. フィルター名を入力し、**保存** を選択します。**保存フィルター** は **フィルター** ウィンドウ下部に表示されます。

フィルタリング条件

フィルタリング条件はエージェントタイプとメトリクスタイプに分かれます。

エージェントは基本的な監視対象のみ選択肢として提供し、メトリクスはプロジェクトで収集したすべてのメトリクスデータのタグを選択肢として提供します。メトリクスタイプは、非エージェント形式でデータを収集し、その後フィルタリングを適用する場合に有用です。メトリクスの詳細は[次のドキュメント](#)を参照してください。

タイプ	エージェント	メトリクス
選択オプション	WhaTap プロジェクトの監視対象 (アプリケーション、サーバー、データベース、 コンテナ、Kubernetes の POD と NODE)	エージェントを含むすべてのメトリクスタグ
適用範囲	すべてのウィジェット	メトリクスデータを使用するすべてのウィジェット
フィルタリング方式	チェックボックス	一致, 含む, 除外

参照時間を設定する

Flexボード の画面右上の時間セクターで、ウィジェットのデータ参照時間を設定できます。

- : リアルタイムモードを停止して特定の参照時間に設定
- : リアルタイムモードに戻す

印刷する

Flex ボード画面を PDF で保存して印刷できます。

1. Flex ボード画面の右上にある  アイコンをクリックして印刷モードに切り替えます。
2. オプションを設定します。
 - **フィルターオプション表示:** 画面のフィルターオプションを表示/非表示に切り替え
 - **{mode}方向に切換え / {mode}方向に切換え:** 画面を横向きまたは縦向きに変更
3. 完了後、**PDFダウンロード** を選択します。
4. 印刷モードを終了するには、右上の **× 印刷モード終了** をクリックします。

Flex ボード管理

Flex ボードのレイアウト変更、ウィジェットの位置移動やサイズ調整、名称変更などの管理機能を提供します。Flex ボード管理 機能は、**ダッシュボード > Flexボード メニューのダッシュボードリスト**で  アイコンをクリックして入

ることもできます。

1. Flex ボード画面の右上の  アイコンをクリックすると、**Flexボード管理** ウィンドウが開きます。

- **名前変更:** Flex ボード名の変更
- **レイアウト:** ウィジェットの位置・サイズ調整および削除
- **他のユーザーの修正を許可しない:** 変更制限オプション
 - 編集権限のあるメンバーは編集・削除・変更が可能で、オプションを有効化すると **他のユーザーの修正を許可しない: On** に変更されます
- **ウィジェット.json:** Flex ボードに含まれるウィジェットを JSON 形式で編集・管理
- **オプション.json:** Flex ボードの設定を JSON 形式で編集・管理（プロジェクト情報、Flex ボードフィルター、時間選択などの設定値）

 **ウィジェット.json と オプション.json** 機能は、ダッシュボード管理者権限のユーザーにのみ提供します。

2. すべての設定を完了し、**保存** ボタンをクリックします。

ダッシュボードをお気に入りに登録する

作成した Flex ボードはお気に入りに登録して素早くアクセスできます。**ダッシュボード > Flexボード** の **ダッシュボードリスト** で、お気に入り登録するダッシュボードの ☆ ボタンをクリックすると、★ に変わります。

ウィジェットの管理

Flexボードに配置したウィジェットを管理して、ダッシュボードをユーザーがカスタマイズします。

- ✓ ダッシュボードはメトリクスウィジェットを含むさまざまなウィジェットテンプレートを活用して構成できます。
各テンプレートの特徴と使用方法については、[ウィジェットテンプレート](#) ドキュメントで詳しく確認できます。

ウィジェット追加

作成済みのFlexボードにウィジェットを追加できます。

1. Flexボードメニューのダッシュボードリストで、ウィジェットを追加するダッシュボードを選択します。
 2. 画面左上の編集モードを選択します。
 3. 左側のウィジェットテンプレート一覧から目的のウィジェットを選択します。メトリクスウィジェットを追加するには、すべてのメトリクスを選択してから、目的のウィジェットをクリックします。
- ホーム画面の統合Flexボードを選択し、ウィジェットを追加した場合は、プロジェクト選択ボタンをクリックして、収集対象のプロジェクトを選択します。
 - 追加するウィジェットがある場合は、同じプロセスを繰り返してください。
4. すべての手順が完了したら、画面上部のビューモードを選択してレイアウトを確認し、 (リストヘ) ボタンをクリックします。

ウィジェットの属性変更

Flexボードメニューのダッシュボードリストで、ウィジェットのプロパティを変更するボードを選択します。編集モードに入ります。

ウィジェット名の変更

1. 名前を変更するウィジェットの名前の右側の  ボタンをクリックします。
2. ウィジェットの題名、フォントサイズ、色を設定し、**保存**ボタンを選択します。
 - 保存する際に、**すべてのウィジェットに適用**トグルボタンの機能を活性化すると、フォントサイズと色が他のウィジェットにも適用されます。
 - ウィジェットのタイトルをロックするには  ボタンを選択し、**保存**ボタンをクリックします。

時間設定

1. 時間を設定するウィジェットの右上の  ボタンを選択します。
2. **時間選択**ウィンドウで時間を設定してください。
 - ダッシュボード画面で設定した時間を適用するには、**ダッシュボード設定に従う**をチェックします。ダッシュボードの時間設定は右上にあります。
 - 特定の時間を設定する場合は、**ダッシュボード設定に従う**のチェックを外します。＜または＞ボタンを使用して照会する時間を設定します。
3. 時間設定が終了したら **×** ボタンをクリックします。

 ウィジェットのデータタイプによっては、時間設定機能をサポートしていない場合があります。

出力データオプションの設定

チャート型ウィジェットの出力データオプションを設定して、ユーザーが必要なデータを選択できます。

1. ウィジェットの右上にある  アイコンをクリックして、出力データのオプションを設定します。
 - プロジェクト、エージェント、アプリケーションなどを選択してデータをフィルタリングできます。
 - 個々のデータまたはマージされたデータを選択して、チャートに表示されるグラフのタイプを変更できます。
 -  >  を選択します。**チャート設定**画面で詳細オプションを変更できます。設定できるオプションは次

のとおりです。

- **チャートタイプ**：集計する現在の数値をチャートに表示するオプションです。
 - **y軸 チャート最大値**：Y軸にチャートの最大値を自動または固定で選択できます。
 - **水平線設定**：追加ボタンを選択し、しきい値または範囲を水平線として表示できます。複数を追加して適用できます。水平線を追加した後は、**適用**ボタンを選択してチャートに反映してください。
 - **チャートの最大値を表示**：チャートに表示されるグラフの最大値を表示します。
 - **サブチャート**：ウィジェットのメインチャートにサブチャートを追加できます。サブチャートの種類を選択し、レイアウトとサイズ、データのマージ有無を選択します。他のウィジェットにも適用する場合は、**すべてのウィジェットに適用**ボタンを選択します。サブチャートは、データをさまざまな方法で確認するとき便利です。
2. オプション設定がすべて完了したら、**チャート設定**画面の左上の×ボタンをクリックするか、**チャート設定**領域の外側をクリックします。

❗ ウィジェットのデータタイプによって選択できるオプションは異なる場合があります。

メトリクスウィジェットデータの変更

メトリクスウィジェットのデータ照会の条件を変更できます。

1. **Flexボード > ダッシュボードリスト**で、ウィジェットデータを変更するダッシュボードを選択します。
2. **編集モード**で、変更したいウィジェットの右上にある  アイコンをクリックします。
3.  アイコンをクリックすると、データ設定編集ウィンドウが表示されます。
4. 次を参照して設定を変更するには、**適用アイコンウィジェットに適用**ボタンをクリックします。
 - ウィジェットを作成すると、デフォルトで生成されるメトリクスのオプションがタブに表示されます。
 - メトリクス オプションには、**カテゴリ**と**フィールド**、**タグ**が含まれます。
 - **カテゴリ**項目は現在、変更を加える機能を制限しています。
 - **フィールド**、**タグ**項目は複数選択できます。選択したタグオプションでチャートデータをグループ化します。
 -  ボタンをクリックすると、元のデータを表示できます。

- オブジェクトのマージおよび時間マージ項目の詳細については、[データマージオプション](#)を参照してください。

データマージオプション

Flexボードのすべてのメトリクス一覧に配置されたマトリクスウィジェットでは、データのマージオプションを提供します。データのマージは、オブジェクトのマージと時間マージ方法を提供します。

• オブジェクトのマージ

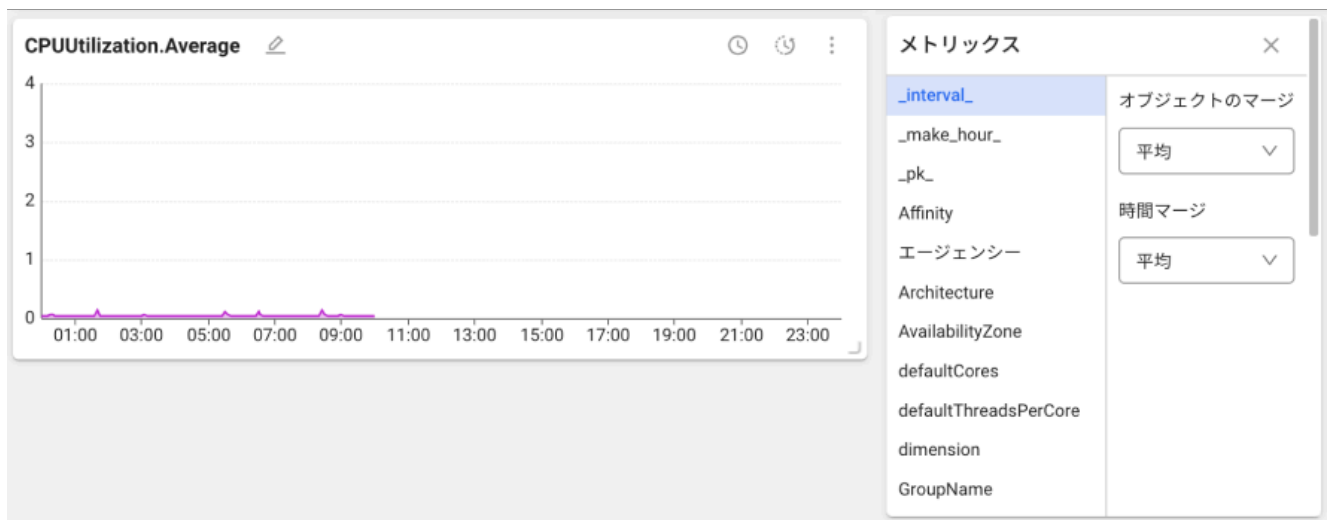
左側で選択したタグ(例：エージェント名)に基づいてチャートデータをグループ化するために使用します。この場合、異なるフィールド(例：CPU)値を持つデータ間でタグが一致する場合、そのデータは結合されます。

• 時間マージ

主に検索時間が長い場合に使用します。ソースデータとフィールド値が同じデータを時間範囲に基づいて一定の時間間隔(5分または1時間)でデータを結合します。この結合方法は、時間結合と呼ばれます。

• AWS CloudWatchメトリクス

AWSモニタリング指標の場合、統計手法はフィールド名の後にサフィックス(接尾辞)として追加されます。データマージの既定値は、このサフィックスを参照しています。



1. 編集モードですべてのメトリクスボタンを選択し、ウィジェットテンプレート一覧をすべてのメトリクス一覧に変更します。
2. メトリクスウィジェットをレイアウトに配置します。

3. メトリクス ウィジェットの右上にある  ボタンを選択します。
4. タグ(例: エージェント名)オプションを選択すると、データ結合オプションが表示されます。ウィジェットのサイズによっては、メトリクス オプションの位置が異なる場合があります。

ウィジェットのコピー

1. ウィジェットの右上にある  ボタンをクリックします。
2.  ボタンをクリックします。
3. コピー画面が表示されたら、**+ ウィジェット追加**を選択します。
4. 一覧にあるウィジェットの名前を変更します。
5. **適用**ボタンを選択します。
6. **編集モード**画面でコピーしたウィジェットで確認できます。

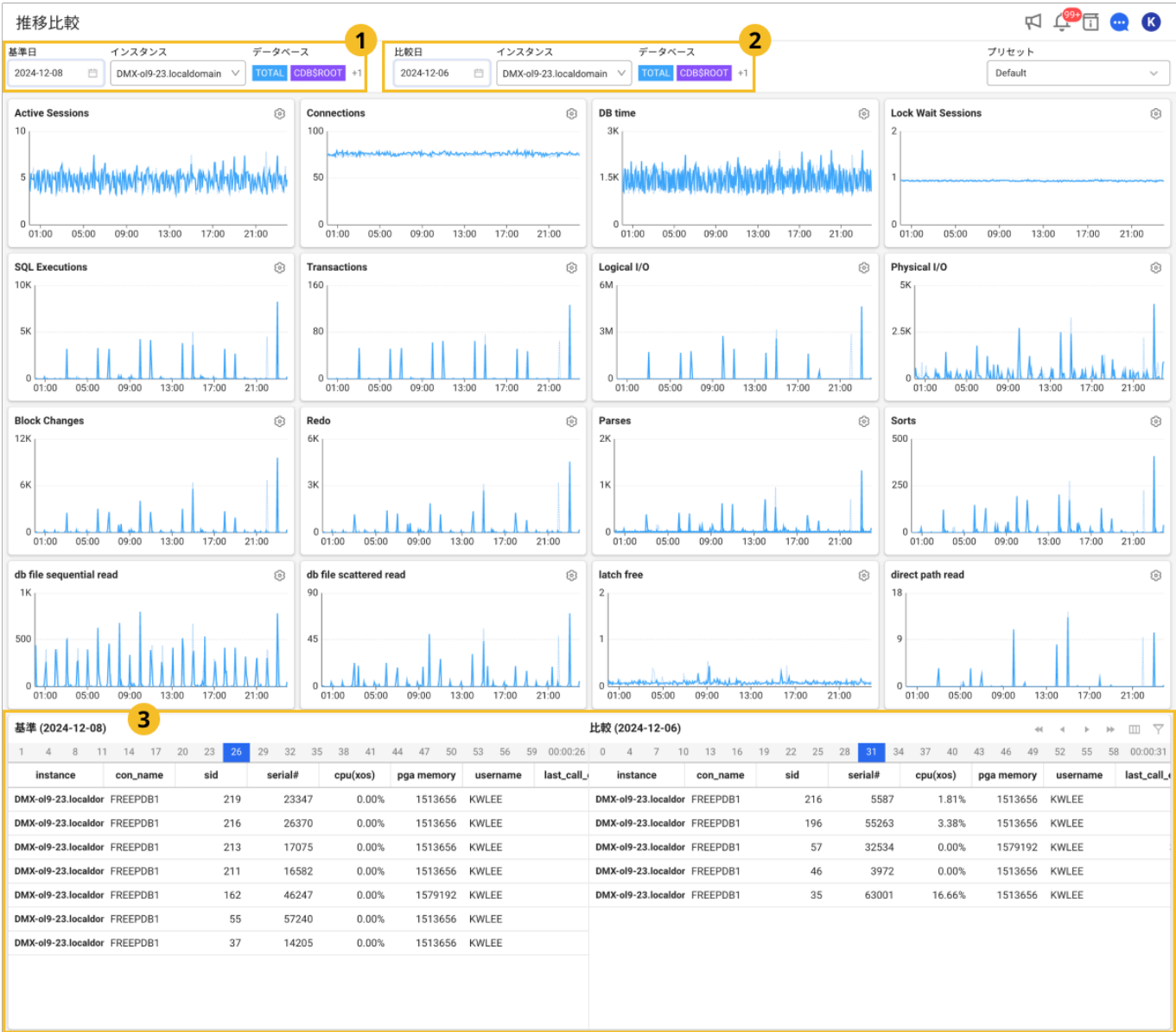
ウィジェットの削除

1. ウィジェットの右上にある  ボタンをクリックします。
2.  ボタンをクリックします。
3. 確認メッセージの画面が表示されると、**削除**ボタンを選択します。削除を取り消すには**キャンセル**ボタンをクリックします。

推移の比較

分析 > 推移比較 で、データベースの主要指標を時間帯ごとに確認し、パフォーマンスの変化を追跡します。
異なる日付の指標を比較することで、運用の傾向を分析できます。
アクティブセッション テーブルでは、現在実行中のセッションと長時間実行中のセッションの両方を確認できます。

基本画面ガイド



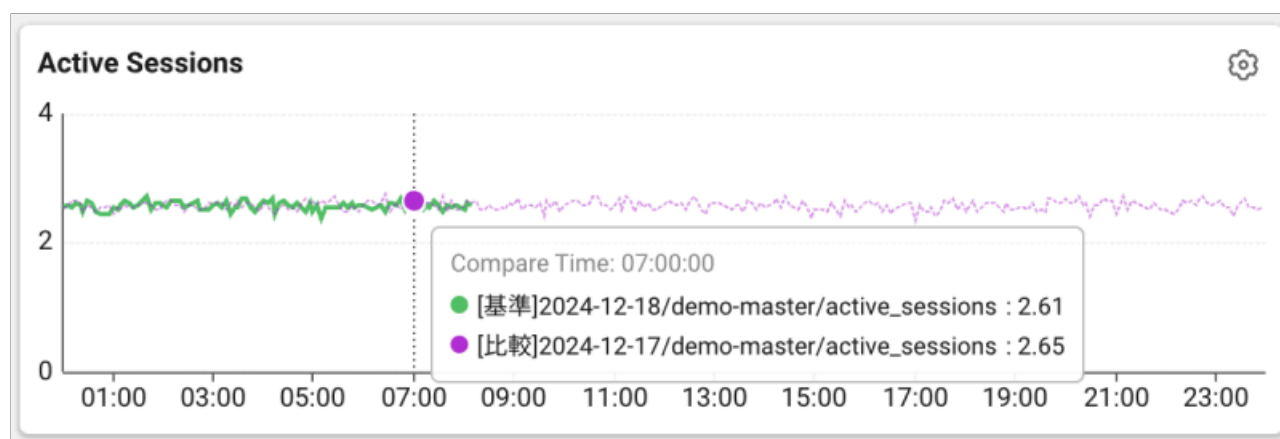
- ① 基準日、② 比較日、インスタンス項目を選択すると、すべてのウィジェットに選択した項目を基準に、データが自動的に反映されます。
- ウィジェットに表示されている指標の意味を知りたい場合は、名前の横の ⓘ ボタンを選択するか、次の文書を参照してください。

① ⓘ ボタンのツールチップ機能は多言語ではサポートされていません。

- 他のウィジェットの位置を変更する場合は、ウィジェットの上部を選択した状態でドラッグします。ただし、ウィジェットのサイズは変更できません。
- ③ アクティブセッションテーブルの左側には、**基準日**、右側には、**比較日**に該当するデータを表示します。
- ③ アクティブセッションテーブル一覧の文字の色は黒 → オレンジ → 赤の順でセッションの実行速度が遅いことを意味します。

比較対象の選択

① **基準日、インスタンス**を選択し、② **比較日、インスタンス**を選択します。チャートの上にマウスをオーバーすると表示されるツールチップを通じて、その時点の詳細データを確認できます。ツールチップの内容は、マウスの位置に応じて値が更新されます。



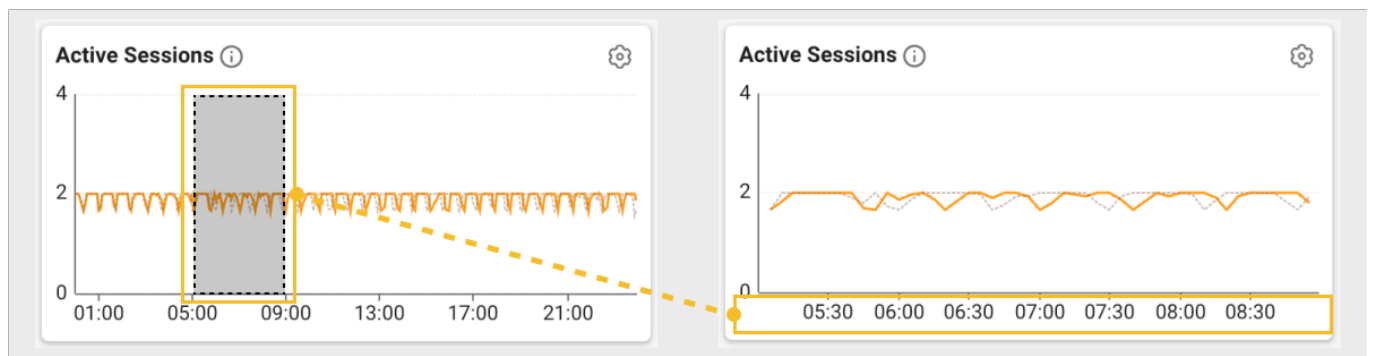
- シリーズごとに固有の色が表示され、視覚的に簡単に区別できます。実線は**基準日**のデータ、点線は**比較日**のデータを意味します。
- **Compare Time**は、現在マウスポインターが存在するデータの時間値を表示します。
- 選択したデータ時点のメトリクス値を表示し、シリーズごとに区別して提供します。各シリーズには、データポイントの日付とエージェント(インスタンス)名、指標名、値が含まれます。

- ① ⓘ
- 画面上部の**データベース**項目で比較したいデータベースを選択することもできます。
 - 画面上部の**データベース**項目の色は、グラフチャートで同じ色で表されます。

アクティブセッション情報の比較

基準 (2024-12-08)								比較 (2024-12-06)																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																	
1	4	8	11	14	17	20	23	26	29	32	35	38	41	44	47	50	53	56	59	62	65	68	71	74	77	80	83	86	89	92	95	98	101	104	107	110	113	116	119	122	125	128	131	134	137	140	143	146	149	152	155	158	161	164	167	170	173	176	179	182	185	188	191	194	197	200	203	206	209	212	215	218	221	224	227	230	233	236	239	242	245	248	251	254	257	260	263	266	269	272	275	278	281	284	287	290	293	296	299	302	305	308	311	314	317	320	323	326	329	332	335	338	341	344	347	350	353	356	359	362	365	368	371	374	377	380	383	386	389	392	395	398	401	404	407	410	413	416	419	422	425	428	431	434	437	440	443	446	449	452	455	458	461	464	467	470	473	476	479	482	485	488	491	494	497	500	503	506	509	512	515	518	521	524	527	530	533	536	539	542	545	548	551	554	557	560	563	566	569	572	575	578	581	584	587	590	593	596	599	602	605	608	611	614	617	620	623	626	629	632	635	638	641	644	647	650	653	656	659	662	665	668	671	674	677	680	683	686	689	692	695	698	701	704	707	710	713	716	719	722	725	728	731	734	737	740	743	746	749	752	755	758	761	764	767	770	773	776	779	782	785	788	791	794	797	800	803	806	809	812	815	818	821	824	827	830	833	836	839	842	845	848	851	854	857	860	863	866	869	872	875	878	881	884	887	890	893	896	899	902	905	908	911	914	917	920	923	926	929	932	935	938	941	944	947	950	953	956	959	962	965	968	971	974	977	980	983	986	989	992	995	998	1001	1004	1007	1010	1013	1016	1019	1022	1025	1028	1031	1034	1037	1040	1043	1046	1049	1052	1055	1058	1061	1064	1067	1070	1073	1076	1079	1082	1085	1088	1091	1094	1097	1100	1103	1106	1109	1112	1115	1118	1121	1124	1127	1130	1133	1136	1139	1142	1145	1148	1151	1154	1157	1160	1163	1166	1169	1172	1175	1178	1181	1184	1187	1190	1193	1196	1199	1202	1205	1208	1211	1214	1217	1220	1223	1226	1229	1232	1235	1238	1241	1244	1247	1250	1253	1256	1259	1262	1265	1268	1271	1274	1277	1280	1283	1286	1289	1292	1295	1298	1301	1304	1307	1310	1313	1316	1319	1322	1325	1328	1331	1334	1337	1340	1343	1346	1349	1352	1355	1358	1361	1364	1367	1370	1373	1376	1379	1382	1385	1388	1391	1394	1397	1400	1403	1406	1409	1412	1415	1418	1421	1424	1427	1430	1433	1436	1439	1442	1445	1448	1451	1454	1457	1460	1463	1466	1469	1472	1475	1478	1481	1484	1487	1490	1493	1496	1499	1502	1505	1508	1511	1514	1517	1520	1523	1526	1529	1532	1535	1538	1541	1544	1547	1550	1553	1556	1559	1562	1565	1568	1571	1574	1577	1580	1583	1586	1589	1592	1595	1598	1601	1604	1607	1610	1613	1616	1619	1622	1625	1628	1631	1634	1637	1640	1643	1646	1649	1652	1655	1658	1661	1664	1667	1670	1673	1676	1679	1682	1685	1688	1691	1694	1697	1700	1703	1706	1709	1712	1715	1718	1721	1724	1727	1730	1733	1736	1739	1742	1745	1748	1751	1754	1757	1760	1763	1766	1769	1772	1775	1778	1781	1784	1787	1790	1793	1796	1799	1802	1805	1808	1811	1814	1817	1820	1823	1826	1829	1832	1835	1838	1841	1844	1847	1850	1853	1856	1859	1862	1865	1868	1871	1874	1877	1880	1883	1886	1889	1892	1895	1898	1901	1904	1907	1910	1913	1916	1919	1922	1925	1928	1931	1934	1937	1940	1943	1946	1949	1952	1955	1958	1961	1964	1967	1970	1973	1976	1979	1982	1985	1988	1991	1994	1997	2000	2003	2006	2009	2012	2015	2018	2021	2024	2027	2030	2033	2036	2039	2042	2045	2048	2051	2054	2057	2060	2063	2066	2069	2072	2075	2078	2081	2084	2087	2090	2093	2096	2099	2102	2105	2108	2111	2114	2117	2120	2123	2126	2129	2132	2135	2138	2141	2144	2147	2150	2153	2156	2159	2162	2165	2168	2171	2174	2177	2180	2183	2186	2189	2192	2195	2198	2201	2204	2207	2210	2213	2216	2219	2222	2225	2228	2231	2234	2237	2240	2243	2246	2249	2252	2255	2258	2261	2264	2267	2270	2273	2276	2279	2282	2285	2288	2291	2294	2297	2300	2303	2306	2309	2312	2315	2318	2321	2324	2327	2330	2333	2336	2339	2342	2345	2348	2351	2354	2357	2360	2363	2366	2369	2372	2375	2378	2381	2384	2387	2390	2393	2396	2399	2402	2405	2408	2411	2414	2417	2420	2423	2426	2429	2432	2435	2438	2441	2444	2447	2450	2453	2456	2459	2462	2465	2468	2471	2474	2477	2480	2483	2486	2489	2492	2495	2498	2501	2504	2507	2510	2513	2516	2519	2522	2525	2528	2531	2534	2537	2540	2543	2546	2549	2552	2555	2558	2561	2564	2567	2570	2573	2576	2579	2582	2585	2588	2591	2594	2597	2600	2603	2606	2609	2612	2615	2618	2621	2624	2627	2630	2633	2636	2639	2642	2645	2648	2651	2654	2657	2660	2663	2666	2669	2672	2675	2678	2681	2684	2687	2690	2693	2696	2699	2702	2705	2708	2711	2714	2717	2720	2723	2726	2729	2732	2735	2738	2741	2744	2747	2750	2753	2756	2759	2762	2765	2768	2771	2774	2777	2780	2783	2786	2789	2792	2795	2798	2801	2804	2807	2810	2813	2816	2819	2822	2825	2828	2831	2834	2837	2840	2843	2846	2849	2852	2855	2858	2861	2864	2867	2870	2873	2876	2879	2882	2885	2888	2891	2894	2897	2900	2903	2906	2909	2912	2915	2918	2921	2924	2927	2930	2933	2936	2939	2942	2945	2948	2951	2954	2957	2960	2963	2966	2969	2972	2975	2978	2981	2984	2987	2990	2993	2996	2999	3002	3005	3008	3011	3014	3017	3020	3023	3026	3029	3032	3035	3038	3041	3044	3047	3050	3053	3056	3059	3062	3065	3068	3071	3074	3077	3080	3083	3086	3089	3092	3095	3098	3101	3104	3107	3110	3113	3116	3119	3122	3125	3128	3131	3134	3137	3140	3143	3146	3149	3152	3155	3158	3161	3164	3167	3170	3173	3176	3179	3182	3185	3188	3191	3194	3197	3200	3203	3206	3209	3212	3215	3218	3221	3224	3227	3230	3233	3236	3239	3242	3245	3248	3251	3254	3257	3260	3263	3266	3269	3272	3275	3278	3281	3284	3287	3290	3293	3296	3299	3302	3305	3308	3311	3314	3317	3320	3323	3326	3329	3332	3335	3338	3341	3344	3347	3350	3353	3356	3359	3362	3365	3368	3371	3374	3377	3380	3383	3386	3389	3392	3395	3398	3401	3404	3407	3410	3413	3416	3419	3422	3425	3428	3431	3434	3437	3440	3443	3446	3449	3452	3455	3458	3461	3464	3467	3470	3473	3476	3479	3482	3485	3488	3491	3494	3497	3500	3503	3506	3509	3512	3515	3518	3521	3524	3527	3530	3533	3536	3539	3542	3545	3548	3551	3554	3557	3560	3563	3566	3569	3572	3575	3578	3581	3584	3587	3590	3593	3596	3599	3602	3605	3608	3611	3614	3617	3620	3623	3626	3629	3632	3635	3638	3641	3644	3647	3650	3653	3656	3659	3662	3665	3668	3671	3674	3677	3680	3683	3686	3689	3692	3695	3698	3701	3704	3707	3710	3713	3716	3719	3722	3725	3728	3731	3734	3737	3740	3743	3746	3749	3752	3755	3758	3761	3764	3767	3770	3773	3776	3779	3782	3785	3788	3791	3794	3797	3800	3803	3806	3809	3812	3815	3818	3821	3824	3827	3830	3833	3836	3839	3842	3845	3848	3851	3854	3857	3860	3863	3866	3869	3872	3875	3878	3881	3884	3887	3890	3893	3896	3899	3902	3905	3908	3911	3914	3917	3920	3923	3926	3929	3932	3935	3938	3941	3944	3947	3950	3953	3956	3959	3962	3965	3968	3971	3974	3977	3980	3983	3986	3989	3992	3995	3998	4001	4004	4007	4010	4013	4016	4019	4022	4025	4028	4031	4034	4037	4040	4043	4046	4049	4052	4055	4058	4061	4064	4067	4070	4073	4076	4079	4082	4085	4088	4091	4094	4097	4100	4103	4106	4109	4112	4115	4118	4121	4124	4127	4130	4133	4136	4139	4142	4145	4148	4151	4154	4157

特定時間の比較



特定の時間を拡大してデータを比較できます。画面に配置されたウィジェットから目的の時間帯を選択して、チャートをドラッグしてください。すべてのウィジェットとアクティブセッションテーブルに、ドラッグした特定の時間帯のデータを表示します。

プリセット

インスタンスモニタリングメニューから、ユーザーが設定したウィジェットの設定とアクティブセッションテーブルの配置状態をプリセットとして読み込むことができます。

❗ プリセット設定の詳細については、[次の文書](#)を参考にしてください。

カラム情報の案内

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

カラム情報に関する詳しい内容は、[次のリンク](#)を参照してください。

項目	説明
username	DBユーザー名
command	COMMAND_TYPE と同一、SQLのcommand number

項目	説明
status	セッションの状態
schemaname	スキーマ、usernameと同一
osuser	セッションに接続したOSのユーザー名
spid	OSプロセスの識別子
process	セッションに接続したクライアントプロセスID
logon_time	セッションが作成された時点
last_call_et	セッションがアクティブになっている場合、アクティブになってから経過した時間(秒)
sql_text	query、実行されているstatement、リテラル文字列を除外したSQL文
sql_param	query から分離したリテラル値
prev_sql_text	以前に実行したquery
prev_sql_param	以前に実行したquery param
program	クライアントで実行したプログラムの情報
type	session type - user background recursive (whatap.conf)で設定可能
seq#	現在または最後の待機を固有に識別する番号(待機するたびに増加)
event	セッションがキューに登録されている場合は、セッションが待機しているリソースまたはイベントの数
p1	最初の待機イベントのパラメータ(小数点)
p2	2番目の待機イベントのパラメータ(小数点)

項目	説明
p3	3番目の待機イベントのパラメータ(小数点)
wait_class	待機中のイベントのクラス名
wait_time	セッションが現在待機中の場合、値は0です。0より大きい場合は、最後の待機時間(millisecons)
seconds_in_wait	セッションが現在待機中の場合、セッションが待機のために待機した時間を意味します。セッションが待機中でない場合、最後の待機開始からの時間を意味します。
state	待機状態、WAITING, WAITED UNKNOWN TIME, WAITED SHORT TIME, WAITED KNOWN TIME
machine	セッションに接続したクライアントサーバー名(hostname)
port	クライアントポート番号
terminal	端末名
module	クライアントで実行したプログラムでユーザが設定したモジュール(module)名
アクション	現在実行されているモジュールのユーザが設定したアクション名
client_info	ユーザが設定したクライアント情報
client_identifier	ユーザが設定したクライアントの識別子
session logical reads	バッファキャッシュ内でデータを読み込んだ回数
physical reads	物理ディスクからデータを読み出した回数
excute count	SQLの実行数
parse count (hard)	構文分析の呼び出し合計数(実際の構文分析)
parse count (total)	構文分析呼び出しの総数(hard、soft、describe)

項目	説明
opened cursors current	現在開いているcursorの数
db block changes	データベース内のすべてのブロックの変更回数
session pga memory	セッションの現在のPGAサイズ(size)
undo_segid	undo情報id
undo_blk	undoブロックの数
undo_rec	undoレコードの数
sql_address	SQL文(statement)を識別する情報
sql_hash_value	SQL文(statement)を識別する情報
sql_id	現在実行中のSQL文(statement)のSQL識別子
sql_child_number	現在実行中のSQL文(statement)の下位番号
sql_exec_start	当該セッションで現在実行中のSQL実行は、開始点にあります。
sql_exec_id	SQL実行識別子
prev_sql_addr	最後に実行されたSQL文(statement)を識別する情報
prev_child_number	最後に実行されたSQL文(statement)の下位番号
prev_exec_start	最後に実行されたSQL文(statement)のSQL実行の開始時点
prev_exec_id	最後に実行されたSQL文(statement)のSQL実行識別id
plsql_entry_object_id	スタックの一番上にあるPL/SQL下位プログラムのオブジェクトid
plsql_entry_subprogram_id	スタックで一番上にあるPL/SQL下位プログラムのサブプログラムid

項目	説明
plssql_object_id	現在実行中のPL/SQLオブジェクトのサブプログラムid
taddr	トランザクションのアドレス(address)
lockwait	lock wait address、セッションが待機中のロックアドレス
row_wait_obj	ROW_WAIT_ROW# で指定された行を含むテーブルのオブジェクトID
row_wait_file	ROW_WAIT_ROW# で指定された行を含むデータファイルの識別子
row_wait_block	ROW_WAIT_ROW# で指定された行を含むブロックの識別子
row_wait_row	現在ロックされている行
pdml_status	値が ENABLED の場合、セッションが PARALLEL DML 有効モードです。値が DISABLE の場合、セッションで PARALLEL DML 有効モードに対応していません。値が FORCED の場合、セッションが PARALLEL DML を強制的に使用するように変更されたものです。
pq_status	値が ENABLED の場合、セッションが PARALLEL QUERY 有効モードです。値が DISABLE の場合、セッションで PARALLEL QUERY 有効モードに対応していません。値が FORCED の場合、セッションが PARALLEL QUERY を強制的に使用するように変更されたものです。
blocking_session_status	ブロックしているセッションの存在有無に関する詳細を提供
blocking_instance	ブロックしているセッションのインスタンス識別子
blocking_session	ブロックしているセッションのセッション識別子
final_blocking_session_status	最終ブロックセッションの存在有無に関する詳細を提供
final_blocking_instance	最終ブロックしているセッションのインスタンス識別子
final_blocking_session	最終ブロックしているセッションのセッション識別子

項目	説明
service_name	セッションのサービス名やDB名と類似
saddr	セッションアドレス
con_id	データに関連するコンテナのid(マルチデータベースをサポートする場合のみ)

ⓘ WhaTapは、クライアントに関する情報をデフォルトで保存します。

項目	説明
Instance	インスタンス（エージェント）名
sid	セッションID
serial#	セッションの一意な識別子と共に使用するセッションシリアル番号
holder_type	ロックを保持しているセッションのロックタイプ（例：TX、TMなど）
holder_mode	保持中のロックモード（例：0～6、6はexclusive）
waiter_type	ロックを待機しているセッションのロックタイプ
waiter_mode	現在要求中のロックモード
id1	ロック識別子1（ロック対象オブジェクトを識別するために使用、例：トランザクションIDなど）
id2	ロックの詳細情報またはバージョン情報を表します

次の項目は、XOSエージェントをインストールした場合に収集される指標です。

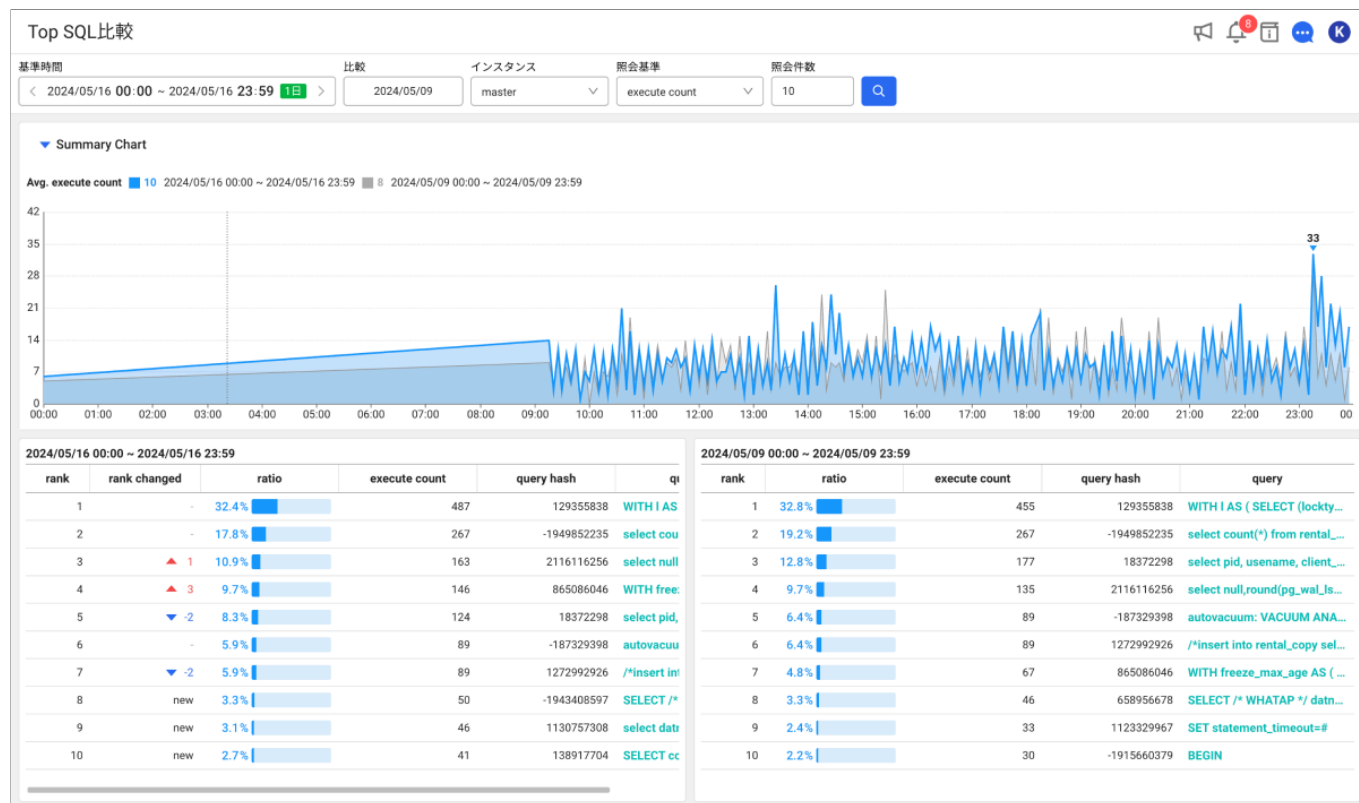
項目	説明	単位
pid	プロセスID	-

項目	説明	単位
cpuusage	プロセスが使用するCPU使用率	%
cputime	プロセスがこれまでに使用した累積CPU時間	秒
elapsed	プロセスが実行されてからの経過時間	秒
vsize	Virtual memory size. 全体の仮想メモリ使用量（RAM + Swap + 予約）	KB
rss	Resident Set Size. 現在のプロセスが占有している実際の物理メモリサイズ	KB
pss	Proportional Set Size. 共有メモリを含む実際の物理メモリ使用量	KB
state	プロセスの現在の状態コード（例：R、S、D、T、Z など）	-
ioread	累積読み込みバイト数（read_bytes）	バイト（B）
iowrite	累積書き込みバイト数（write_bytes）	バイト（B）
uid	このプロセスを実行したユーザーID	-
cmd	実行されたコマンドの短縮形式	-
longcmd	実行されたコマンドの完全な形式（フルパスおよび引数を含む）	-

Top SQL比較

ホーム画面 > プロジェクト選択 > 分析 > Top SQL比較

特定の日付を基準に設定された時間帯のSQLの推移を比較できます。ユーザは複数の基準を選択してSQLの性能を比較し**Summary Chart**を通じて時系列グラフで確認できます。また、画面下のテーブルリストでは、基準日付と比較日付の**Top SQL**順位と騰落を比較できます。



- **SQL性能把握:** ユーザーは**Top SQL比較**メニューを通じてSQLの性能を簡単に把握できます。SQLの実行回数、最大経過時間、総経過時間、待機時間などを比較することで、システムの性能問題を迅速に感知できます。
- **時間別の推移確認:** **Summary Chart**を通じて時間によるSQLの性能変化を視覚的に確認できます。これにより、ユーザーはシステムの負荷が発生した時間帯や性能低下が発生した区間を素早く把握して対応できます。
- **Top SQL比較:** 画面下のテーブルリストでは、基準日付と比較日付の**Top SQL**順位を比較できます。これにより、ユーザーはSQLの性能変化を確認し、比較して性能改善に必要な措置を取ることができます。

Top SQL比較

画面の上でTop SQL比較のための条件を設定した後、 ボタンを選択します。

- **基準時間:** 比較のための基準時間を設定します。最大24時間まで設定できます。緑のボタンをクリックすると、照会時間を選択できるオプションが表示されます。ユーザーが希望する時間を直接設定するには、日付と時間エリアを選択します。

- ①。照会できる時間範囲はm、最大24時間です。
- 。基準時間オプションの利用方法の詳細については。

- **比較:** 基準時間で設定した時間と比較する日付または時間を選択します。希望する条件を選択した後、**確認**ボタンを選択します。

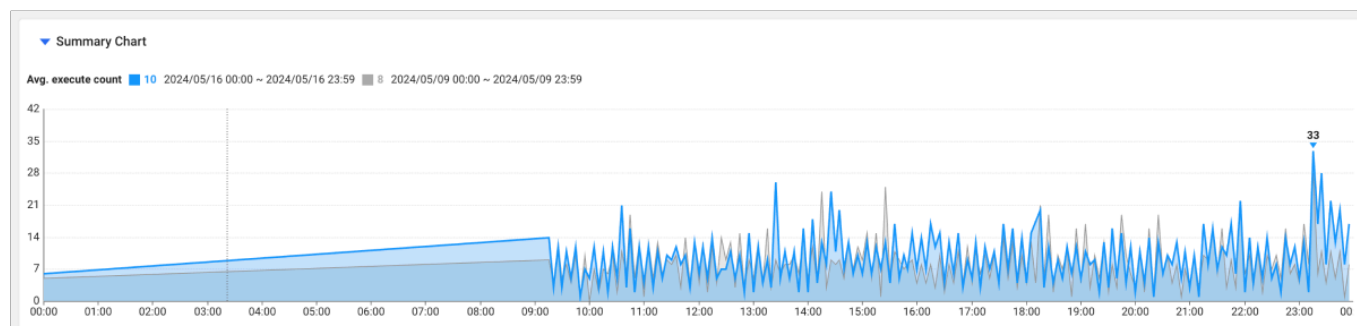
日付	時間
比較 2024/05/16 インスタンス master 日付 時間 1日前 2024/05/16 7日前 2024/05/10 30日前 2024/04/17 カスタマイズ 2024-05-16 キャンセル 確認	比較 2024/05/16 インスタンス master 日付 時間 (n) 時間前 1 キャンセル 確認

- **インスタンス:** SQL比較のためにインスタンス対象を選択します。
- **照会基準:** 比較基準を選択します。選択した項目に基づいてチャートを作成し、順位をソートします。
 - **excute count:** 実行回数に基づいて照会結果をソートします。
 - **elapsed max:** 最大経過時間に基づいて照会結果をソートします。

- **elapsed time:** 合計経過時間に基づいて照会結果をソートします。
- **elapsed wait:** 待ち時間に基づいて照会結果をソートします。
- **照会件数:** 画面の下にあるSQL順位テーブルのリストに表示される個数を設定してください。

Summary Chart

Summary Chartでは、基準日付と比較日付のSQL推移を比較できる時系列グラフを提供します。これにより、ユーザーは時間によるSQLの性能変化を容易に把握できます。



- ■ 青色のグラフ: 基準日付を意味し、右側の数値は平均値を意味します。
- ■ グレーのグラフ: 比較の日付を意味し、右側の数値は平均値を意味します。

SQL順位の比較

画面下のSQL順位テーブルでは、上位SQL一覧が表示されます。表示される個数は、画面上の**照会件数**で設定できます。

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

左側に配置されたテーブルは、基準日付に該当するSQL順位が、右側には比較日付に該当するSQL順位が表示されます。基準日付テーブルの**rank changed**カラムでは、比較日付に比べて変化した騰落を確認できます。

例えば、ユーザーが**elapsed time**(経過時間)を基準に比較したとき、比較日付に比べて基準日付に同じSQL文の経過時間がさらに増えた場合は、当該SQL文の性能を検討する必要があります。クエリ最適化またはインデックスを使用して性能を最適化してください。

クエリプレビュー

クエリプレビュー

SQL順位テーブルで**query**カラム項目にマウスオーバーすると、該当SQL文のクエリを確認できます。

ⓘ SQLの順位テーブルで**query**のカラム項目を選択すると**SQL詳細画面**が表示されます。

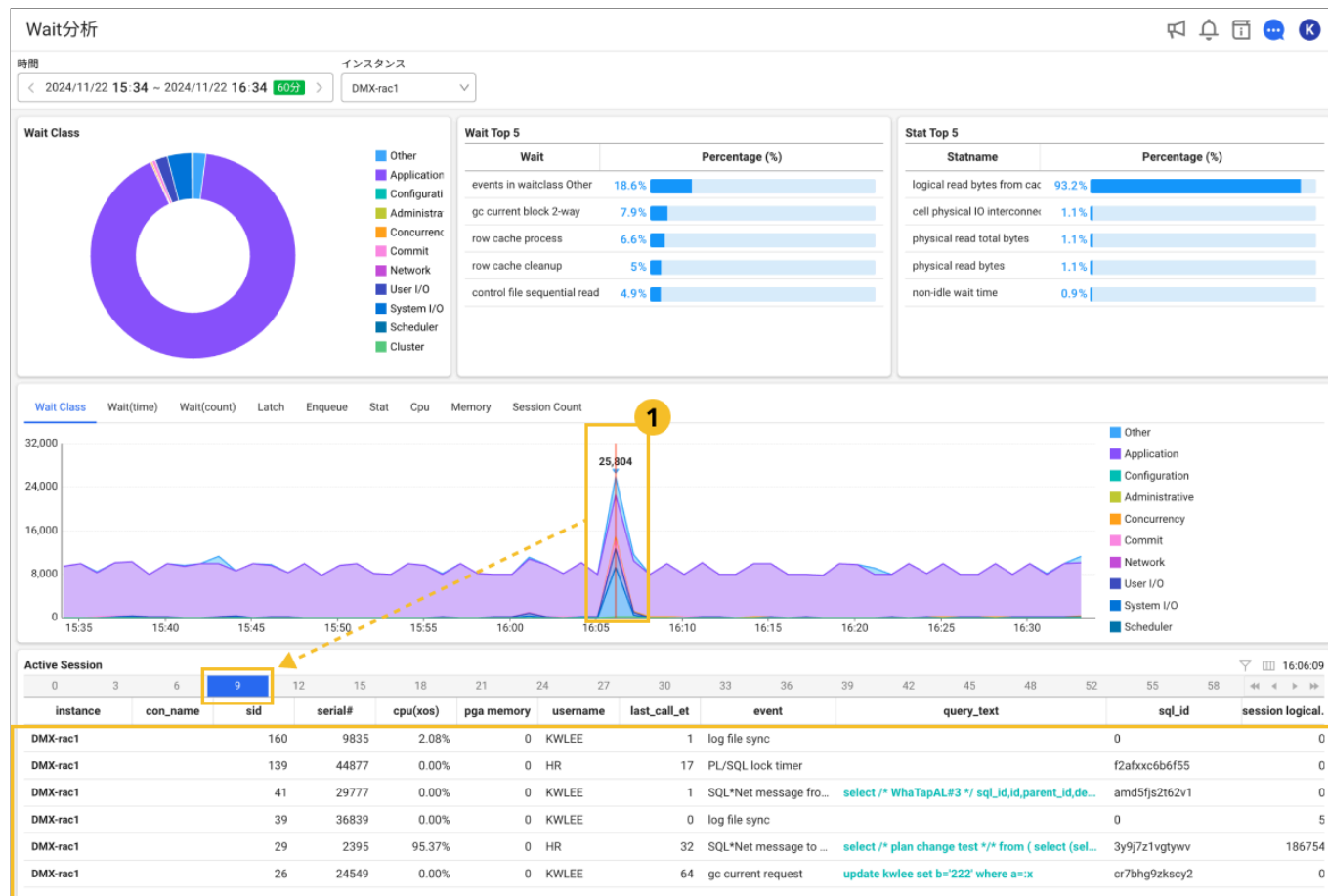
Wait分析

ホーム画面 > プロジェクト選択 > 分析 > Wait分析

遂行中のセッションが何の待機イベントで待機しているのか、頻繁に発生する待機イベントはどのようなのかを確認することができます。5秒ごとに収集したアクティブセッションを待機(Wait)の観点から分析することができます。

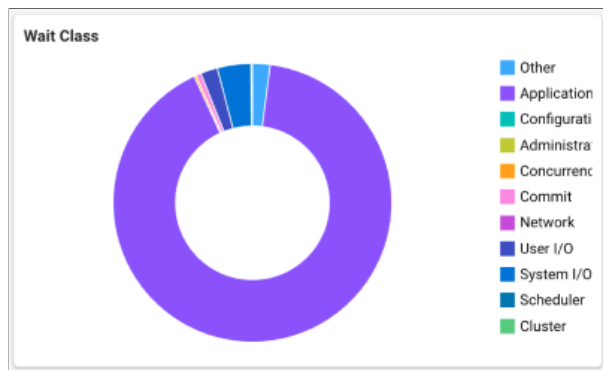
- ❗ • この機能は、DBXエージェント1.9.0バージョン以降に対応しています。
- 収集された wait および stat 情報は、`v$system_wait_class`、`v$system_event`、`v$sysstat` ビューに基づいています。
- 各指標の詳細な説明については、[次の文書](#)を参照してください。

照会する**時間**と**インスタンス**を選択します。設定した時間とインスタンスに該当する待機情報を確認することができます。照会期間中に待機(遅延)が頻繁に発生したアクティブセッションがあったら、どのようなタイプの待機イベントが発生したのかを分析するのに活用できます。



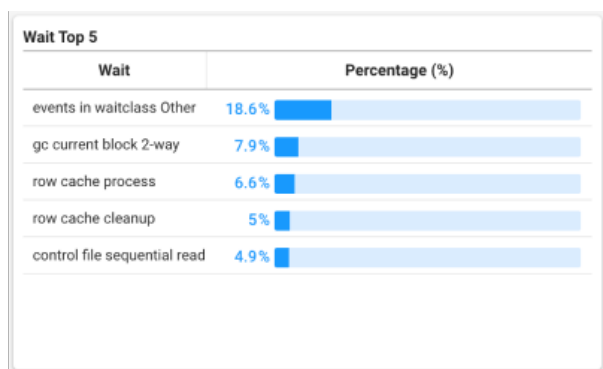
真ん中の時系列チャートでは、時間帯別Wait Classの分布及びセッション数を視覚的に表示します。これにより、特定の時間帯の待機イベントの増加パターンを確認できます。Wait(time)、Wait(count)、Latch、Enqueue、Stat、Cpu、Memory、Session Countなどの分析カテゴリを選択できます。

時系列チャートで①詳細時間を選択すると、その時間のアクティブセッションを照会でき、Waitイベントが発生したSQL文を確認できます。



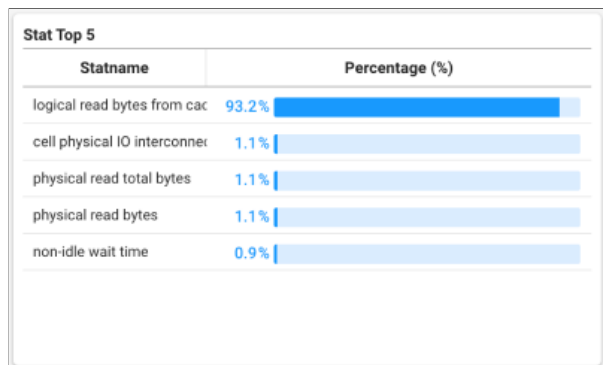
Wait Class

円形チャートを通じて待機イベントをWait Class別に分類して視覚化します。これにより、主要な待機イベントの原因を素早く特定できます。Wait Classが異常に高い場合は、そのClassに関するリソースを集中的にチェックします。



Wait Event Top 5

発生頻度の高い待機イベントを上位5つまで表示します。各待機イベントの割合(%)を通じて、どのイベントがシステムの性能に最も大きな影響を及ぼすかを把握でき、システムのボトルネック現象が何かを確認するのに役立ちます。



Stat Top 5

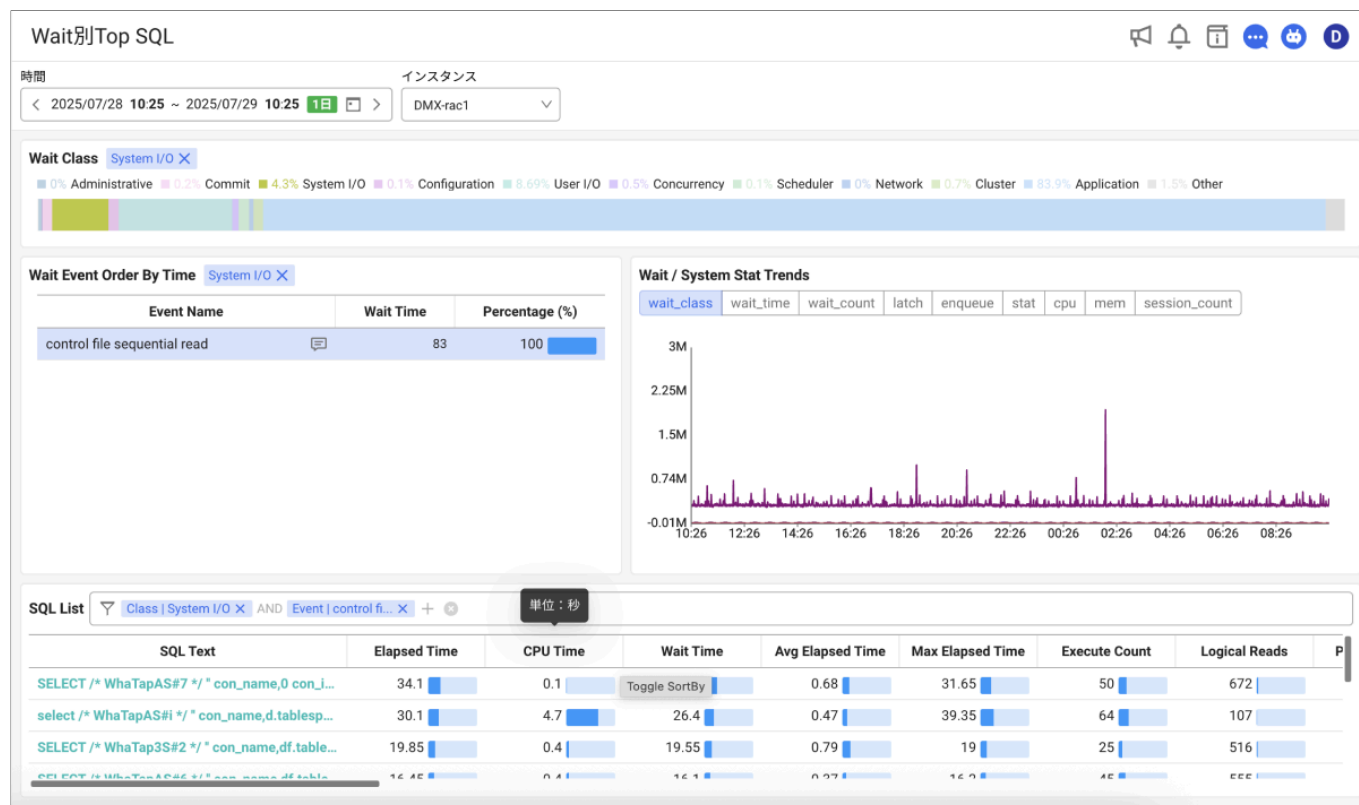
主要Statデータのうち上位5つを割合(%)に表示します。データベースで最も影響を及ぼすタスクやリソースの使用状況を素早く理解できます。

- ✔
 • Wait分析結果は原因診断を助けるツールとして活用され、実際の問題解決はDBAの判断によって実行されます。
 - 時間範囲を適切に調整して、問題が発生した時間帯を正確に分析する必要があります。
 - Wait Class及びイベントの特性を事前に熟知すれば、分析速度が速くなります。
-
- !
 • **Active Session**セクションの一覧で**instance**カラム項目を選択すると、**インスタンスモニタリング**メニューに移動します。 **インスタンスモニタリング**メニューの詳細は[次の文書](#)を参照してください。
 - **Active Session**セクションの一覧でセッション項目を選択すると、**セッション**詳細画面が表示されます。
 - **Active Session**セクション一覧で**query_text**カラム項目を選択すると**SQL**詳細画面が表示されます。
 - テーブルヘッダーカラムを非表示にしたり、希望する項目を追加できます。カラムの順序を変更することもできます。
 - テーブルデータをフィルタリングして、目的のアイテムをすばやく照会できます。

Wait別 Top SQL

Wait別 Top SQLは、チューニングの優先順位を迅速に把握することに重点を置いた分析機能です。特定の時間帯に最も多くの待機時間が発生した Wait Class と Wait Event をチャートとテーブルで一目で確認でき、Top SQL テーブルでは実行回数と平均時間を比較できます。

基本画面案内



Wait Class チャート

チャートまたは凡例のクラスをクリックすると、そのクラスがフィルターとして適用されます。もう一度クリックするとフィルターが解除されます。

- チャート上部の Wait Class 棒グラフは、選択された分析期間中の全体待機時間の比率を視覚的に表示します。
- 選択されたフィルターはタグ形式で表示されます。

Wait Event リスト

クラスが選択されていない場合、指定した時間範囲内で最も頻繁に発生したイベントが基準となって表示されます。Wait Class フィルターが適用されると、そのクラスに該当するイベント一覧が表示されます。

Wait Event リストで **Wait Event** を選択すると、**SQL List** フィルターにその内容が追加されます。

Wait / System Stat 傾向

時間帯別の Wait Class の分布およびセッション数を視覚的に表示します。特定の時間帯における待機イベントの増加パターンを確認できます。分析カテゴリには Wait(time)、Wait(count)、Latch、Enqueue、Stat、Cpu、Memory、Session Count があります。

SQL List

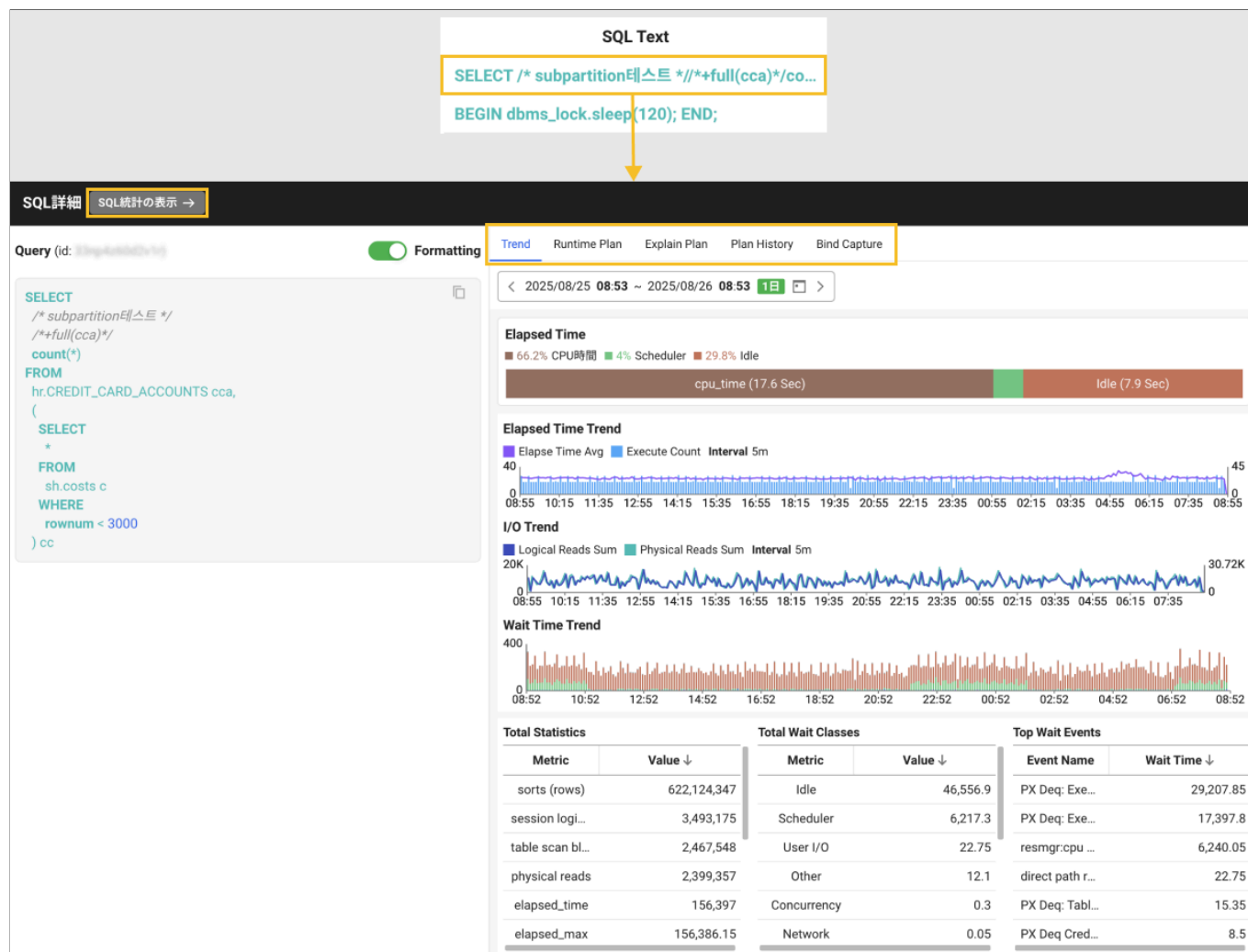
各 SQL 項目について、Elapsed Time、CPU Time、Wait Time、Execute Count、Logical Reads などの主要な実行パフォーマンス指標を表示し、迅速な比較分析が可能です。

- デフォルト（フィルター未選択）: Elapsed Time が最も高い上位50件のクエリを表示
- クラス選択時: 選択したクラスに対して Wait Time が最も高い上位50件のクエリを表示

詳細クエリの表示

取得した結果リストで **query** 列を選択すると **SQL詳細** ウィンドウが表示されます。

SQL統計の表示 → ボタンをクリックすると、該当SQLクエリに関連する統計情報を確認できる **SQL統計** メニューに移動します。



SQL詳細 ウィンドウのタブは次のとおりです。

- **Trend:** 特定SQLの実行時間、I/O、待機イベントを総合的に分析し、パフォーマンスボトルネックを素早く把握できます。
- **Runtime Plan:** 選択したSQLクエリの実行計画とランタイム情報を提供します。実行回数、平均実行時間、平均物理読み取り数などの詳細情報を確認できます。
- **Explain Plan:** オプティマイザが予測した実行計画を表示します。コスト、処理内容、オブジェクト名、カーディナリティなどの情報を提供します。
- **Plan History:** データベースで実行されたSQLクエリの実行計画履歴を確認できます。
- **Bind Capture:** データベースで実行されたSQLクエリに使用されたバインド変数の値を確認できます。これにより、実際に実行されたクエリの内容を把握できます。

- ① リアルタイムのバインド値ではなく、データベースにキャプチャされた値（`v$sql_bind_capture`）です。件数が多い場合は最大100件まで表示されます。

Trend タブ

上部チャートで時間帯ごとの Elapsed Time、I/O、Wait Time の推移を可視化します。

下部統計テーブルでは累積実行統計、Wait Class比率、主要Wait Eventを一目で比較できます。

Elapsed Time

タイムセレクトで選択された区間の **CPU使用時間**、**Wait Class所要時間**、**Idle時間** を合算して表示します。

例:

- CPU Time: SQL実行で実際にCPUを使用した時間
- Scheduler: セッションがCPUを待機していた時間
- Idle: 実行されず待機またはアイドル状態の時間（どの項目に最も時間が費やされたか直感的に確認可能）

Elapsed Time Trend

時間帯ごとの **平均Elapsed Time** と実行回数（Execution Count）を5分単位で集計し、推移を表示します。
3時間未満の場合は1分単位で集計されます。

特定の時点でElapsed Timeが急増したか、実行回数に比べ応答時間が増加したかを確認でき、性能低下の発生時点を素早く把握するのに役立ちます。

I/O Trend

クエリ実行中に発生した **Logical Reads** と **Physical Reads** を5分単位で表示します。
3時間未満の場合は1分単位で表示します。

- Logical Reads: バッファキャッシュからデータを読み取った回数
- Physical Reads: ディスクから直接データを読み取った回数。キャッシュ効率やI/O負荷を比較してボトルネックを把握可能

Wait Time Trend

クエリ実行中に発生した **Wait Event** **所要時間** を時間帯別に表示します。

特定の時点でWait Timeが急上昇した場合、DBリソース競合やI/O遅延が発生していることを意味します。どのイベント（ロック、I/O、並列処理など）がボトルネックを引き起こしたのかを追跡できます。

Total Statistics

選択区間中の **SQL実行累積統計** を表示します。

例:

- sorts(rows): ソートされた総行数
 - session logical reads: セッション論理読み取り回数
 - table scan blocks gotten: テーブルスキャン時に読み取ったブロック数
 - elapsed_time: 総所要時間
- SQL実行がリソースをどのように使用したか集計データを確認できます。

Total Wait Classes

SQL実行中に発生した **待機時間(Wait Time)** をWait Class単位で合算した値です。

Wait Classごとの所要時間が多いほど、その領域がボトルネックである可能性が高いです。

例: Idle, Scheduler, User I/O, Concurrency など

Top Wait Events

選択区間で発生した **詳細Wait Event**の上位リストを表示します。

Wait Classよりも詳細に、どのイベントがパフォーマンスに最も大きな影響を与えたかを確認できます。

例: PX Deq: Execution Msg, resmgr:cpu quantum, direct path read など

キューブ

キューブメニューは、長時間にわたる負荷の分布をヒットマップ1枚で表示し、そのうち関心のある区間を選んで詳細な指標へ掘り下げられる画面です。Oracle Pro製品でのみ提供します。

パフォーマンス問題を追跡するには、まず問題が発生した時点を見つける必要があります。しかし、指標チャートの照会範囲を1日単位で広く取ると、短時間だけ跳ね上がった区間が平坦になって目立たなくなり、逆に範囲を狭めるとどの区間を見るべきか分かりません。キューブは照会期間全体を5分または1日単位のセルに分割して色の濃淡で表示するため、負荷が集中した区間を色で先に見つけ、その区間のみを詳細に分析できます。

この画面では次の作業ができます。

- 指標を1つ選び、照会期間全体の分布をヒットマップで確認
- ヒットマップで負荷が高いセルを選択し、該当区間のOS・セッション・統計・待機指標を確認
- 同じ区間のTop SQLを続けて確認

ヒットマップを照会する

画面左側の領域でヒットマップの照会条件を指定します。

1. **開始日**で照会を開始する日付を選択してください。
2. **照会期間**でヒットマップの単位を選択してください。****日（5分）****を選択すると1日を5分間隔のセルで表示し、****月（日別）****を選択すると1か月を1日間隔のセルで表示します。
3. **集計**で、1つのセルに複数の値が入る場合に適用する方式を選択してください。**平均**または**最大**を選択できます。
4. **メトリクス**でヒットマップに表示する指標を選択してください。
5. 画面右上の**インスタンスセクター**で照会対象のインスタンスを選択してください。

指標

`db_oracle_dma_counter` カテゴリの指標はすべて選択できます。主に使用する指標は次のとおりです。

Item	Metric	Description
CPU	cpu(xos)	XOSエージェントが収集したサーバーCPU使用率
Memory	mem(xos)	XOSエージェントが収集したサーバーメモリ使用率
Active Sessions	active_sessions	アクティブセッション数
DB Time	DB time	データベースがリクエストの処理に使った時間
Logical I/O	session logical reads	セッションの論理読み取り回数
Physical I/O	physical reads	ディスクから直接読み取った回数
SQL Executions	execute count	SQL実行回数
Lock Wait	lock_wait_sessions	ロックを待機しているセッション数

詳細区間を分析する

ヒットマップでセルを選択すると、画面右側に該当区間の詳細指標が表示されます。**日（5分）**を基準に照会した場合、選択したセルの5分区間が分析対象になります。

詳細領域のタブは次のとおりです。

Item	Description
OS	XOSエージェントが収集したCPU・メモリ・ディスクI/O・ネットワークI/O・CPU I/O Waitの推移
Session	アクティブセッション、コネクション、DB Time、ロック待機セッション、長時間実行セッション、Enqueue待機の推移
Stat	SQL実行回数、トランザクション（コミット・ロールバック）、論理I/O、物理I/O、ブロック変更、Redoの推移
Wait	選択区間の待機イベント分布とLatch・Enqueue指標

Item	Description
Top-N	選択区間のTop SQL一覧

Waitタブと**Top-N**タブは、それぞれ**Wait分析**、**Top SQL**画面と同じウィジェットを使用します。2つの画面についての詳細は、[Wait分析](#)、[Top SQL](#)の文書を参照してください。

Top-Nタブには**Top SQL**画面のオプションバーと分析基準タブがそのまま含まれています。キューブで選択した区間が照会時間として適用されます。

Sessionタブでは、選択した区間のアクティブセッション一覧とプロセス情報を確認できます。過去の区間を照会する画面のため、セッション一覧はリアルタイムに更新されません。アクティブセッションとプロセス情報は、それぞれ下位タブで確認できます。

ロックツリー

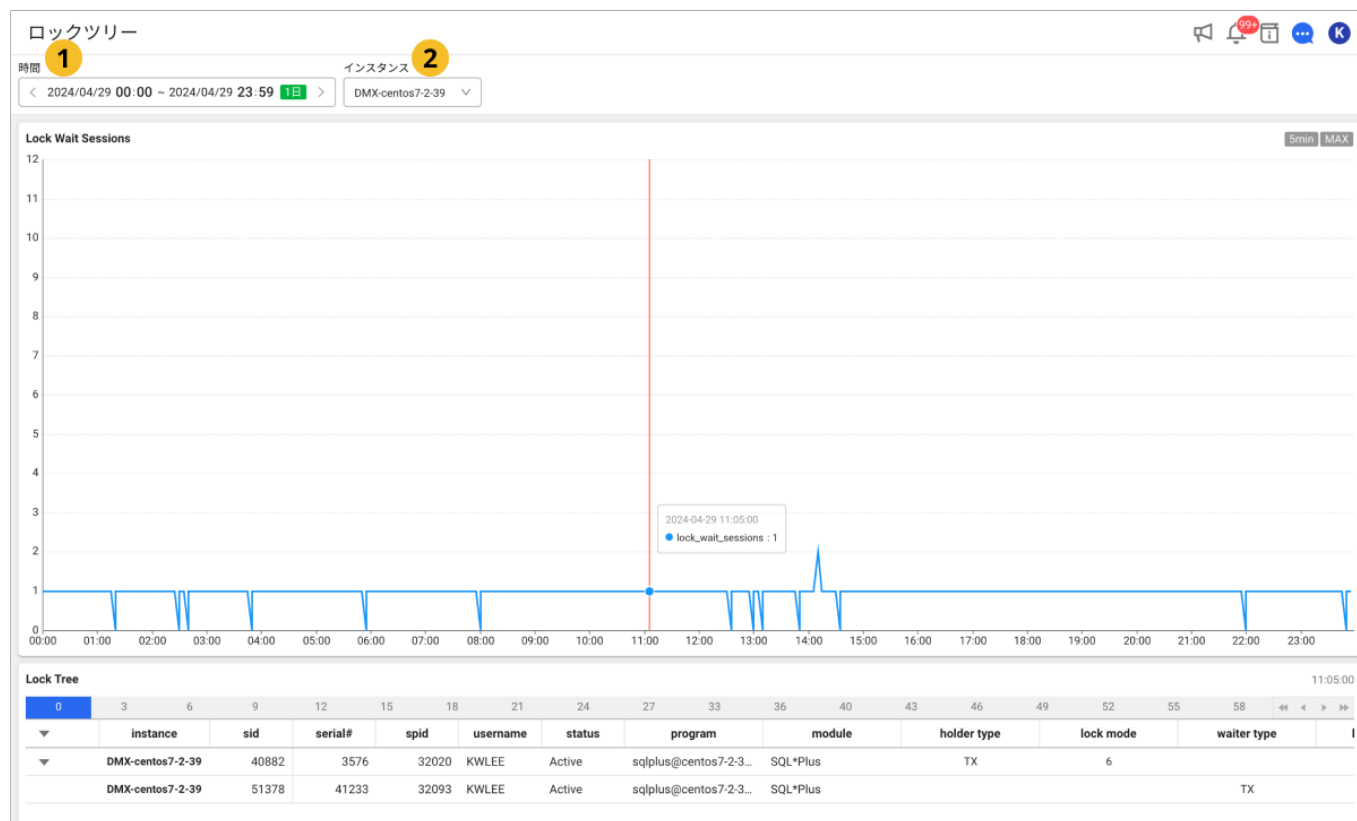
ホーム画面 > プロジェクト選択 > 分析 > ロックツリー

一日の間に発生したロック(lock)の推移を分析するツールです。この機能によってロックを発生させたセッション(lock holder)と待機しているセッション(lock waiter)に対するツリー構造を視覚的に確認できます。特定時点で発生したロック情報のholderとwaiter間の関係を分析できます。WhaTapの**ロックツリー**機能により、データベース管理者はロック関連の問題を迅速に識別して解決でき、データベースの性能を最適化できます。

主な機能は次のとおりです。

- **ロック推移分析:** 設定した照会期間に発生したロックの推移を時間とともに追跡し、グラフで表示します。これにより、ロックがどのように発生し、解消されるかを視覚的に把握できます。
- **ロックHolderおよびWaiter確認:** 各ロックに対するholderとwaiterをツリー構造で確認できます。これにより、特定のロックが発生した状況でロックを発生させたセッションと待機するセッションとの関係を把握できます。
- **セッション間の関係分析:** ロックを発生させたセッションと待機するセッションとの関係を分析し、クエリー実行およびトランザクション処理過程で発生したロックの原因を識別できます。
- **時間別ロックトレンドの視覚化:** 1-Min Lock TrendsとLock Wait Sessions機能により、ロック発生データを時間単位で分析できます。

基本的な使い方



目的の①日付と時間を設定し、数字②の一覧からインスタンスを選択します。設定した時間にロックが発生した場合は、[lock wait sessions](#)と画面下のテーブルに表示されているデータを表示します。

Lock Wait Sessions

[Lock Wait Sessions](#)は、ロック待機セッションの情報を分析できる機能です。照会期間によってロック待機セッションデータを3時間以上照会すると5分間隔で、3時間未満照会すると5秒間隔で提供します。また、チャートをドラッグして時間範囲を流動的に調整できます。

1-Min Lock Trends

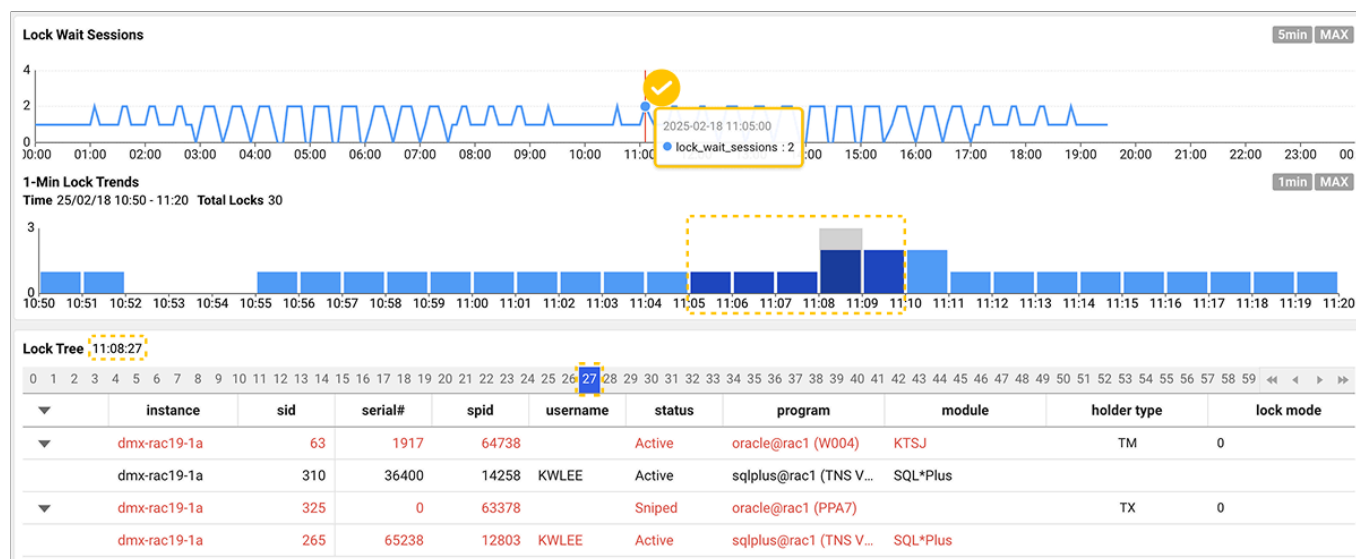
[Lock Wait Sessions](#)で特定の時間帯をクリックすると、その時間基準で15分前から15分後までの計30分間のロック発

生状況を可視化して表示します。これにより、30分単位でロック発生トレンドを効果的に分析できます。

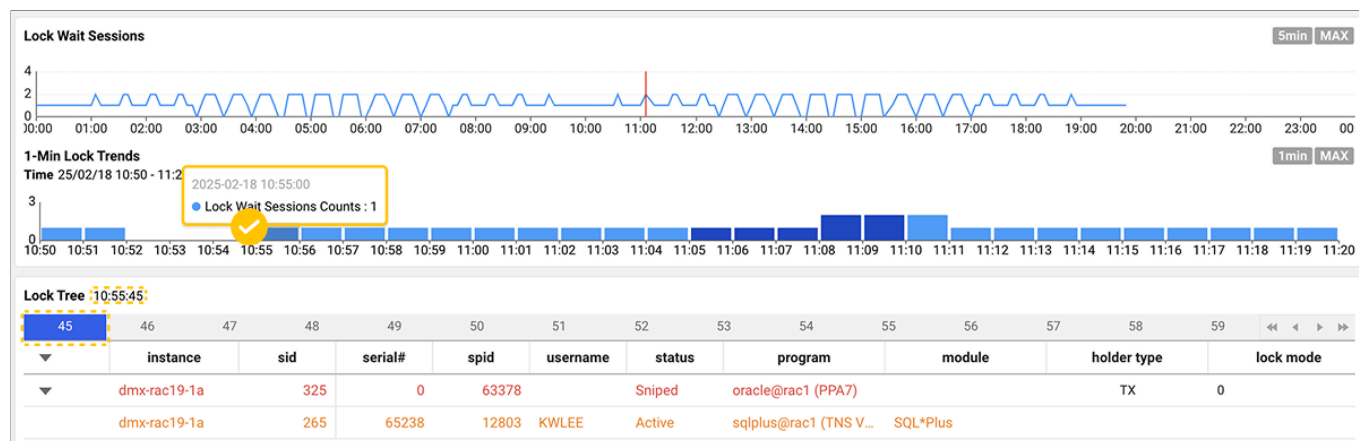
❗ **Total Locks**は、その時間帯に発生したロックの合計個数を意味します。

時間照会

Lock Wait Sessionsでクリックした時間に発生した最大ロック数を基準に**1-Min Lock Trends**と**Lock Tree**テーブルの時間が設定されます。これにより、ロックが最も多く発生した区間を把握できます。



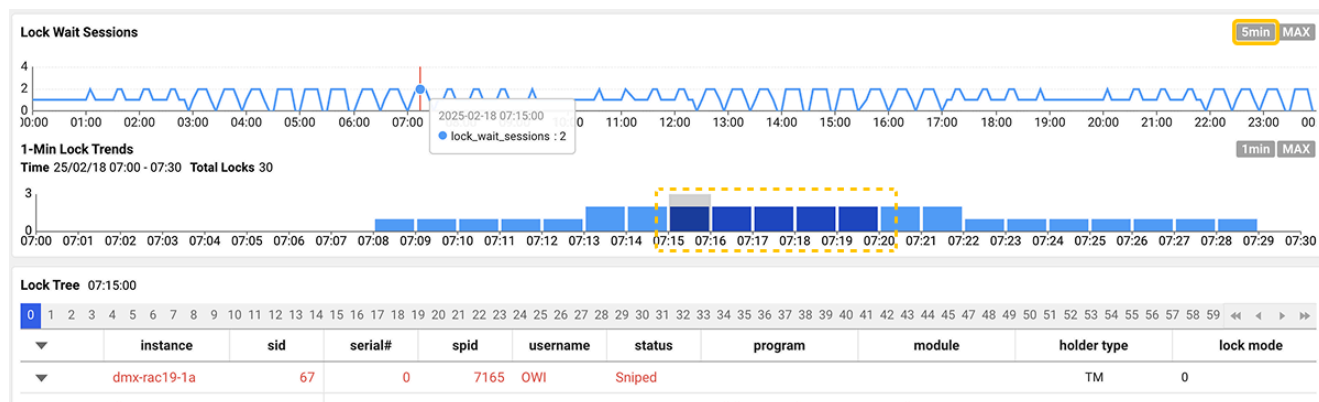
Lock Wait Sessionsはそのままにして**1-Min Lock Trends**チャートで他の時間帯のバーをクリックした場合、該当する分の最大ロック数を持つ秒の時点で**Lock Tree**テーブルの時間範囲が同期されます。これにより、時間帯を変更する時に**Lock Tree**テーブルに反映された情報を確認できます。



・3時間以上照会した場合 (5分単位 of データ使用)

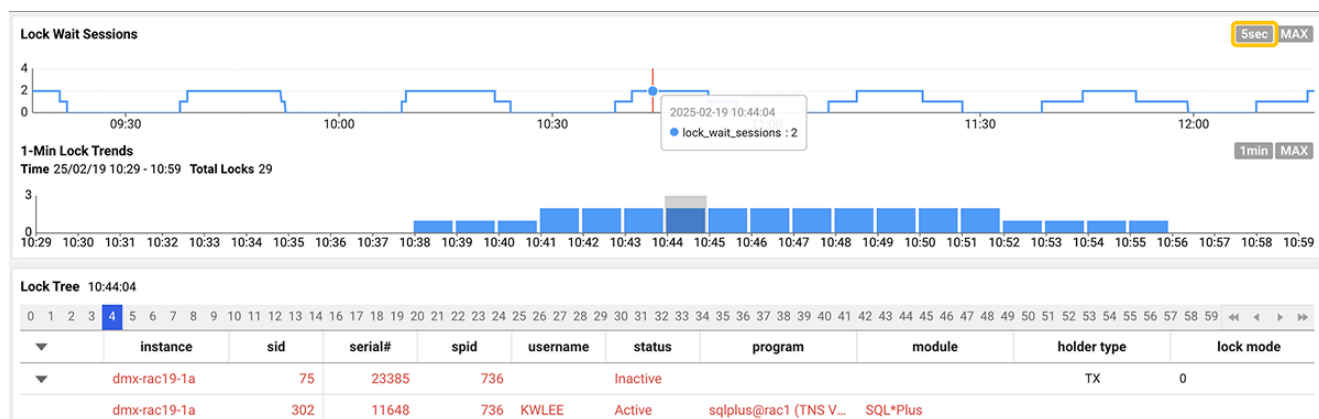
照会時点から5分の区間でバーの色が濃く表示されます。

07:15:00 時点をクリックした場合、07:15:00~07:20:00 区間バーのチャートの色が濃く表示されます。



・3時間未満照会した場合 (5秒単位 of データ使用)

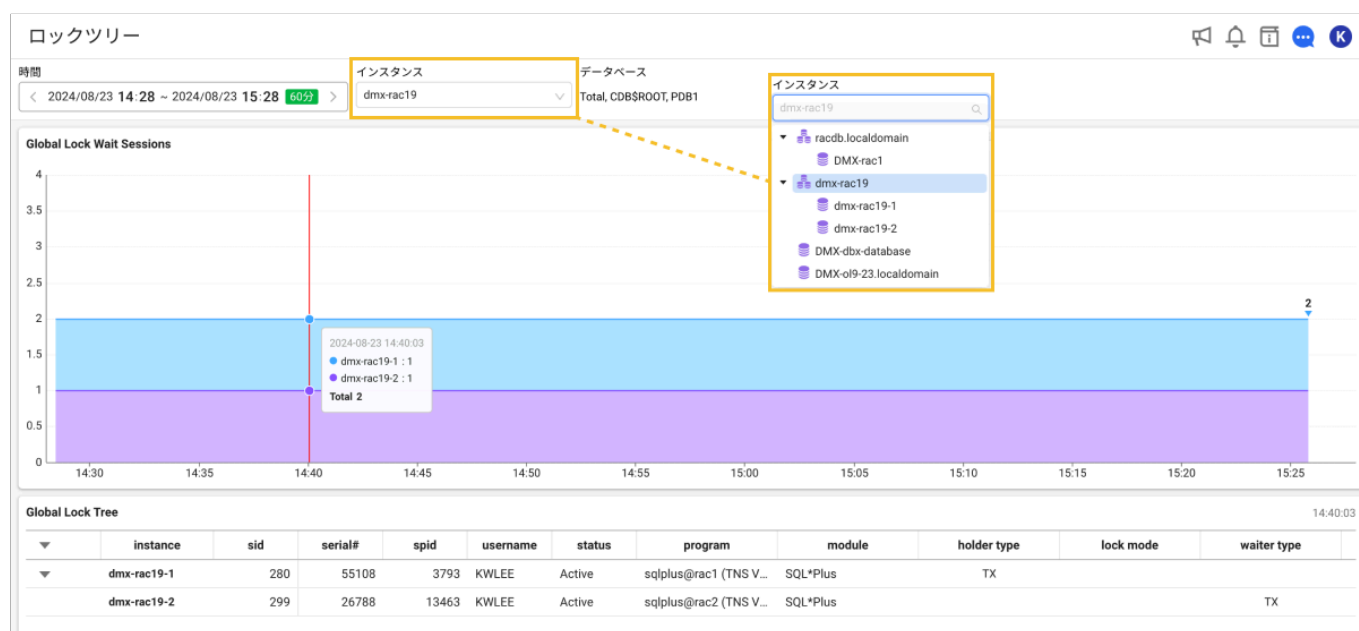
短い時間間隔で発生したロック状況を分析する時に役立ちます。



Global Lock情報の確認

クラスターで構成されたOracleデータベース環境で、Global Lock情報の確認は、いろいろメリットがあります。特に、Oracle Real Application Clusters (RAC) 環境で、このような情報は性能の最適化と問題の解決に不可欠です。

クラスターで構成された環境であれば、**インスタンス**からクラスターを選択できます。



• Global Lock Wait Sessions

選択した時間範囲で発生したGlobal Lockの待機セッション数を時間帯ごとに可視化して表示します。特定の時間

帯に発生したロックの待機数を一目で把握できます。各色は異なるインスタンスを表し、ロックが発生したインスタンスごとに区分されて表示します。

• Global Lock Tree

ロックに関する詳細情報をツリー構造で確認できます。ロックが発生したインスタンス、セッションID(**sid**)、セッション状態(**status**) など重要な情報が含まれます。ロックのホルダータイプ(**holder type**)とロックモード(**lock mode**)、そしてロックの待機タイプ(**waiter type**)を通じて、ロックの原因と影響を迅速に把握できます。

ツリー構造により、どのセッションがロックを掛けており、どのセッションが待機中なのかを把握できます。

- ❗ • クラスターを選択し、照会できる最大の時間範囲は3時間です。
- Global Lock Treeセクション一覧で項目を選択すると**セッション詳細**が確認できます。

- ✔ • RAC環境では、複数のインスタンスが同じデータに同時にアクセスできます。Global Lock情報は、このような同時アクセス状況でデータの一貫性を確保するための主要なメカニズムの1つです。これをモニタリングすることで、データ一貫性の維持有無を確認できます。
- Global Lock待ち時間は、システムの性能に大きな影響を与える可能性があります。クラスターのインスタンス間で発生するGlobal Lockの競合をモニタリングすると、性能低下の原因となるボトルネックを早期に検知できます。
- Global Lock情報は、トランザクションがクラスターでどのように処理されるかに対する重要なデータを提供します。同時実行制御が適切に行われないと、トランザクションの待ち時間が長くなったり、デッドロックが発生する可能性があります。
- RAC環境では、インスタンス間の負荷分散が非常に重要です。Global Lock情報を通じて、どのインスタンスがより多く競合しているのかを把握でき、この情報で負荷を再分配したり、リソース管理を改善できます。

カラム情報の案内

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

カラム情報に関する詳しい内容は、[次のリンク](#)を参照してください。

項目	説明
username	DBユーザー名
command	COMMAND_TYPE と同一、SQLのcommand number
status	セッションの状態
schemaname	スキーマ、usernameと同一
osuser	セッションに接続したOSのユーザー名
spid	OSプロセスの識別子
process	セッションに接続したクライアントプロセスID
logon_time	セッションが作成された時点
last_call_et	セッションがアクティブになっている場合、アクティブになってから経過した時間(秒)
sql_text	query、実行されているstatement、リテラル文字列を除外したSQL文
sql_param	query から分離したリテラル値
prev_sql_text	以前に実行したquery
prev_sql_param	以前に実行したquery param
program	クライアントで実行したプログラムの情報
type	session type - user background recursive (whatap.conf)で設定可能
seq#	現在または最後の待機を固有に識別する番号(待機するたびに増加)
event	セッションがキューに登録されている場合は、セッションが待機しているリソースまたはイベントの数

項目	説明
p1	最初の待機イベントのパラメータ(小数点)
p2	2番目の待機イベントのパラメータ(小数点)
p3	3番目の待機イベントのパラメータ(小数点)
wait_class	待機中のイベントのクラス名
wait_time	セッションが現在待機中の場合、値は0です。0より大きい場合は、最後の待機時間(milliseconds)
seconds_in_wait	セッションが現在待機中の場合、セッションが待機のために待機した時間を意味します。セッションが待機中でない場合、最後の待機開始からの時間を意味します。
state	待機状態、WAITING, WAITED UNKNOWN TIME, WAITED SHORT TIME, WAITED KNOWN TIME
machine	セッションに接続したクライアントサーバー名(hostname)
port	クライアントポート番号
terminal	端末名
module	クライアントで実行したプログラムでユーザーが設定したモジュール(module)名
アクション	現在実行されているモジュールのユーザーが設定したアクション名
client_info	ユーザーが設定したクライアント情報
client_identifier	ユーザーが設定したクライアントの識別子
session logical reads	バッファキャッシュ内でデータを読み込んだ回数
physical reads	物理ディスクからデータを読み出した回数
excute count	SQLの実行数

項目	説明
parse count (hard)	構文分析の呼び出し合計数(実際の構文分析)
parse count (total)	構文分析呼び出しの総数(hard、soft、describe)
opened cursors current	現在開いているcursorの数
db block changes	データベース内のすべてのブロックの変更回数
session pga memory	セッションの現在のPGAサイズ(size)
undo_segid	undo情報id
undo_blk	undoブロックの数
undo_rec	undoレコードの数
sql_address	SQL文(statement)を識別する情報
sql_hash_value	SQL文(statement)を識別する情報
sql_id	現在実行中のSQL文(statement)のSQL識別子
sql_child_number	現在実行中のSQL文(statement)の下位番号
sql_exec_start	当該セッションで現在実行中のSQL実行は、開始点にあります。
sql_exec_id	SQL実行識別子
prev_sql_addr	最後に実行されたSQL文(statement)を識別する情報
prev_child_number	最後に実行されたSQL文(statement)の下位番号
prev_exec_start	最後に実行されたSQL文(statement)のSQL実行の開始時点
prev_exec_id	最後に実行されたSQL文(statement)のSQL実行識別id

項目	説明
plsql_entry_object_id	スタックの一番上にあるPL/SQL下位プログラムのオブジェクトid
plsql_entry_subprogram_id	スタックで一番上にあるPL/SQL下位プログラムのサブプログラムid
plsql_object_id	現在実行中のPL/SQLオブジェクトのサブプログラムid
taddr	トランザクションのアドレス(address)
lockwait	lock wait address、セッションが待機中のロックアドレス
row_wait_obj	ROW_WAIT_ROW# で指定された行を含むテーブルのオブジェクトID
row_wait_file	ROW_WAIT_ROW# で指定された行を含むデータファイルの識別子
row_wait_block	ROW_WAIT_ROW# で指定された行を含むブロックの識別子
row_wait_row	現在ロックされている行
pdml_status	値が ENABLED の場合、セッションが PARALLEL DML 有効モードです。値が DISABLE の場合、セッションで PARALLEL DML 有効モードに対応していません。値が FORCED の場合、セッションが PARALLEL DML を強制的に使用するように変更されたものです。
pq_status	値が ENABLED の場合、セッションが PARALLEL QUERY 有効モードです。値が DISABLE の場合、セッションで PARALLEL QUERY 有効モードに対応していません。値が FORCED の場合、セッションが PARALLEL QUERY を強制的に使用するように変更されたものです。
blocking_session_status	ブロックしているセッションの存在有無に関する詳細を提供
blocking_instance	ブロックしているセッションのインスタンス識別子
blocking_session	ブロックしているセッションのセッション識別子
final_blocking_session_status	最終ブロックセッションの存在有無に関する詳細を提供

項目	説明
final_blocking_instance	最終ブロックしているセッションのインスタンス識別子
final_blocking_session	最終ブロックしているセッションのセッション識別子
service_name	セッションのサービス名やDB名と類似
saddr	セッションアドレス
con_id	データに関連するコンテナのid(マルチデータベースをサポートする場合のみ)

ⓘ WhaTapは、クライアントに関する情報をデフォルトで保存します。

項目	説明
Instance	インスタンス（エージェント）名
sid	セッションID
serial#	セッションの一意的識別子と共に使用するセッションシリアル番号
holder_type	ロックを保持しているセッションのロックタイプ（例：TX、TMなど）
holder_mode	保持中のロックモード（例：0～6、6はexclusive）
waiter_type	ロックを待機しているセッションのロックタイプ
waiter_mode	現在要求中のロックモード
id1	ロック識別子1（ロック対象オブジェクトを識別するために使用、例：トランザクションIDなど）
id2	ロックの詳細情報またはバージョン情報を表します

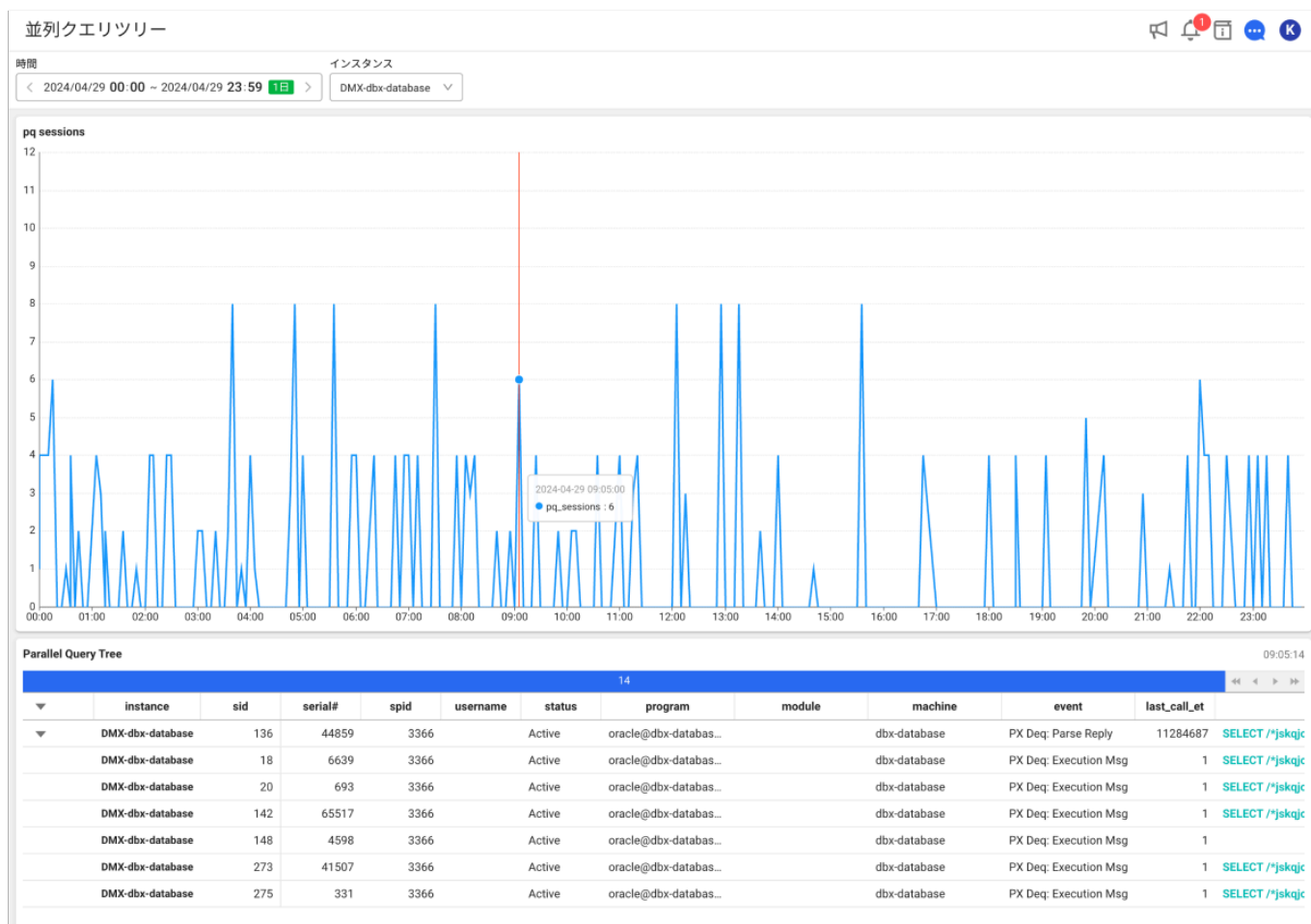
次の項目は、XOSエージェントをインストールした場合に収集される指標です。

項目	説明	単位
pid	プロセスID	-
cpuusage	プロセスが使用するCPU使用率	%
cputime	プロセスがこれまでに使用した累積CPU時間	秒
elapsed	プロセスが実行されてからの経過時間	秒
vsize	Virtual memory size. 全体の仮想メモリ使用量 (RAM + Swap + 予約)	KB
rss	Resident Set Size. 現在のプロセスが占有している実際の物理メモリサイズ	KB
pss	Proportional Set Size. 共有メモリを含む実際の物理メモリ使用量	KB
state	プロセスの現在の状態コード (例: R、S、D、T、Z など)	-
ioread	累積読み込みバイト数 (read_bytes)	バイト (B)
iowrite	累積書き込みバイト数 (write_bytes)	バイト (B)
uid	このプロセスを実行したユーザーID	-
cmd	実行されたコマンドの短縮形式	-
longcmd	実行されたコマンドの完全な形式 (フルパスおよび引数を含む)	-

並列クエリツリー

並列クエリツリー(Parallel Query Tree)は、1日の間にParallel Queryが発生した傾向を分析できる機能を提供します。実際のタスクを実行するSlave Processと、それらにタスクを割り当てたCoordinator Processとを区別して表示し、全体の実行情報をわかりやすく表示します。特定の時点で実行したParallel Query情報を提供し、どのような目的で使われているかを確認するのに役立ちます。

基本画面ガイド



時間項目で照会する時間を設定した後インスタンス項目で並列クエリ情報を照会するインスタンスを選択します。

グラフチャートで設定した条件に合ったパラレルクエリ情報の推移を確認することができます。CoordinatorとSlaveの構造をツリー形式で表現して提供します。

- ❗ **Parallel Query Tree**セクションのテーブル一覧で**query**のカラム項目を選択すると**SQL詳細**画面が表示されます。

カラム情報の案内

- Active session** **Lock tree** **Process information**

カラム情報に関する詳しい内容は、[次のリンク](#)を参照してください。

項目	説明
username	DBユーザー名
command	COMMAND_TYPE と同一、SQLのcommand number
status	セッションの状態
schemaname	スキーマ、usernameと同一
osuser	セッションに接続したOSのユーザー名
spid	OSプロセスの識別子
process	セッションに接続したクライアントプロセスID
logon_time	セッションが作成された時点
last_call_et	セッションがアクティブになっている場合、アクティブになってから経過した時間(秒)
sql_text	query、実行されているstatement、リテラル文字列を除外したSQL文

項目	説明
sql_param	query から分離したリテラル値
prev_sql_text	以前に実行したquery
prev_sql_param	以前に実行したquery param
program	クライアントで実行したプログラムの情報
type	session type - user background recursive (whatap.conf)で設定可能
seq#	現在または最後の待機を固有に識別する番号(待機するたびに増加)
event	セッションがキューに登録されている場合は、セッションが待機しているリソースまたはイベントの数
p1	最初の待機イベントのパラメータ(小数点)
p2	2番目の待機イベントのパラメータ(小数点)
p3	3番目の待機イベントのパラメータ(小数点)
wait_class	待機中のイベントのクラス名
wait_time	セッションが現在待機中の場合、値は0です。0より大きい場合は、最後の待機時間(milliseconds)
seconds_in_wait	セッションが現在待機中の場合、セッションが待機のために待機した時間を意味します。セッションが待機中でない場合、最後の待機開始からの時間を意味します。
state	待機状態、WAITING, WAITED UNKNOWN TIME, WAITED SHORT TIME, WAITED KNOWN TIME
machine	セッションに接続したクライアントサーバー名(hostname)
port	クライアントポート番号

項目	説明
terminal	端末名
module	クライアントで実行したプログラムでユーザが設定したモジュール(module)名
アクション	現在実行されているモジュールのユーザが設定したアクション名
client_info	ユーザが設定したクライアント情報
client_identifier	ユーザが設定したクライアントの識別子
session logical reads	バッファキャッシュ内でデータを読み込んだ回数
physical reads	物理ディスクからデータを読み出した回数
excute count	SQLの実行数
parse count (hard)	構文分析の呼び出し合計数(実際の構文分析)
parse count (total)	構文分析呼び出しの総数(hard、soft、describe)
opened cursors current	現在開いているcursorの数
db block changes	データベース内のすべてのブロックの変更回数
session pga memory	セッションの現在のPGAサイズ(size)
undo_segid	undo情報id
undo_blk	undoブロックの数
undo_rec	undoレコードの数
sql_address	SQL文(statement)を識別する情報
sql_hash_value	SQL文(statement)を識別する情報

項目	説明
sql_id	現在実行中のSQL文(statement)のSQL識別子
sql_child_number	現在実行中のSQL文(statement)の下位番号
sql_exec_start	当該セッションで現在実行中のSQL実行は、開始点にあります。
sql_exec_id	SQL実行識別子
prev_sql_addr	最後に実行されたSQL文(statement)を識別する情報
prev_child_number	最後に実行されたSQL文(statement)の下位番号
prev_exec_start	最後に実行されたSQL文(statement)のSQL実行の開始時点
prev_exec_id	最後に実行されたSQL文(statement)のSQL実行識別id
plsql_entry_object_id	スタックの一番上にあるPL/SQL下位プログラムのオブジェクトid
plsql_entry_subprogram_id	スタックで一番上にあるPL/SQL下位プログラムのサブプログラムid
plsql_object_id	現在実行中のPL/SQLオブジェクトのサブプログラムid
taddr	トランザクションのアドレス(address)
lockwait	lock wait address、セッションが待機中のロックアドレス
row_wait_obj	ROW_WAIT_ROW# で指定された行を含むテーブルのオブジェクトID
row_wait_file	ROW_WAIT_ROW# で指定された行を含むデータファイルの識別子
row_wait_block	ROW_WAIT_ROW# で指定された行を含むブロックの識別子
row_wait_row	現在ロックされている行
pdml_status	値が ENABLED の場合、セッションが PARALLEL DML 有効モードです。値が DISABLE の場合、セッションで PARALLE DML 有効モードに対応していません。値

項目	説明
	が FORCED の場合、セッションが PARALLEL DML を強制的に使用するように変更されたものです。
pq_status	値が ENABLED の場合、セッションが PARALLEL QUERY 有効モードです。値が DISABLE の場合、セッションで PARALLEL QUERY 有効モードに対応していません。値が FORCED の場合、セッションが PARALLEL QUERY を強制的に使用するように変更されたものです。
blocking_session_status	ブロックしているセッションの存在有無に関する詳細を提供
blocking_instance	ブロックしているセッションのインスタンス識別子
blocking_session	ブロックしているセッションのセッション識別子
final_blocking_session_status	最終ブロックセッションの存在有無に関する詳細を提供
final_blocking_instance	最終ブロックしているセッションのインスタンス識別子
final_blocking_session	最終ブロックしているセッションのセッション識別子
service_name	セッションのサービス名やDB名と類似
saddr	セッションアドレス
con_id	データに関連するコンテナのid(マルチデータベースをサポートする場合のみ)

ⓘ WhaTapは、クライアントに関する情報をデフォルトで保存します。

項目	説明
Instance	インスタンス（エージェント）名
sid	セッションID

項目	説明
serial#	セッションの一意的識別子と共に使用するセッションシリアル番号
holder_type	ロックを保持しているセッションのロックタイプ（例：TX、TMなど）
holder_mode	保持中のロックモード（例：0～6、6はexclusive）
waiter_type	ロックを待機しているセッションのロックタイプ
waiter_mode	現在要求中のロックモード
id1	ロック識別子1（ロック対象オブジェクトを識別するために使用、例：トランザクションIDなど）
id2	ロックの詳細情報またはバージョン情報を表します

次の項目は、XOSエージェントをインストールした場合に収集される指標です。

項目	説明	単位
pid	プロセスID	-
cpuusage	プロセスが使用するCPU使用率	%
cputime	プロセスがこれまでに使用した累積CPU時間	秒
elapsed	プロセスが実行されてからの経過時間	秒
vsize	Virtual memory size. 全体の仮想メモリ使用量（RAM + Swap + 予約）	KB
rss	Resident Set Size. 現在のプロセスが占有している実際の物理メモリサイズ	KB
pss	Proportional Set Size. 共有メモリを含む実際の物理メモリ使用量	KB
state	プロセスの現在の状態コード（例：R、S、D、T、Z など）	-
ioread	累積読み込みバイト数（read_bytes）	バイト（B）

項目	説明	単位
iowrite	累積書き込みバイト数（write_bytes）	バイト（B）
uid	このプロセスを実行したユーザーID	-
cmd	実行されたコマンドの短縮形式	-
longcmd	実行されたコマンドの完全な形式（フルパスおよび引数を含む）	-

セッション履歴

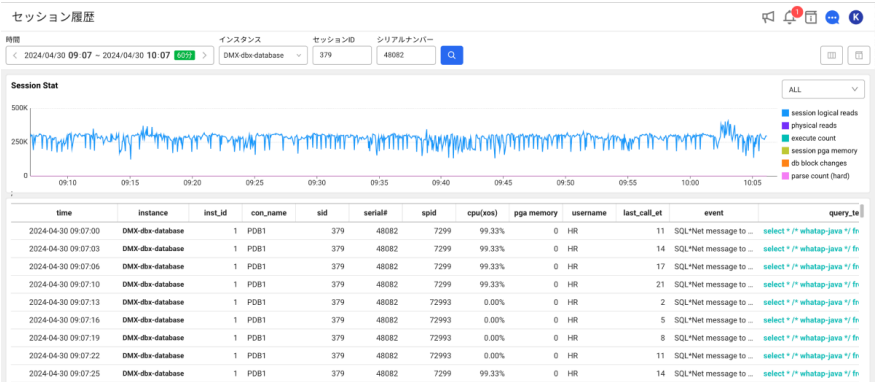
ホーム画面 > プロジェクト選択 > 分析 > セッション履歴

セッション履歴メニューでは、特定のセッションが照会期間中にどのように作業を行ったかを照会できます。長時間実行されたセッションに対して、遂行中のSQLがどれ程リソースを使用したかを確認し、原因を分析できる情報を提供します。

基本的な使い方

必須入力値としてモニタリングしようとするセッション番号(プロセス ID)を入力する必要がありますが、インスタンスモニタリングメニューからアクセスするのが効率的です。

1. ダッシュボード > インスタンスモニタリングメニューに移動します。
2. アクティブセッションタブで、ID列のセッションをクリックします。
3. セッション詳細のセッション履歴 > ボタンをクリックして、セッション履歴に移動します。
4. 選択したセッションのプロセスID (PID) に基づいて収集されたセッション履歴を確認できます。



選択したセッションから5秒ごとに収集したアクティブセッション情報を活用してチャートと一覧を構成し、分析に活用できる情報を提供します。

- **Wait:** 待機中のイベントの割合をチャートで確認できます。
- **Cpu:** CPU使用量の変化をチャートで確認できます。
- **Memory:** メモリ使用量の変化をチャートで確認できます。

- **Disk I/O:** ディスク使用量の変化をチャートで確認できます。

一覧の一番右の**query**カラム項目を選択すると、SQL文を確認できる**SQL詳細**画面が表示されます。選択したクエリ(Query)に対するFullTextおよびPlan情報が含まれています。

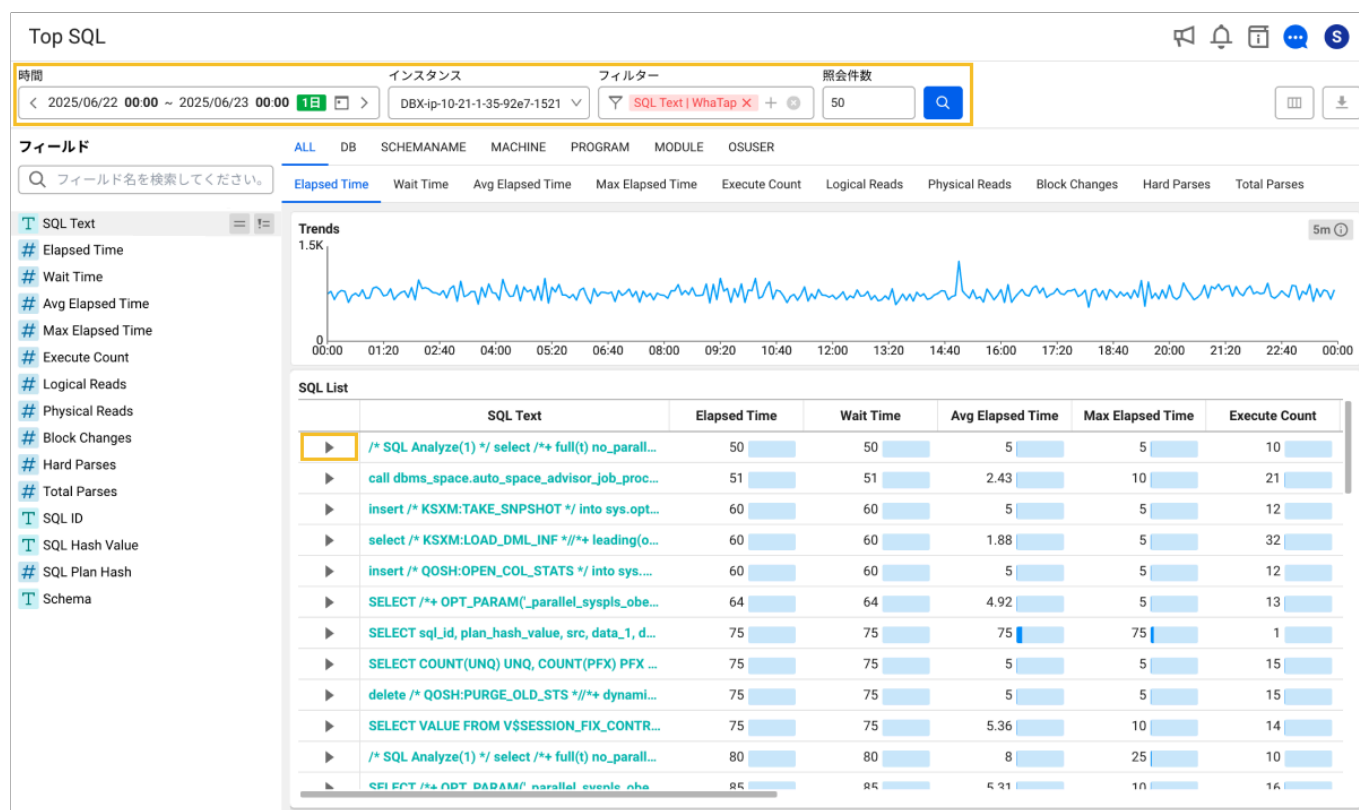
- ① • セッションテーブル一覧で**query_text**のカラム項目を選択すると、SQLクエリ文とPlan情報を確認できる**SQL詳細**画面が表示されます。
- テーブルヘッダーカラムを非表示にしたり、希望する項目を追加できます。カラムの順序を変更することもできます。

Top SQL

ホーム画面>プロジェクト選択> 実験室 > Top SQL

TOP SQLは、既存のSQL統計メニューの機能を拡張し、長い期間データを照会して分析できます。最大30日~最低1分単位で照会でき、分析指標をタブの形で提供して一目で見ることができます。また、推移チャートを追加して特定時間のSQL発生パターンを簡単に把握できます。

基本画面ガイド



1. 時間でデータを照会する日時を選択してください。

- 緑のボタンをクリックして照会時間を選択するか、日付と時間のテキストエリアをクリックすると、日付と時間をオプションとして選択できます。

2. インスタンスでデータを照会する対象を選択してください。

3. フィルター条件を追加してください。フィールドでフィルターに追加するカラムを簡単に検索して追加できます。
4. 照会件数でテーブルに表示する件数を設定してください。
 - データは 10 ~100件までテーブルに表示します。
5.  ボタンをクリックします。
 - 照会した内容をCSV形式のファイルでダウンロードするには、 ボタンをクリックしてください。
 - テーブルヘッダーの各カラムをクリックすると、選択したカラムをソートすることができます。

フィルタリング条件の設定

目的のデータを素早く見つけるために、フィールド条件を追加したり、フィルター条件を設定することができます。

サイドバーフィールドからフィルターに条件を追加する

フィールドでフィールド名を確認し、検索して希望の条件のデータを照会できます。

- フィールド検索機能を使って、目的のフィールドを素早く見つけることができます。
 - フィールドの種類は、 (文字タイプ)、 (数値タイプ) アイコンで区別できます。
1. フィールドリストの希望する項目にマウスを合わせると、そのフィールドのデータタイプに応じて条件アイコンが表示されます。
 - **String** (文字) タイプのフィールド条件： (含む)、 (含まない)
 - **Number** (数値) タイプのフィールド条件： (等しい)、 (以上)、 (以下)
 2. 希望する条件アイコンをクリックして、値を入力します。
 3. 適用ボタンをクリックしてフィールド条件を追加します。

上部オプションからフィルターに条件を追加する

1. フィルターオプションで+アイコンをクリックしてください。
2. フィルターキー項目で目的のフィルタリング基準を選択します。
 - 選択した項目の値が文字に該当する場合は含む(青)、含まない(赤)の条件を選択できます。

example

3. 条件項目で条件を選択し、一致させる文字列または数値を入力してください。
4. 適用ボタンをクリックします。
 - フィルタリング条件を追加するには、追加ボタンをクリックし、1～4の手順を繰り返してください。追加した条件は、AND(&&) 条件として適用されます。

フィルター条件を削除する

- 特定の条件を削除するには、フィルター条件の右側にある  ボタンをクリックしてください。
- すべての条件を削除するには、 完全削除ボタンをクリックしてください。
- フィルターオプションに適用された条件をすばやく削除するには、 ボタンをクリックしてください。



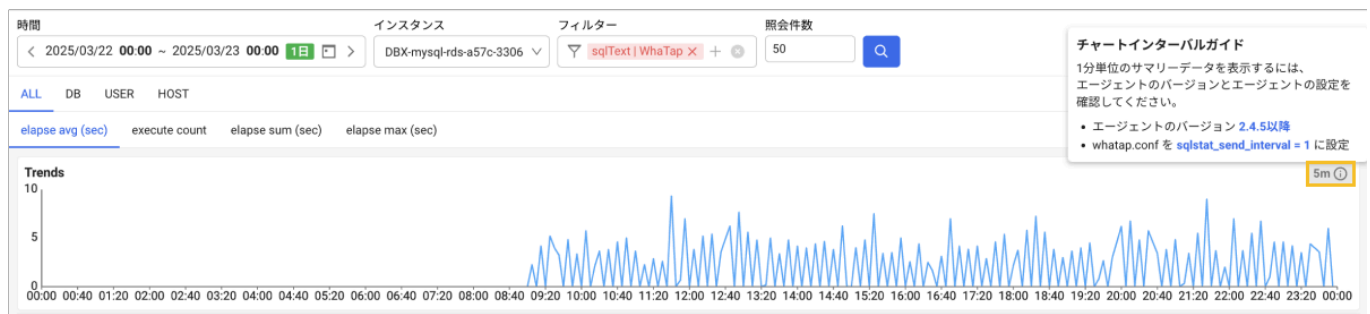
`sqlText` に「WhaTap」が含まれている場合は、除外するようにデフォルトで設定されています。

```
<ImgLang img='db-sql-filter-default.png' desc='WhaTap' />
```

フィルター条件を修正する

1. フィルターオプションに適用されている項目をクリックします。
2. フィルターの修正画面で希望する項目を修正し、適用ボタンをクリックしてください。

Trendsチャートを確認する



Trends チャートのSQL統計データ収集周期とデータ照会範囲は、以下の基準によって異なります。

- 3時間以下
 - SQL統計データ収集サイクルオプションが1分に設定された場合: 1分間隔でデータ収集
 - SQL統計データ収集サイクルオプションが1分に設定されていない場合: 5分間隔でデータ収集
- 1日以下: 5分間隔でデータ収集
- 30日以下: 1時間間隔でデータ収集

ⓘ SQL統計データ収集サイクルオプションの設定(チャート・インターバル)

1分単位のサマリー データを表示するには、エージェントバージョンとエージェントの設定を確認する必要があります。

1. エージェントバージョン2.4.5以降に更新します。
2. whatap.confファイルに `sqlstat_send_interval = 1` の設定を追加します。

クエリ別の実行推移を確認

使用区分ごとに希望するデータを分類して照会したり、ソート順を変更することができます。希望するタブをクリックしてください。

テーブルの項目別の左の▶ボタンを選択します。クエリ文の5分当たりの実行回数と平均実行時間が表示されます。各棒グラフにカーソルを合わせると、その時間の数値が表示されます。

ソート基準	説明
elapsed avg (sec)	平均実行時間が最も長い値からソート
execute count	実行回数の多い順にソート
elapsed sum (sec)	総実行時間が長い値からソート
elapsed max (sec)	最大実行時間が長い値からソート
elapsed wait	クエリの待ち時間が長い順にソート
physical read	ディスクから読み取ったデータブロックが多い順にソート
session logical reads	メモリから読み取ったデータブロックが多い順にソート

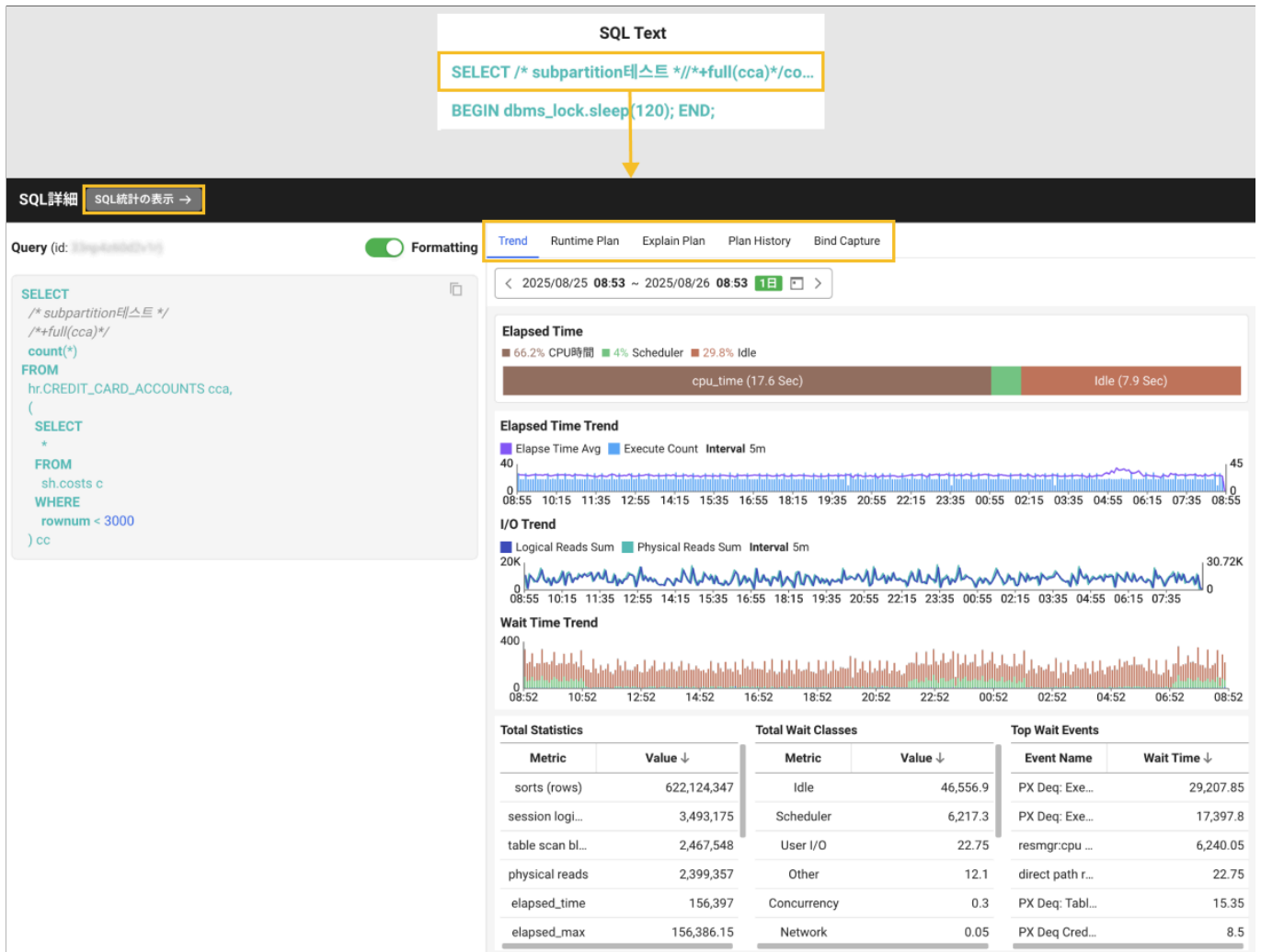
クエリプレビュー

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

照会された結果リストからマウスポインタを[sql](#)カラム項目に移動すると、クエリ文をプレビューできます。

クエリの詳細を見る

照会された結果一覧で[query](#)項目を選択するとSQL詳細画面が表示されます。



SQL統計の表示 → ボタンをクリックすると、該当SQLクエリに関連する統計情報を確認できる **SQL統計** メニューに移動します。

- **Trend:** 特定のSQLの実行時間、I/O、待機イベントを総合的に分析し、性能ボトルネックを迅速に確認できます。
- **Runtime Plan:** 選択したSQLクエリの実行計画とランタイム情報を提供します。実行回数、平均実行時間、平均物理読み取りなどの詳細情報を確認できます。
- **Explain Plan:** オプティマイザが予測した実行計画を表示します。コスト、操作、オブジェクト名、カーディナリティなどの情報を提供します。
- **Plan History:** データベースで実行されたSQLクエリの実行計画履歴を確認できます。
- **Bind Capture:** データベースで実行されたSQLクエリに使用されたバインド変数の値を確認できます。これによりクエリ実行の実際の内容を確認できます。

- ① 実際のリアルタイムのバインド値ではなく、データベースにキャプチャされた値(`v$sql_bind_capture`)です。件数が多い場合は100件まで表示されます。

Trend タブ

上部のチャートでは時間帯ごとの Elapsed Time、I/O、Wait Time の推移を視覚化します。下部の統計テーブルでは累積実行統計、Wait Class 比率、主要な Wait Event を一目で比較できます。

Elapsed Time

タイムセレクトで選択した区間の CPU使用時間、Wait Class時間、Idle時間などを合算して表示します。

例:

- CPU Time: SQL実行に実際にCPUを使用した時間
- Scheduler: セッションがCPUを待機していた時間
- Idle: 実行されずに待機またはアイドル状態にあった時間（どの項目が最も時間を消費したかを直感的に確認可能）

Elapsed Time Trend

時間ごとに 平均 Elapsed Time と実行回数(Execution Count)を5分単位で集計して推移を表示します。3時間未満なら1分単位で表示します。

特定の時点でElapsed Timeが急増したか、実行回数に比べ応答時間が増加したかを確認でき、性能劣化のタイミングを迅速に把握できます。

I/O Trend

クエリ実行中に発生した Logical Reads と Physical Reads を5分単位で表示します。3時間未満なら1分単位で表示します。

- Logical Reads: バッファキャッシュからデータを読み込んだ回数
- Physical Reads: ディスクから直接データを読み込んだ回数で、キャッシュ効率やI/O負荷を比較して性能ボトルネックを特定可能

Wait Time Trend

クエリ実行中に発生した **Wait Event時間** を時間ごとに表示します。

特定の時点でWait Timeが急増していれば、DBリソース競合やI/O遅延が発生していることを意味します。どのイベント（ロック、I/O、並列処理など）がボトルネックを引き起こしたかを追跡できます。

Total Statistics

選択区間中の **SQL実行累積統計** を表示します。

例:

- sorts(rows): ソートされた総行数
- session logical reads: セッション論理読み取り回数
- table scan blocks gotten: テーブルスキャン時に読み込んだブロック数
- elapsed_time: 総経過時間
→ SQL実行がリソースをどのように使用したかを集計データで確認可能

Total Wait Classes

SQL実行中に発生した **待機時間(Wait Time)** を Wait Class 単位で合算した値です。

Wait Classごとの時間が長いほど、その領域がボトルネックである可能性が高まります。

例: Idle, Scheduler, User I/O, Concurrency など

Top Wait Events

選択区間で発生した **詳細な Wait Event の上位リスト** を表示します。

Wait Classより細かく、どのイベントが性能に最も影響を与えたかを確認できます。

例: PX Deq: Execution Msg, resmgr:cpu quantum, direct path read など

詳細チャートを確認する

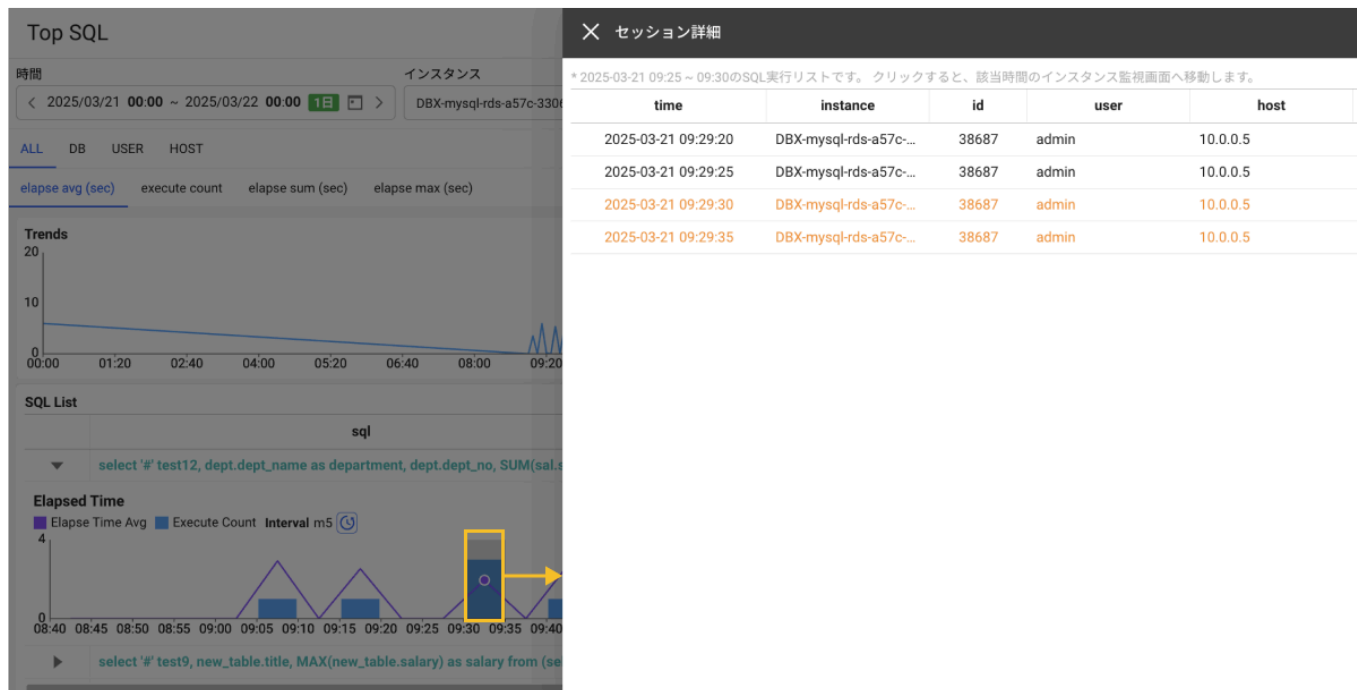
詳細チャートを確認する

データは、照会する時間に応じて別のサイクルに読み込みされます。照会可能な時間範囲は**1分(m1), 5分(m5), 1時

間（h1）単位で提供されます。最近の時間に再度変更するには、![最近時間アイコン](/img/ic-recent-time-active.svg)アイコンをクリックします。設定された時間に応じてデータが自動的に調整されます。

詳細チャートはトレンドチャートと同じように照会時間によってデータの周期が変わり、ドラッグして照会時間範囲を調整できます。

詳細情報を確認する



チャートの棒グラフを選択すると、5分間の統計情報が確認できる**セッション詳細**画面が表示されます。選択した時間のSQL実行一覧を表示し、項目を選択すると新しい画面とともに**インスタンスメニュー**へ移動します。

インスタンスメニューからは、選択した時間帯のアクティブセッションテーブルと、様々な指標の推移を確認することができます。

テーブルカラムの設定

テーブルヘッダーカラムを非表示にしたり、希望する項目を追加できます。カラムの順序を変更することもできます。ボタンをクリックします。

- ❗ • 設定を完了した後は、**確認**ボタンを選択すると、設定事項がテーブルに反映されます。
- ③ 検索バーにテキストを入力し、カラム項目を検索できます。入力したテキストと一致するカラム項目が表示されます。
- 画像は、製品またはプロジェクト、メニューによって異なる場合があります。

カラムを追加する

① 一覧でテーブルヘッダーカラムに追加する項目を選択します。すべての項目を追加するには、**すべて選択**を選択します。

カラムを削除する

① 一覧から削除するカラム項目のチェックボックスのチェックを解除します。または② 一覧から削除する項目の右側の×ボタンを選択します。

カラムの順序の変更

② 一覧で順序を変更するアイテムをドラッグして、希望する位置に移動できます。

設定事項の初期化

変更事項をすべてキャンセルし、初期化するには、🔄 **初期化**ボタンを選択します。

SQL Plan 分析

ホーム画面 > プロジェクト選択 > 分析 > SQL分析

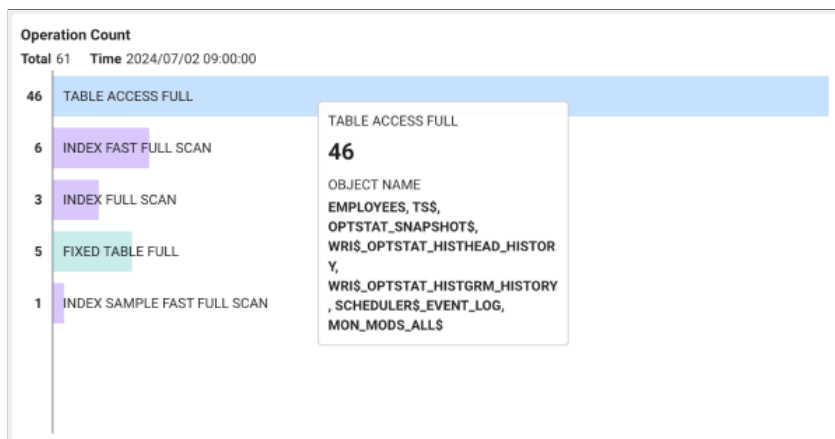
データベースで実行される SQL 文を分析し、パフォーマンス問題を診断するための有用な情報を提供します。
SQL Plan 分析は [Access Statistics](#) タブと [Plan Change History](#) タブで構成されています。

Access Statistics

Access Statistics タブでは、SQL ごとの Full Scan の発生時点、種類、回数を確認できます。

頻繁に Full Scan が発生するクエリは、インデックスを使用するように修正することで、パフォーマンスを改善し、リソース使用量を削減できます。

1. 分析 > SQL Plan 分析 の **Access Statistics** タブに移動します。
2. 照会時間とインスタンスを選択します。
3. 時間 および インスタンス オプションを設定し、 ボタンをクリックします。
必要に応じて、フィルター オプションを設定して結果をフィルタリングできます。
4. **Access Count** と **Operation Count** セクションで検索結果を確認します。
 - **Access Count:** 時間帯ごとの Full Scan 発生回数を表示する棒グラフです。
特定の時間帯を選択すると、その時間に実行された SQL の一覧が画面下部に表示され、右側の **Operation Count** チャートも更新されます。
 - **Operation Count:** Full Scan が発生した種類とその回数を確認できます。
各バーにマウスを重ねるとオブジェクト名が表示されます。



- ❗ • テーブル一覧でquery_text項目を選択するとSQL詳細画面が表示されます。

参照結果フィルタリング

取得結果は次の基準でフィルタリングできます。

- Access Type: Full Scan または Index Scan を基準に結果をフィルタリング
- Object Name: オブジェクト名を基準にフィルタリング

Access Typeフィルター条件を追加する

1. 画面上部のフィルター オプションで、+ ボタンをクリックします。
2. フィルターの修正 で Access Type 項目と 一致 条件を選択し、FULL または INDEX を選びます。
3. 適用 ボタンをクリックします。
4. 選択した条件を SQL 一覧に反映するには、🔍 ボタンをクリックします。

- ❗ Full Scan、Index Scanをすべて検索するには、**フィルターの修正**でFULL、INDEXオプションをすべて選択します。または+ボタンを選択してフィルターオプションを追加できます。

Object Nameフィルター条件を追加する

1. 画面上部の **フィルター** オプションで、**+** ボタンをクリックします。
2. **フィルタの追加** で **Object Name** 項目と **一致** 条件を選択します。
3. 空の入力欄をクリックし、表示されたオブジェクトリストから目的のオブジェクト名を選択します。
4. **適用** ボタンをクリックします。
5. 選択した条件を SQL 一覧に反映するには、 ボタンをクリックします。

❗ フィルター条件の追加と削除

- フィルタリング条件を追加するには、**追加** ボタンをクリックし、手順 1～5 を繰り返します。追加された条件は AND (&&) 条件として適用されます。
- 条件追加中に一部の項目を削除する場合は、フィルター条件の右側にある  ボタンをクリックします。すべての条件を削除するには、 **完全削除** ボタンをクリックします。
- **フィルター** オプションに適用された条件をすばやく削除するには、**×** ボタンをクリックします。

フィルター条件を修正する

1. 画面上部の **フィルター** オプションで、適用されている項目をクリックします。
2. **フィルターの修正** ウィンドウで目的の項目を修正し、**適用** ボタンをクリックします。

Plan Change History

Plan Change History タブでは、同じ SQL ID でも実行計画が変更されるとパフォーマンスに影響を与える可能性があります。

オプティマイザーによるプラン変更を検知・監視することで、不要な変更を防ぎ、SQL パフォーマンスの一貫性を維持できます。

1. **分析 > SQL Plan 分析** の **Plan Change Historys** タブに移動します。
2. 照会時間とインスタンスを選択します。
3. **時間** および **インスタンス** オプションを設定し、 ボタンをクリックします。

4. **Plan Change Count** セクションで特定の時間帯を選択すると、その時間に発生したプラン変更の一覧が表示されます。
 - **Plan Change Count** セクション: 時間帯ごとに発生したプラン変更の回数を確認できる棒グラフチャートです。

Plan変更事項を確認する

1. **SQL Plan 分析** の **Plan Change Historys** タブの一覧から特定の変更項目をクリックします。
2. **Query** セクションに、プラン変更前後の詳細が表示されます。
プランの差分を比較して、パフォーマンス変化の原因を特定できます。
 - **Query** セクションの右上にある  ボタンをクリックすると、そのセクションを新しいウィンドウで確認できます。
3. **Query** セクションを閉じるには、右上の  ボタンをクリックします。

参照結果フィルタリング

照会結果は次の基準でフィルタリングできます。

- sql_id
- sql_hash_value
- after_plan_hash_value
- before_plan_hash_value

フィルター条件を追加する

1. フィルターオプションで + ボタンを選択します。
2. フィルターキー項目で目的のフィルタリング基準を選択します。
 - 選択した項目の値が文字に該当する場合は **含む(青)**、**含まない(赤)** の条件を選択できます。

- 選択した項目の値が数字に該当する場合は **==** (等しい), **>=** (より大きいか等しい), **<=** (より小さいか等

しい) 条件を選択できます。

3. 条件項目で条件を選択します。
4. 条件に一致する文字列または数字を入力します。
5. 適用ボタンを選択します。

❗ フィルター条件の追加と削除

- フィルタリング条件を追加するには、**追加** ボタンをクリックし、手順 1～5 を繰り返します。追加された条件は AND (&&) 条件として適用されます。
- 条件追加中に一部の項目を削除する場合は、フィルター条件の右側にある  ボタンをクリックします。すべての条件を削除するには、 **完全削除** ボタンをクリックします。
- **フィルター** オプションに適用された条件をすばやく削除するには、 ボタンをクリックします。

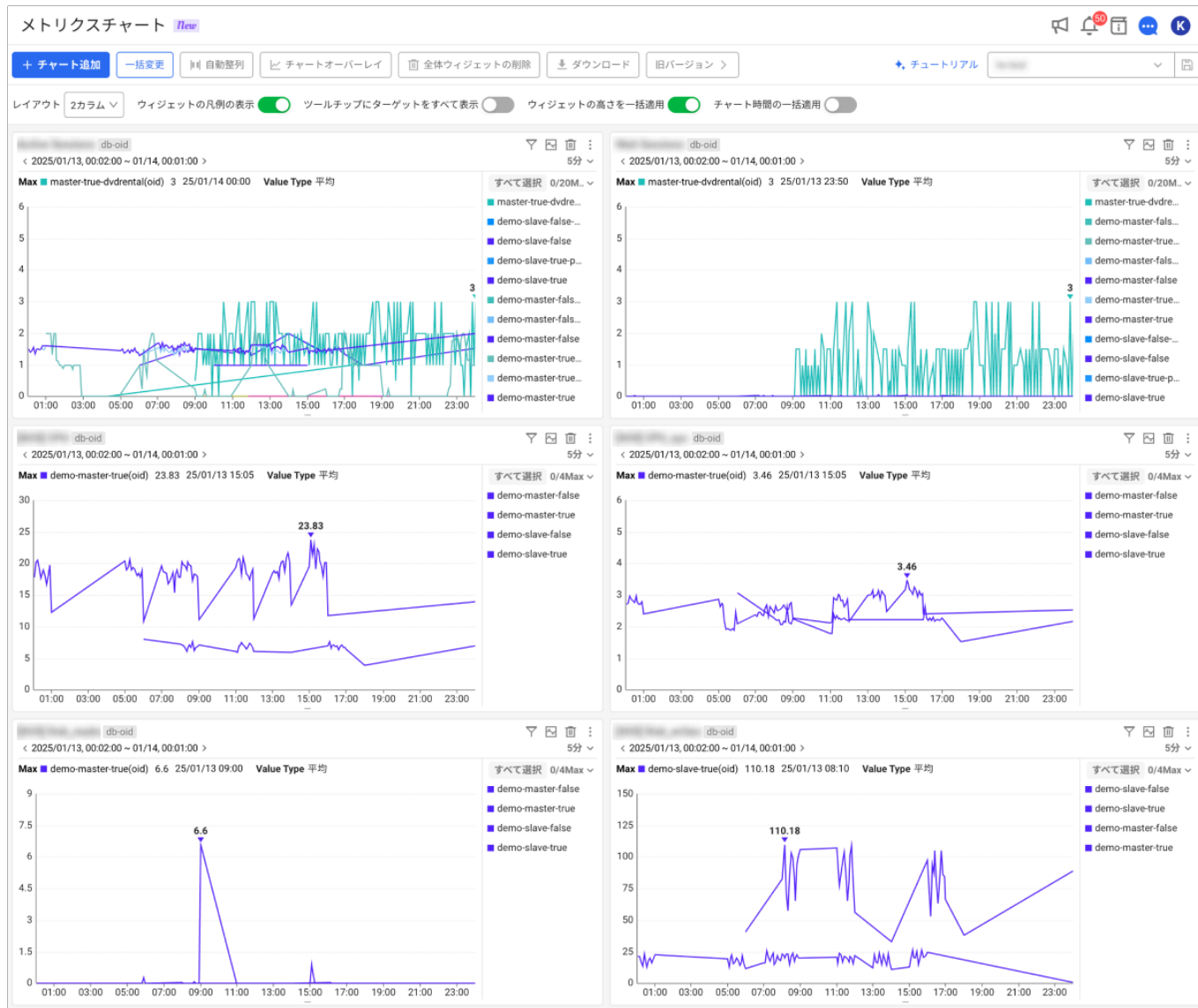
フィルター条件を修正する

1. 画面上部の **フィルター** オプションで、適用されている項目をクリックします。
2. **フィルターの修正** ウィンドウで目的の項目を修正し、**適用** ボタンをクリックします。

メトリクスチャート

ホーム画面 > プロジェクト選択 > 分析 > メトリクスチャート *New*

メトリクスチャート *New* メニューは、WhaTapモニタリングソリューションの主要機能であり、様々なモニタリング対象から収集したデータを時系列チャートで視覚化して提供します。これにより、ユーザーはシステムの性能を効果的にモニタリングし、異常をすばやく識別して、問題解決に必要なインサイトを確認できます。



- 様々なメトリクスを時系列チャートを通じて視覚的に確認でき、障害発生時点の様々なメトリクスを一度に分析できます。
- 特定の時間範囲を選択して過去のデータを照会し、その期間の性能を分析できます。
- 特定の時間帯や条件に合ったデータをフィルタリングすることで、より正確な分析ができます。
- ユーザーは、目的のチャートを画面に配置し、レイアウトを自由に調整できます。
- ユーザーが希望するメトリクスを配置し、プリセットとして保存し、必要に応じて読み込みできます。

❗ 既存のメトリクスチャートメニューとの違い

Service 2.8.0 アップデート以降の変更点は以下の通りです。 アップデート後の変更事項を画面で確認するには、右上の**チュートリアル**ボタンを選択します。

- ユーザーが時間、対象、インターバルオプションを設定した後、モニタリングする対象のメトリクスを特定し、チャートを追加できるようにユーザー経験(UX)を改善しました。
- ダッシュボードに配置されたウィジェットを削除することなく、チャートの照会時間の範囲を変更できます。
- ダッシュボードに配置したすべてのウィジェットのオプションを一括で適用できる**一括変更機能**を提供します。
- プリセット機能により、ダッシュボードに配置されたウィジェットの情報を保存し、読み込みできます。
- 提供する様々なオプションにより、ダッシュボードのレイアウトやウィジェットの機能を調整できます。

チャートウィジェットを追加する

ダッシュボードにメトリクスをウィジェットとして追加するには、画面左上の**チャート追加**ボタンを選択します。

照会時間の設定

時間でメトリクスを照会する日付と時間を設定します。

対象を選択する

プロジェクトに含まれるエージェントの中から、対象を選択します。 個々のエージェントを選択するか、エージェン

ト設定で分類されたグループ単位で選択できます。

対象

全体

エージェント

種類別

サーバー...

対象検索

Q

すべて選択 0/9

すべて折りた...

Q

▼ Project

▼ demo-okind-0

demo-8104

demo-8102

demo-8100

▼ demo-okind-1

demo-8103

demo-8105

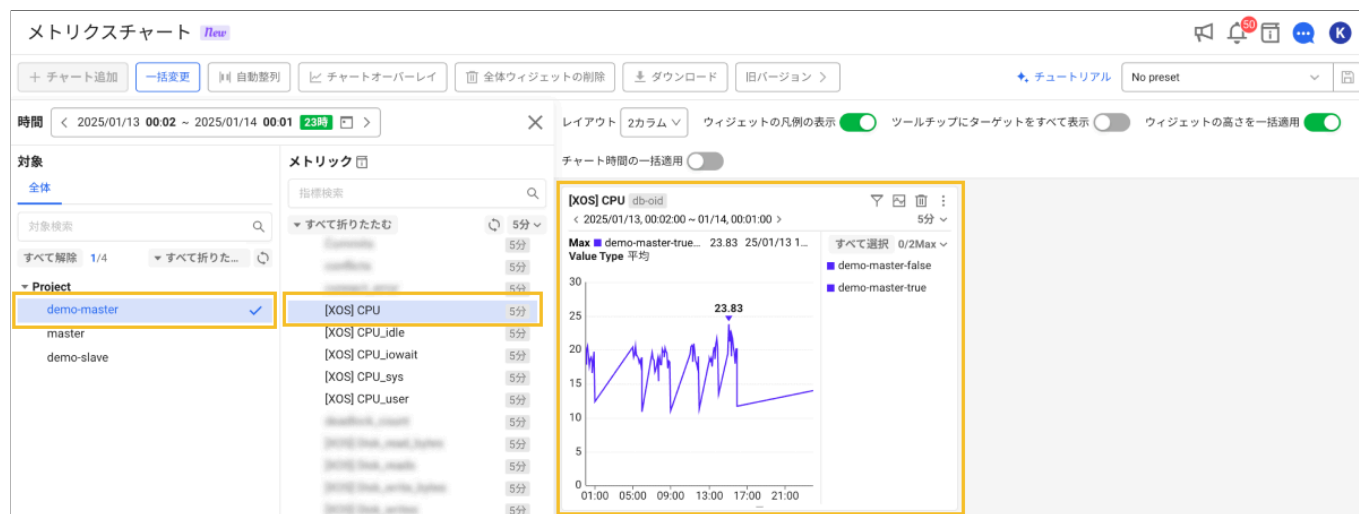
demo-8101

- **エージェント**: 個別エージェントまたはすべてを選択できます。
- **種類別**: エージェント設定で `whatap.okind` に分類されたグループ単位でエージェントをモニタリングします。
- **サーバー別**: エージェント設定で `whatap.onode` に分類されたグループ単位でエージェントをモニタリングします。

- ❗
- 設定した時間と日付によって、選択できる対象が変更される場合があります。
 - エージェントの一覧からマウスをクリックして、対象を選択または解除できます。 `Shift` キーを使用して複数のエージェントを選択できます。
 - **すべて選択**または**すべて解除**ボタンを選択すると、関連する下位対象をすべて選択または解除できます。
 - 希望のエージェントが一覧に表示されない場合は、**リフレッシュ**ボタンを選択します。
 - **すべて折りたたむ**または**すべて開く**ボタンを選択してグループ単位で一覧を隠したり、展開したりできます。

指標を選択する

モニタリングする対象エージェントを選択した後、**メトリックパネル**に選択できる一覧が表示されます。一覧からご希望のメトリクスを選択します。



選択したメトリクスに対応するチャートウィジェットが右側のダッシュボードに配置されます。他のメトリクスを選択し、チャートウィジェットを追加できます。チャート領域を拡大して見るには、**時間オプション**の右にある×ボタンを選択します。**対象とメトリックパネル**が消え、チャート領域のみが画面に表示されます。

- ① **指標検索入力欄**に文字列を入力すると、希望のメトリクスをすばやく検索できます。
- メトリクスのカテゴリとフィールド名に基づいても検索できます。カテゴリおよびフィールドの詳細については、[次の文書](#)を参照してください。
- すべて折りたたむ**または**すべて開く**ボタンを選択して、カテゴリ単位でメトリクス一覧を非表示または表示できます。
- 希望するメトリクス項目が一覧に表示されない場合は、**リフレッシュ**ボタンを選択します。

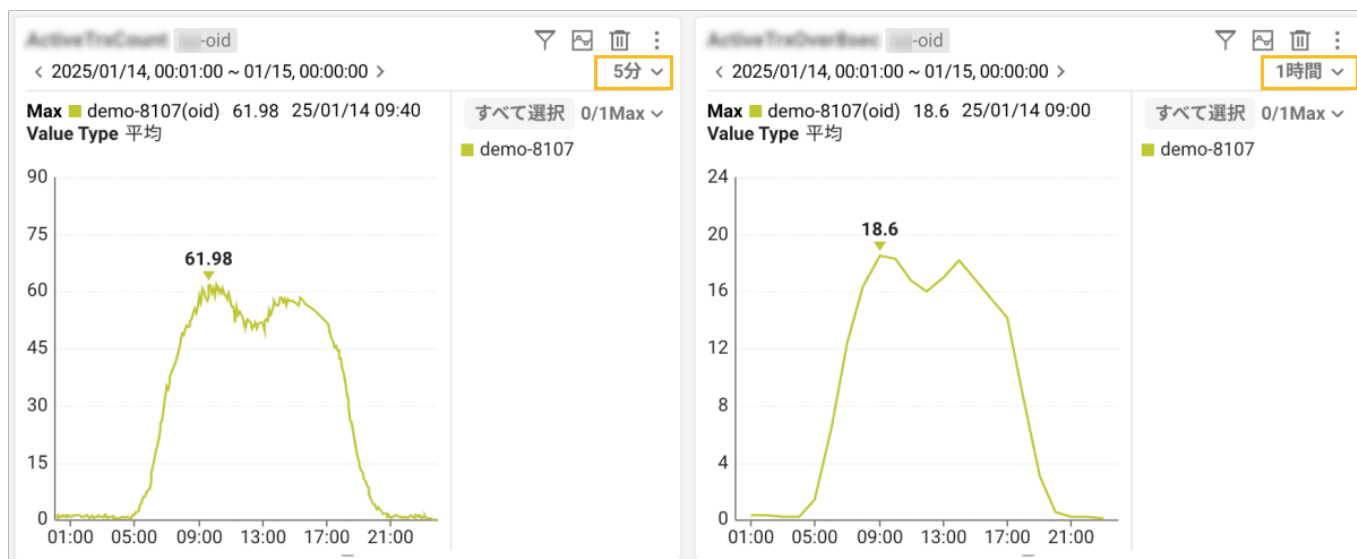
インターバルを設定する



チャートごとにインターバルを設定でき、データの集計単位を調整して、様々な時間間隔のデータを可視化できます。メトリックパネルの右上にある🕒ボタンを選択すると、インターバルオプションを選択できます。

メトリクスごとにデフォルトで設定されたインターバル時間が異なり、照会時間の範囲によってインターバル時間が変更されます。

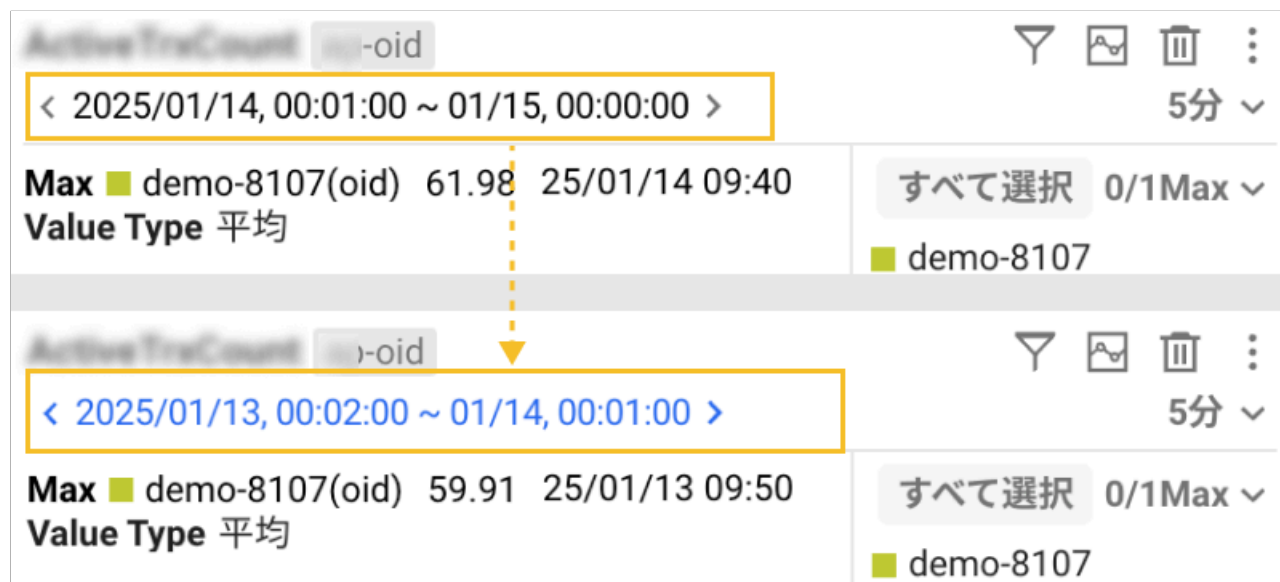
例えば、**5分間**インターバルは5分単位の平均値を表示し、**1時間**インターバルは、1時間単位の平均値を表示します。



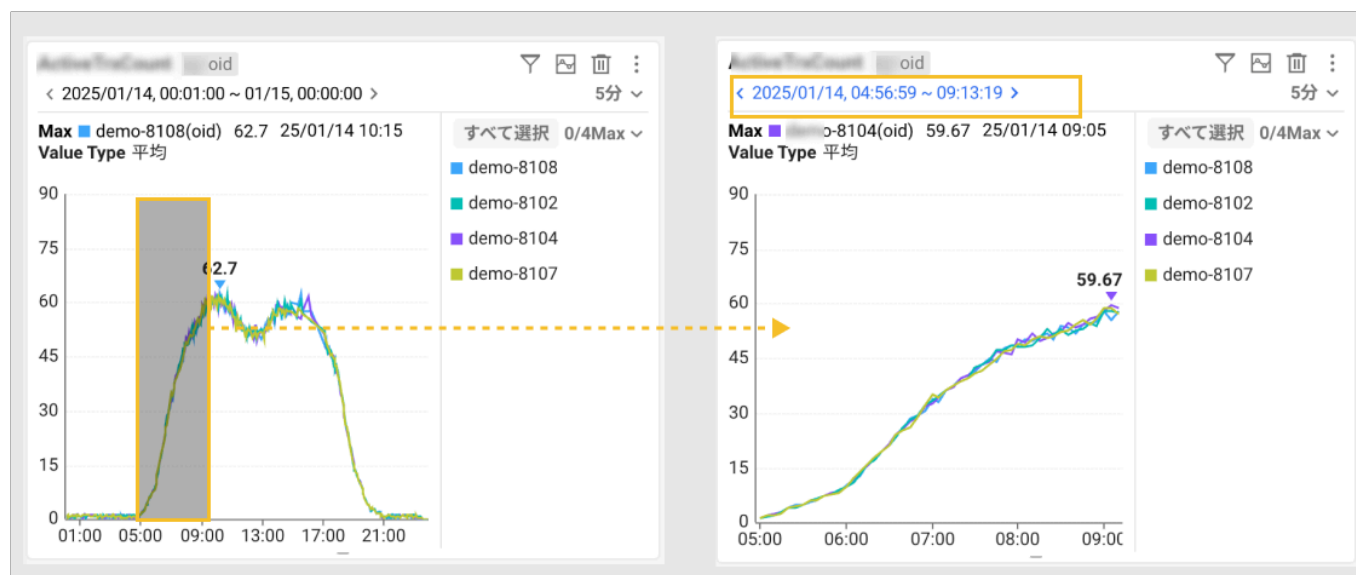
- ☑ インターバルオプションを調整し、短い間隔の詳細データから長い間隔の全体的な推移まで、多様な観点からデータを分析できます。

照会時間を変更する

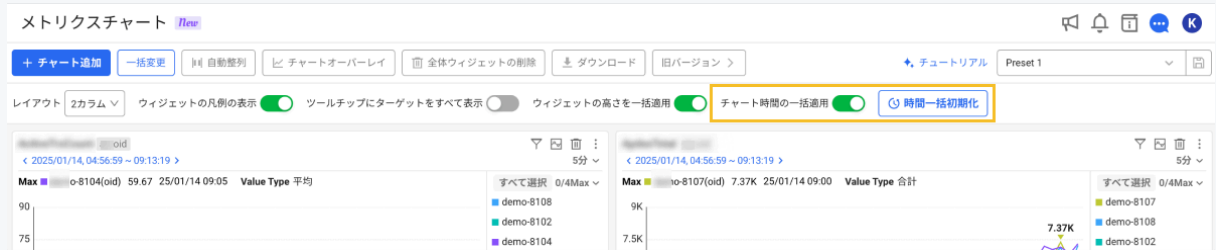
ウィジェットのチャート照会時間の範囲を変更するには、ウィジェット上部の< または > ボタンを選択します。時間変更範囲は、時間オプションに設定された時間基準と同じです。例えば、時間オプションで1日間に設定されている時、< または > ボタンを選択すると、時間を1日単位で変更できます。



またはチャートの特定領域をドラッグすると、ドラッグした範囲に時間を変更して照会できます。



- 画面の上部にある**チャート時間の一括適用**オプションを有効にして、ウィジェットの時間を変更すると、すべてのウィジェットの時間を変更できます。

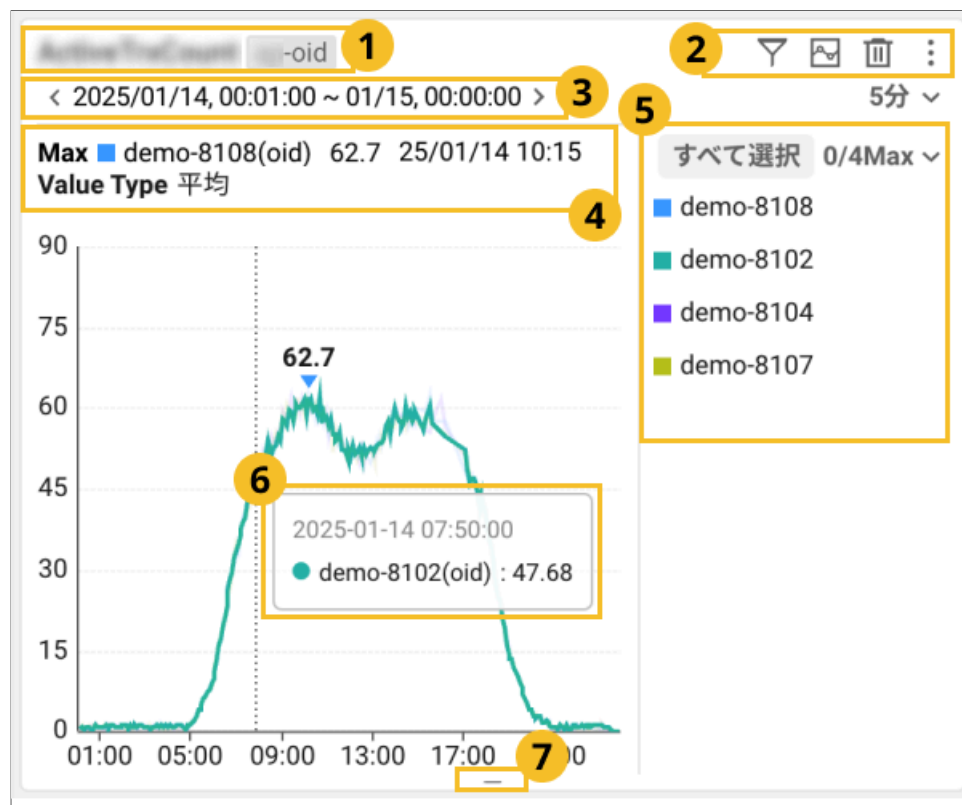


- 照会時間の範囲を変更した後、直近の時間に再度変更するには、**時間一括初期化**ボタンを選択します。
- もし画面の上部にある**チャート時間の一括適用**オプションを有効にした場合、**時間一括初期化**ボタンが表示されます。このボタンを選択すると、すべてのウィジェットの時間を初期化できます。

- 一括変更機能**を利用してウィジェット全体の時間を変更し、モニタリング対象とインターバルも修正できます。詳細については、[次の文書](#)を参照してください。

チャートウィジェットの詳細

チャートウィジェットに表示される各アイテムの詳細を確認します。



- **① メトリクス名とカテゴリ:** チャートウィジェットのメトリクス名を表示します。メトリクスの右側には、照会対象のカテゴリをタグ形式で表示します。ご利用の商品によりカテゴリ名称は異なる場合がございます。
 - `*oid` : 個々のエージェントを選択した時に表示されます。
 - `project` : プロジェクト単位で対象を選択した時に表示されます。
 - `okind` : エージェント設定で `whatap.okind` で分類されたグループを選択した時に表示されます。
 - `onode` : エージェント設定で、`whatap.onode` で分類されたグループを選択した時に表示されます。
- **② ウィジェットオプション:** チャートウィジェットのオプションメニューです。詳細については、[次の文書](#)を参照してください。インターバルオプションの詳細については、[次の文書](#)を参照してください。
- **③ 時間:** チャートのデータを照会する時間範囲を表示します。＜ または ＞ ボタンを選択して照会時間範囲を調整できます。
- **④ データ:** モニタリング対象のうち照会したデータの最大値(Max)とデータ結合方式(Value Type)を確認できます。
- **⑤ 凡例:** モニタリング対象を凡例として表示します。凡例表示の詳細については[次の文書](#)を参照してください。
- **⑥ チャート:** 照会したデータを時系列グラフで表示します。マウスをグラフの上にかざすと、詳細な情報を確認できます。チャートの特定領域をドラッグすると、ドラッグした範囲に時間を変更して照会できます。

- **7 サイズ調整:** チャートウィジェットの真ん中の下部領域をドラッグすると、ウィジェットのサイズを調整できます。

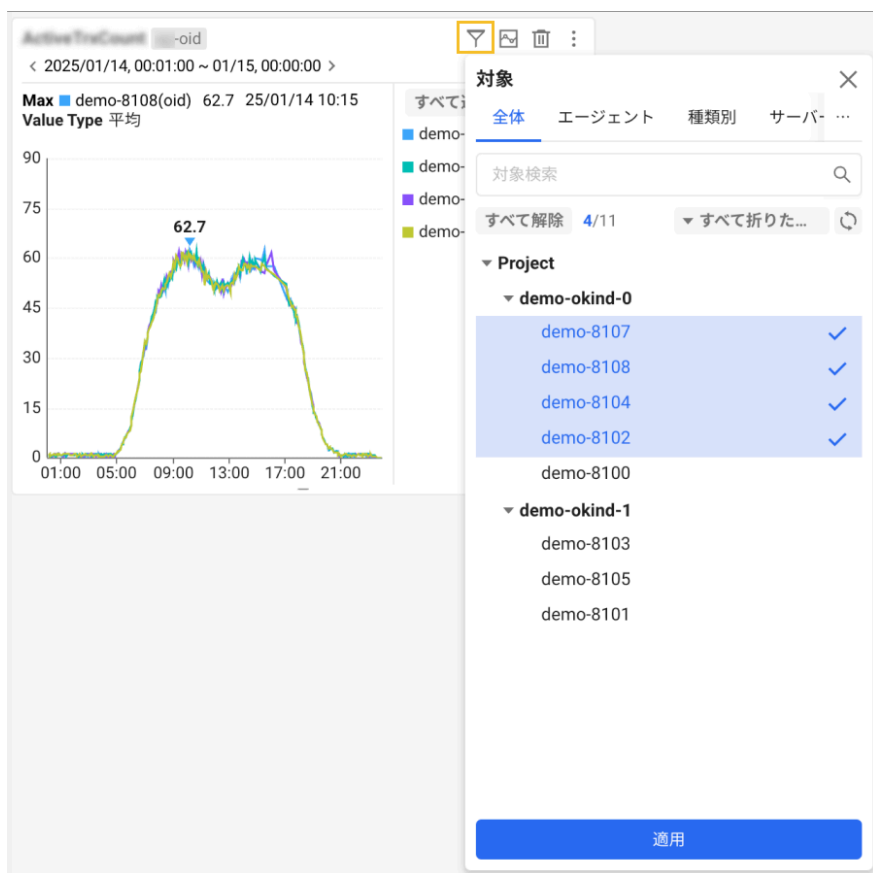
チャートウィジェットオプションを利用する

チャートウィジェットの右上にあるオプションボタンを使用すると、様々な機能が利用できます。

▼対象をフィルタリングする

画面に追加したチャートウィジェットに、モニタリング対象を追加または除外できます。

1. ▼ボタンを選択すると、モニタリング対象を選択できる画面メニューが表示されます。



2. 画面の案内に従って希望する対象を選択します。

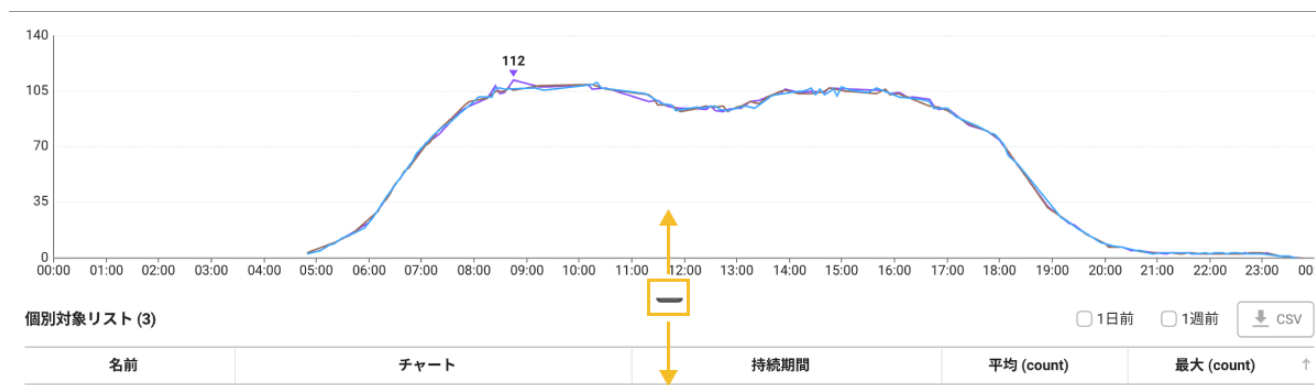
3. 適用ボタンを選択します。

詳細を照会する

チャートのデータをより広い画面で詳細情報を確認するにはボタンを選択します。詳細検索画面が表示されます。個々の対象に対するチャートと平均および最大値を照会できます。



- **1日前**: 1日前のデータと比較して確認できます。
- **1週間前**: 1週間前のデータと比較して確認できます。
- **↓ CSV**: 照会したデータをCSV形式のファイルとしてダウンロードできます。
- グラフチャートの下部にあるアイコンをドラッグすると、チャートのサイズを調整できます。

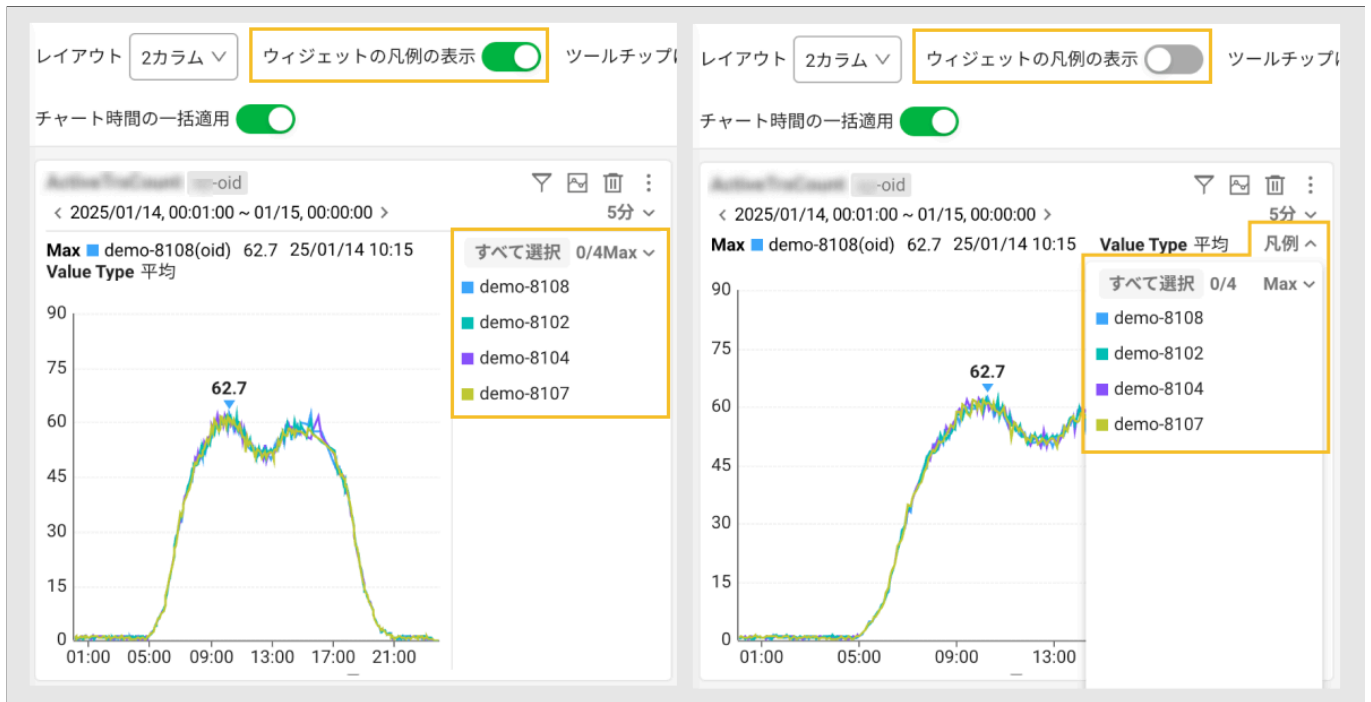


🗑 チャートウィジェットを削除する

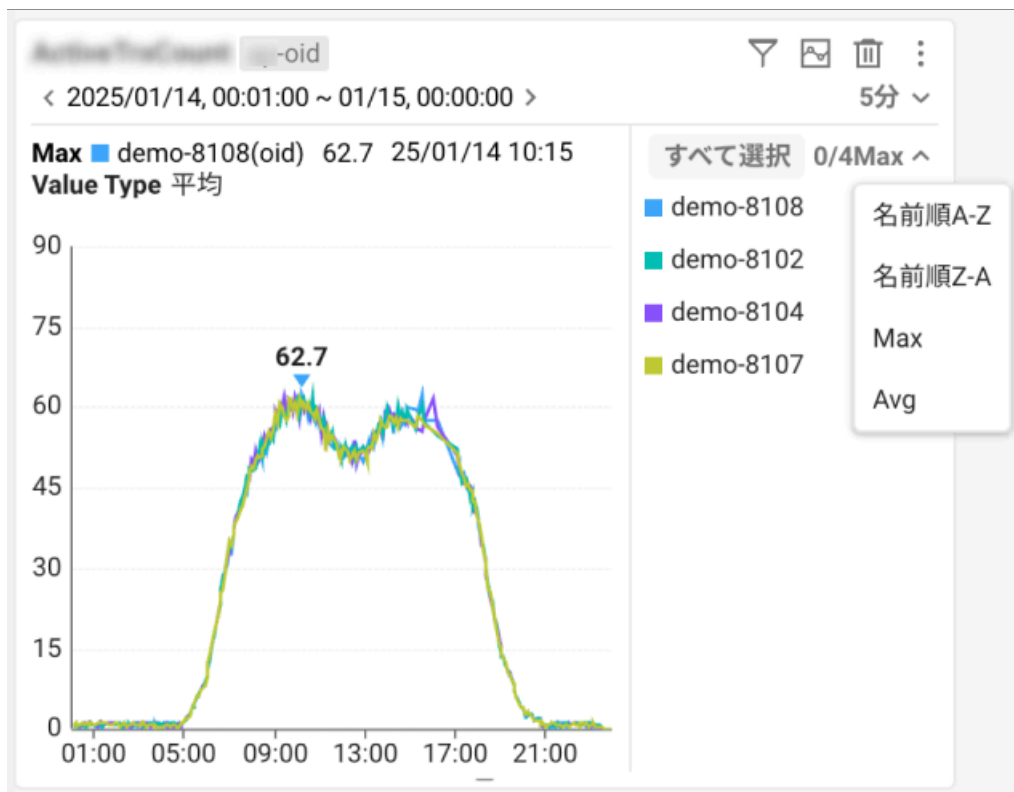
画面に配置したチャートウィジェットを削除するには、🗑 ボタンを選択します。

凡例を活用する

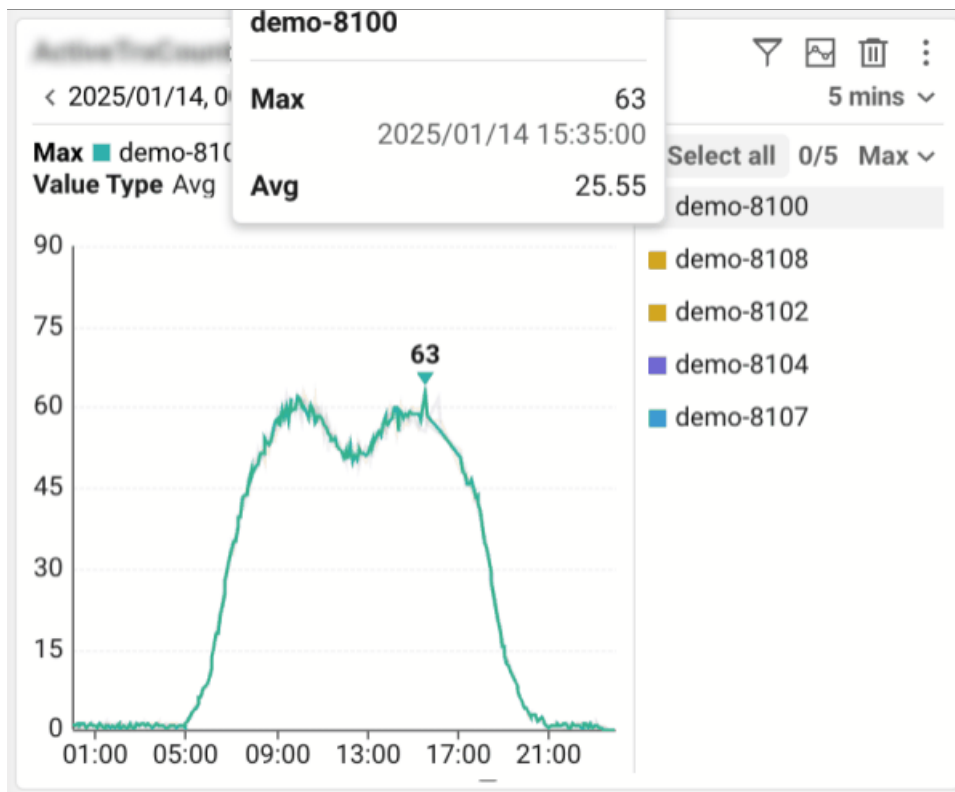
凡例を通じてチャートに表示されるモニタリング対象を確認し、希望する基準で整列したりフィルタリングできます。画面上部のウィジェットの**凡例の表示**オプションをオンまたはオフにすると、ウィジェットのチャートの右側に凡例を表示・非表示できます。オプションをオフにした状態では**凡例**をクリックしてモニタリング対象を確認して選択できます。



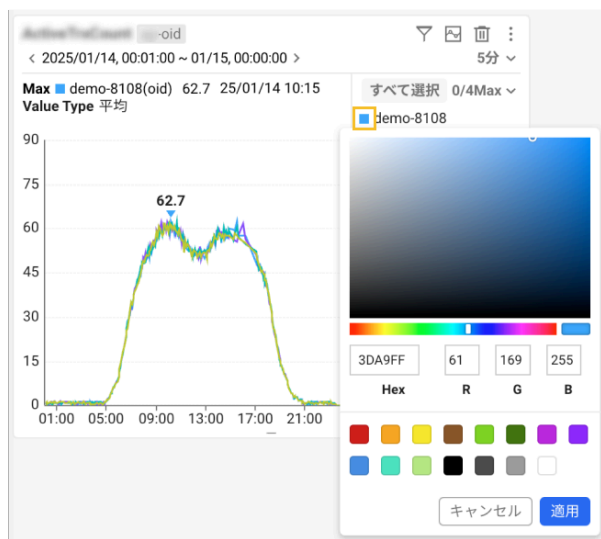
- すべて選択/すべて解除ボタンを選択してモニタリング対象を選択または解除できます。
- 凡例一覧では、モニタリング対象を名前順、最大値、平均値に基づいてソートできます。



- 凡例一覧の各項目にマウスをオーバーすると、そのモニタリング対象に対するグラフを強調して確認でき、Max値が記録された特定時間をツールチップで表示します。または、凡例一覧の対象を選択してもフィルタリングできます。



凡例色を変更する



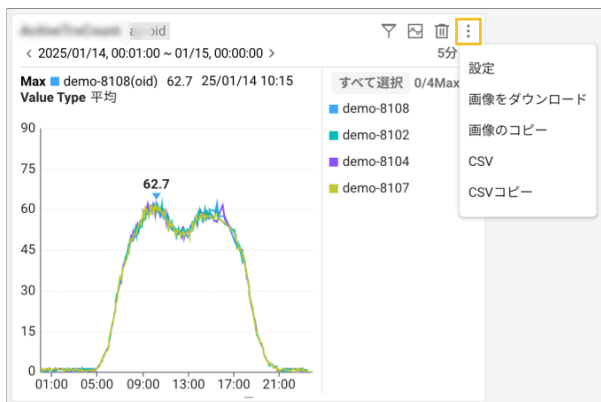
モニタリング対象に該当するグラフの色を変更できます。凡例一覧から色を変更したい対象の左側にある四角い箱

を選択します。希望する色を選択した後**適用**ボタンをクリックします。

❗ 設定した色は、プリセット機能によって保存されません。この機能は、今後のアップデートで提供される予定です。

追加オプションを利用する

チャートウィジェット右上にある  ボタンを選択すると、次の追加オプションが利用できます。



- **設定**
 - **自動Y軸を使用:** 照会したデータ範囲内でY軸の大きさを自動調整できます。
 - **最大値表示:** チャートのグラフ領域に最大値を表示できます。
- **画像をダウンロード:** チャートを画像形式のファイルとしてダウンロードできます。
- **画像のコピー:** チャートを画像形式でクリップボードにコピーできます。
- **CSV:** チャートで照会したデータをCSV形式のファイルにダウンロードできます。
- **CSVコピー:** チャートで照会したデータをCSV形式でクリップボードにコピーできます。

プリセットを設定する

ダッシュボードからユーザーが設定したウィジェットの設定とレイアウト状態を保存して読み込むことができます。ウィジェットのサイズを調整し、希望する場所に配置して新しいプリセットを作成できます。

- ❗ プロジェクト修正権限のあるメンバーのみがプリセットを保存または変更できます。メンバー権限の詳細については、[次の文書](#)を参照してください。

新しいプリセットを作成する

1. ダッシュボードから、希望する形にウィジェットを配置してみてください。サイズを変更して、頻繁に確認するウィジェットのみを配置することもできます。
2. 画面の右上に🏠ボタンを選択します。
3. 新しいプリセット名を入力します。



4. 保存ボタンを選択します。

プリセット一覧から新しく保存したプリセットを確認できます。

- ❗
- 新しく作成したプリセットに変更が加えられた場合は、再度プリセットを保存してください。🏠ボタンを選択し、同じ名前でプリセットを保存します。既存のプリセットに変更を上書きします。
 - ダッシュボードの変更を保存せずに別のメニューに移動すると、変更は保存されません。
 - プリセットはプロジェクト単位で保存され、他のユーザーと共有できます。
 - プリセットに保存される対象は、ダッシュボードのレイアウトとダッシュボードに配置したウィジェット、ウィジェットで修正したオプション(対象, 間隔)です。画面上部の時間オプションは保存されません。

- ❗ プリセットを読み込んでからウィジェットの照会時間を変更するには、**一括変更機能**を利用します。詳細については、[次の文書](#)を参照してください。

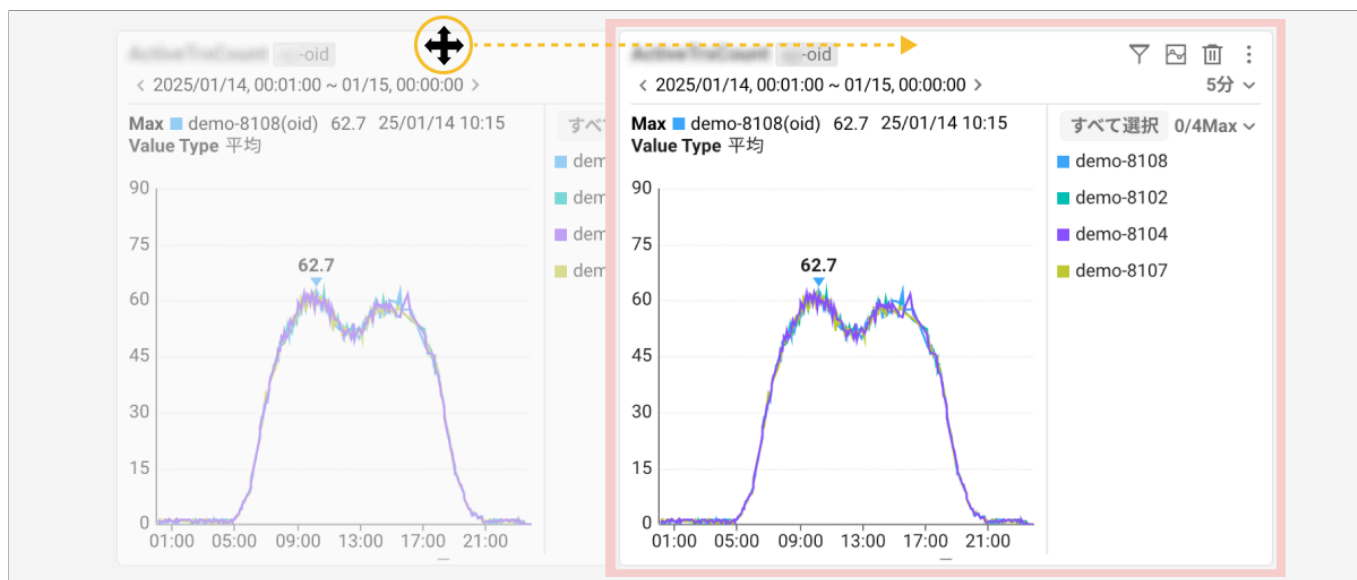
プリセットを削除する

使用していないプリセットは、プリセット一覧から削除できます。プリセット一覧から削除する項目の右側にある×ボタンを選択します。

ダッシュボードウィジェットの編集

ダッシュボードに配置したウィジェットをユーザーが希望する場所に配置したり、サイズを調整できます。

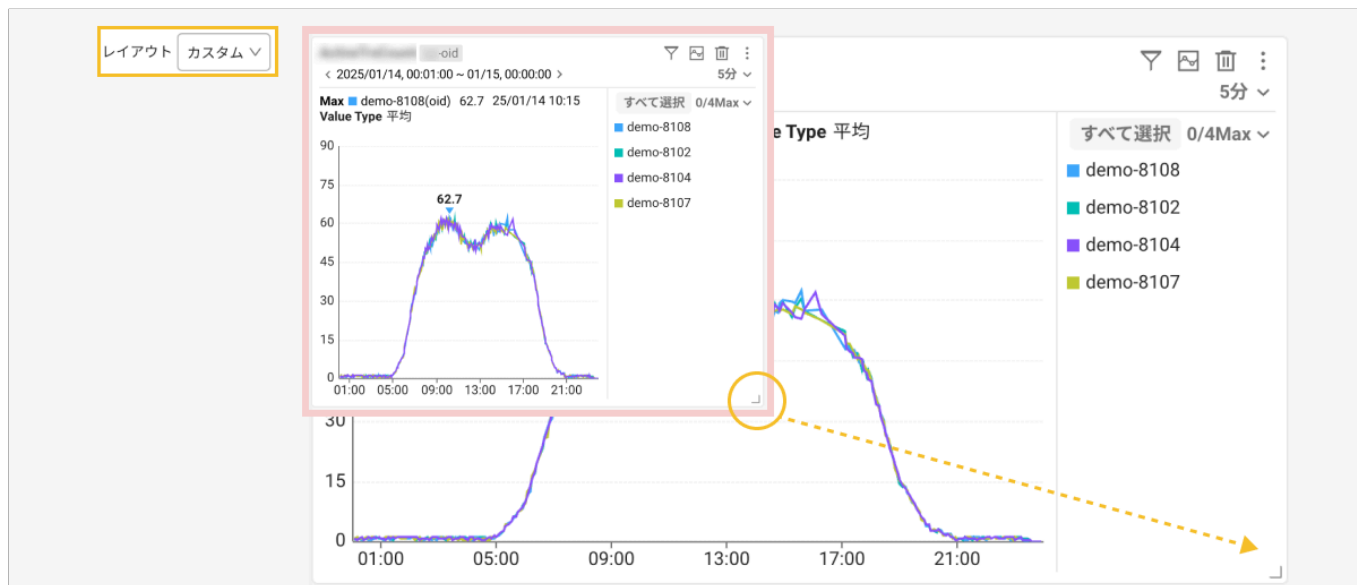
ウィジェットの移動



ウィジェットの上部にマウスをかざすと、**+**に変更されます。この時、クリックしたまま希望の位置にドラッグするとウィジェットを移動できます。

ウィジェットのサイズ変更

ダッシュボードオプションのレイアウトがカスタムに設定されていると、ウィジェットのサイズを調整できます。



ウィジェットの右下にある「」をマウスでクリックした状態で希望するサイズにドラッグします。

- ⓘ レイアウトがカスタムに設定されていない場合は、ウィジェットの高さだけを調整できます。ウィジェットの真ん中の下部領域をマウスでクリックしたままサイズを調整します。

ダッシュボードオプションを利用する

画面上部にあるオプションメニューを利用して、チャートウィジェットに適用された様々な設定を一度に適用できます。また、ダッシュボードのレイアウトを希望する形に設定できます。



チャートウィジェットオプションを一括変更する

画面に配置されたチャートウィジェットに、詳細オプションを一度に適用できる機能を提供します。画面左上に一括変更ボタンを選択します。一括変更画面が表示されたら、オプションを設定し、適用ボタンを選択します。

一括変更

時間

< 2025/01/15 00:00 ~ 2025/01/15 23:59 23時

>

対象

全体

対象検索

すべて解除 1/4

すべて折りたたむ

▼ Project

demo-master ✓

master

demo-slave

間隔

変更しない

5分

1時間

適用ウィジェット

すべて選択

☐ Active Sessions

2025/01/13, 00:02:00 ~ 01/14, 00:01:00 db-oid 5分

☐ Wait Sessions

2025/01/13, 00:02:00 ~ 01/14, 00:01:00 db-oid 5分

☐ [XOS] CPU

2025/01/13, 00:02:00 ~ 01/14, 00:01:00 db-oid 5分

☐ [XOS] CPU_sys

2025/01/13, 00:02:00 ~ 01/14, 00:01:00 db-oid 5分

☐ [XOS] Disk_reads

2025/01/13, 00:02:00 ~ 01/14, 00:01:00 db-oid 5分

☐ [XOS] Disk_writes

2025/01/13, 00:02:00 ~ 01/14, 00:01:00 db-oid 5分

キャンセル

適用

- **時間:** チャートのデータ照会範囲を設定できます。
- **対象:** モニタリング対象を選択します。個々のエージェントを選択したり、エージェント設定を通じて分類したグループ単位で選択できます。

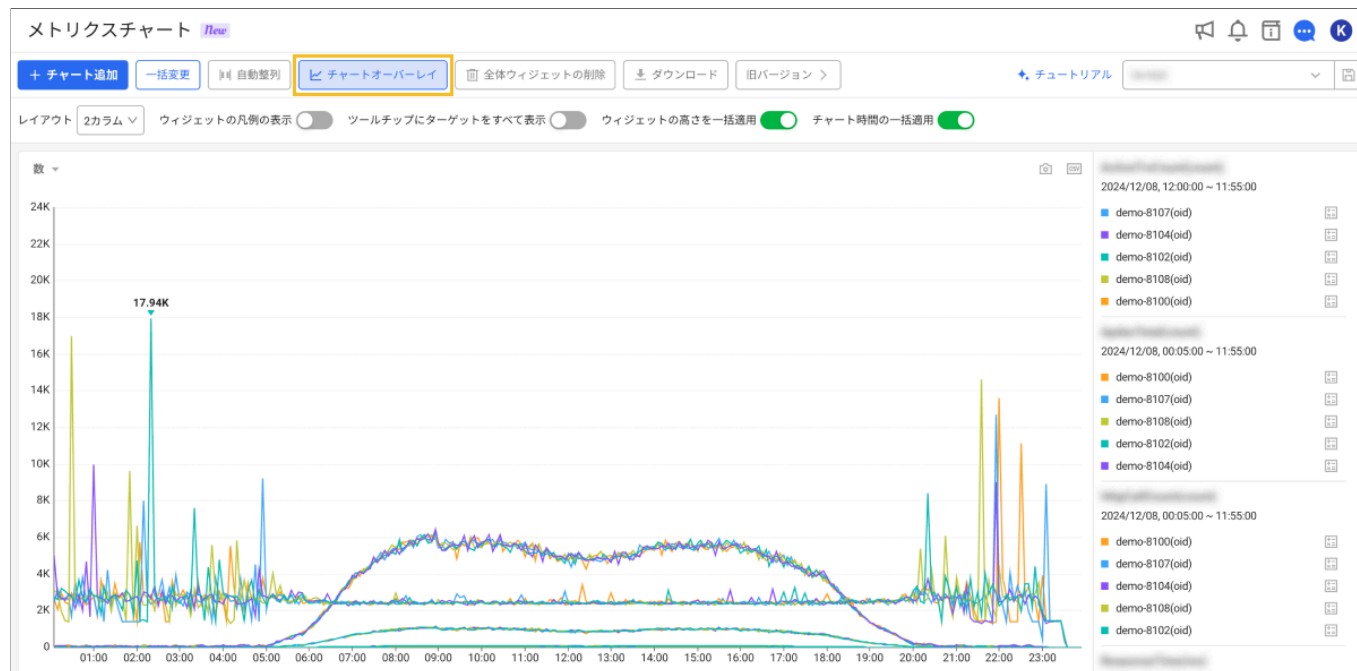
- **エージェント:** 個別エージェントまたはすべてを選択できます。
- **種類別:** エージェント設定で `whatap.okind` に分類されたグループ単位でエージェントをモニタリングします。
- **サーバー別:** エージェント設定で `whatap.onode` に分類されたグループ単位でエージェントをモニタリングします。
- **間隔:** 様々な時間間隔でデータの集計単位を調整できます。
- **適用ウィジェット:** 変更するオプションを適用するチャートウィジェットを選択します。選択したウィジェットに変更したオプションを一度に適用できます。

ダッシュボードを自動ソートする

ダッシュボードに配置したウィジェットの位置を自動的にソートするには**自動整列**ボタンを選択します。ウィジェットの幅、高さが小さい順にウィジェットの配置を自動的にソートします。幅が同じであれば、高さが小さい順にソートします。

複数のチャートを比較する

ダッシュボードに配置した複数のチャートデータを1つのチャートで確認できる機能です。この機能により、同時時間帯の様々なメトリクス情報を他のメトリクスと比較しながら確認できます。**チャートオーバーレイ**ボタンを選択します。画面にオーバーレイチャートを配置できます。



- オーバーレイチャート左上の**数**を選択すると、x軸チャートのデータ基準を変更できる一覧が表示されます。希望する項目を選択して、チャートのデータを比較できます。
- : 照会中のデータに基づいて、チャートを画像形式のファイルとしてダウンロードできます。
- : 照会中のデータに基づいて、チャートをCSV形式のファイルとしてダウンロードできます。
- 右側のセクションでは、メトリクスごとのエージェントの一覧を確認できます。個々のエージェントの色は、チャートのグラフの色と一致します。

配置されたウィジェットを一括削除する

ダッシュボードに配置された複数のウィジェットを一度に削除できます。 **全体ウィジェットの削除**ボタンを選択します。確認メッセージが表示されたら削除ボタンを選択します。

CSVファイルでダウンロードする

ダッシュボードに配置したウィジェットのデータをCSV形式のファイルとしてダウンロードできます。 **ダウンロード > CSV**を選択します。時間帯のメトリクスのデータを確認できます。

イメージファイルでダウンロードする

ダッシュボードに配置されたすべてのウィジェットを1つのイメージファイルでダウンロードできます。 **ダウンロード** > **画像をダウンロード** を選択します。 ダッシュボード画面をそのままイメージで保存できます。

以前のバージョンを利用する

以前のバージョンのメトリクスチャートメニューを利用するには、**旧バージョン**ボタンを選択します。 または **サイトマップ** > **分析** > **メトリクスチャート** **Old** メニューパスで確認できます。

❗ メトリクスチャート **Old** メニューの詳細は、[次の文書](#)を参照してください。

その他のオプションを利用する

次のその他のオプションで、ダッシュボードのレイアウトを調整したり、ウィジェットに適用されたオプションを簡単に一括設定できます。

レイアウト **2カラム** ▼ ウィジェットの凡例の表示 ☒ ツールチップにターゲットをすべて表示 ☐ ウィジェットの高さを一括適用 ☒ チャート時間の一括適用 ☒

- **レイアウト**: ダッシュボードで列の数、つまり1行に配置できるウィジェットの数を設定できます。 最大4列まで設定できます。 **カスタム**を選択すると、列の数に関係なくウィジェットを配置できます。

❗ **カスタム**では、ウィジェットの幅と高さを自由に調整できます。 他のオプションを選択すると、ウィジェットの高さだけを調整できます。

- **ウィジェットの凡例の表示**: オプションをオンまたはオフにすると、ウィジェットのチャートの右側に凡例を表示・非表示できます。 オプションをオフにした状態では**凡例**をクリックしてモニタリング対象を確認して選択できます。 凡例の活用についての詳細は[次の文書](#)を参照してください。
- **ツールチップにターゲットをすべて表示**: チャートにマウスかざすと、そのグラフの位置に該当する値を確認できます。 マウスが位置した時間帯のすべての対象のデータを確認するにはオプションをオンにします。
- **ウィジェットの高さを一括適用**: オプションをオンにすると、すべてのウィジェットの高さを同じく設定できます。

- **チャート時間の一括適用:** チャートウィジェットのデータ照会時間を変更する時、他のウィジェットのチャート時間も一緒に変更するにはオプションをオンにします。

チュートリアルを利用する

画面の右上のチュートリアルボタンを選択すると、[Service 2.8.0](#) アップデート以降の変更を確認できます。

メトリクストレンド分析

メトリクス推移分析 は、過去 30 日以内のメトリクス変化を可視化し、トレンドを比較・分析できる機能です。データベースの主要なパフォーマンス指標を期間別または時間別に照会し、特定の時点や時間帯のパターンを迅速に把握できます。

基本画面

メトリクス推移分析 は、上部の **オプションバー** と下部の **チャート領域** で構成されています。オプションバーで照会範囲と表示メトリクスを選択すると、条件に合ったトレンドグラフが自動的に表示されます。

オプションバー

オプションバーでは、時間範囲の設定、メトリクス選択、プリセット管理などの機能を提供します。

Time Window

特定の時間帯のみのデータを表示したい場合に使用します。
トグルを有効にすると時間範囲入力フィールドが有効になります。デフォルトは無効です。

たとえば、業務時間（09:00～18:00）のデータを分析する場合は、Time Window をオンにして該当時間を指定します。

インスタンス

照会するデータベースインスタンスを選択します。
プロジェクトに接続されたインスタンス一覧が表示され、検索ボックスを使用してインスタンスをすばやく見つけることができます。

メトリクス

表示するメトリクスを選択します。
カテゴリ別にフィールドが一覧表示され、必要な項目を選択して照会できます。

プリセット

頻繁に使用するチャート構成を保存または呼び出すことができます。

- 上部の日付別チャートのクリック状態はプリセットに保存されます。
- 下部の日別トレンドチャートのクリック状態は保存されません。

プリセットの保存

ダッシュボードで設定した指標とレイアウトをプリセットとして保存します。

1. **Time Window**、**メトリクス**などを希望の形で設定します。
2. 画面右上の  アイコンをクリックします。
3. **プリセットとして保存** ウィンドウで名前を入力し、**保存** ボタンをクリックします。

プリセットの呼び出し

1. プリセットドロップダウンリストから保存済みプリセットを選択します。
2. 保存された **Time Window**、**メトリクス**、**チャート**などの設定がそのまま適用されます。

✔ プリセットを保存すると、繰り返し設定することなくすぐに照会できます。

チャート

メトリクス推移分析のチャート領域では、選択したメトリクスの変化を可視化します。

特定期間のトレンド、平均値、最大値などを比較でき、必要に応じて新しいチャートを追加したり不要なチャートを削除できます。

チャートの追加

1. （任意）上部オプションバーで **Time Window** を指定します。
2. **メトリクス** から希望するフィールドを選択します。
3. 画面右上の **+ チャート追加** ボタンをクリックします。

チャートの削除

削除するチャートの右上にある  アイコンをクリックします。

ウィジェット

メトリクス推移分析 の下部チャートは **Interval** と **リンクを開く** 機能を提供します。

これにより特定期間のデータをより詳細に分析したり、該当時間帯の詳細モニタリング画面へ直接移動することができます。

Interval

チャートに表示されるデータの集計単位を変更します。

データを1時間単位または5分単位で照会できます。

デフォルトでは1時間単位の集計データが表示され、より詳細なデータを確認したい場合は5分単位表示に切り替えます。

リンクを開く

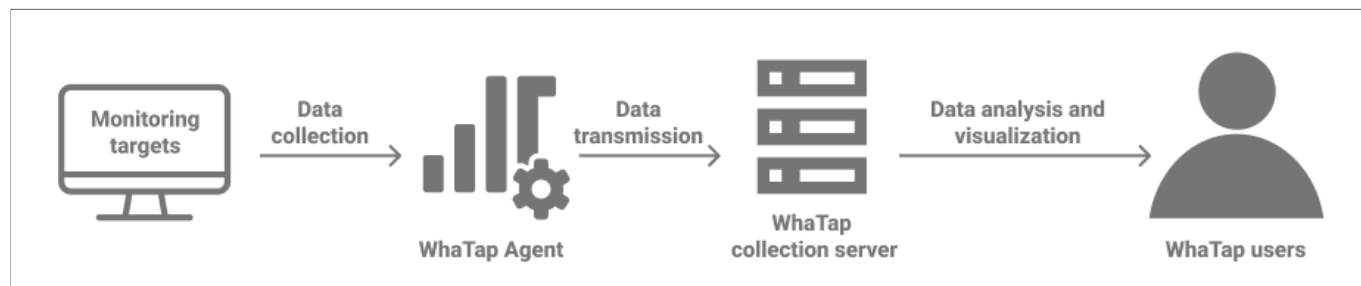
チャートの特定時間区間のバーをクリックすると、その時間に移動して該当区間の詳細モニタリングページを開きます。

❗ **リンクを開く** > ボタンは、特定の時間区間のバーをクリックしたときのみ有効になります。
選択した時間帯の詳細な指標を確認できます。

メトリクス

メトリクスとは？

WhaTapは、モニタリング対象からデータを収集し、ユーザーに提供します。エージェントから収集されるデータをメトリクスと呼びます。



メトリクスは、次の情報で構成されています。

- **Category**：関連する指標をバインドする単位
- **Tags**：収集対象を特定できる個別情報を含むデータ
- **Fields**：エージェントから収集された指標
- **Time**：メトリクスが収集された時間
- **Oid**：メトリクスを収集したエージェントの固有番号

収集したデータを表示するには、ホーム画面でプロジェクトを選択し、[サイトマップ](#) > [メトリクス表示](#)を選択します。

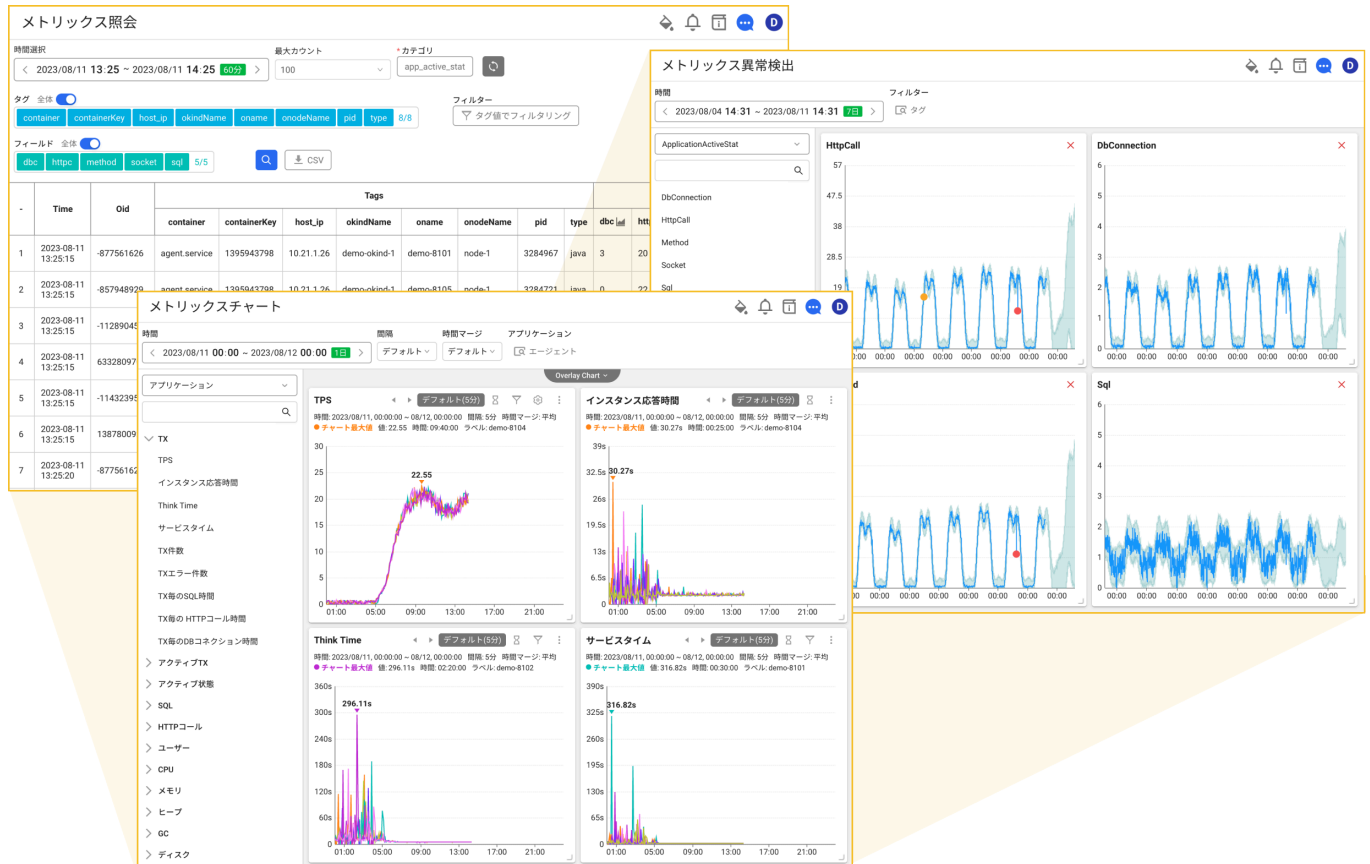
例えば、`db_oracle_dma_counter` という名前のカテゴリを選択し、関連する指標を検索すると、次のようにカテゴリに対応するタグ(**Tags**)とフィールド(**Fields**)項目を表示できます。

.

メトリクスデータの検索と可視化

WhaTapは収集したメトリクスをさまざまな方式で照会できるメニューを提供します。メトリクスの原本データを照会

できるメトリクス照会、可視化したチャートでメトリクスデータを照会できるメトリクスチャート、AIが学習したメトリクスのパターンと比較して予想パターンを外れた異常を探知できるメトリクス異常検出メニューを確認してください。



Oracle Proメトリクス

データベース指標は、データベースの作業量とリソース使用量を包括的に収集します。WhaTapモニタリングサービスは、継続的にメトリクスを収集してデータベースのボトルネックを特定し、性能問題の根本原因を特定するのに役立つデータを提供します。

メトリクス情報に関する詳しい内容は、[次のリンク](#)を参照してください。

agent_status_summary

エージェントの状態に関するメトリクスを10秒間隔で収集したカテゴリです。

Fields

Field	Type	Unit	Description
inActTime	-	ミリ秒(ms)	エージェントが無効になったままの時間
isActive	Boolean	-	現在のエージェントの活性化状態
isRestart	Boolean	-	エージェントが最近再起動したかどうか (<code>true</code> / <code>false</code>)
lastActTime	-	ミリ秒(ms)	最後にエージェントが活性化された状態の時刻 (<code>0</code> : 非活性化された場合)
oid	-	-	プロジェクトに含まれる各エージェントの固有の識別子
startTime	-	ミリ秒(ms)	エージェントが開始された時点でのタイムスタンプ

db_oracle_dma_counter

Oracleデータベースの全般的な性能メトリクスを収集し、システムの状態と性能をモニタリングするのに重要なカテゴ

リです。

Tags

次は、 `db_oracle_dma_counter` カテゴリに収集されるタグに関する案内です。

Tag	Description	Remarks
<code>dbName</code>	データベース名です。	-
<code>oname</code>	エージェント名です。	一意の値

Field

次は、 `db_oracle_dma_counter` カテゴリに収集されるフィールドの案内です。

Metrics	Description	Category
<code>cpu_used_by_this_session</code>	ユーザー呼び出しが開始された時点から終了するまでのセッションで使用したCPU時間(10ミリ秒単位)となります。	Performance
<code>db_time</code>	データベースでユーザーレベルの呼び出しを実行する際にかかった時間(microseconds)です。	Performance
<code>active_sessions</code>	現在、活性化されたセッションの数です。	Connection
<code>connect_error</code>	接続エラーの数です。	Connection
<code>consistent_gets</code>	一貫したデータブロックの読み取り作業の回数です。	I/O
<code>cpu</code>	データベースで使用されたCPUの使用率です。	Performance
<code>cpu_cores</code>	使用可能なCPUコアの数です。	Performance
<code>data_blocks_consistent_read</code> <code>s_-_undo_records_applied</code>	一貫したデータブロックの読み取りに適用されたUndoレコードの数です。	I/O

Metrics	Description	Category
db_block_changes	SGAのブロックで変更されたブロックの数です。	I/O
db_block_gets	SGAのブロックでの直接読み取り作業の回数です。	I/O
db_file_scattered_read_time_waited	Scattered読み取りイベントを待機した時間です。	I/O
db_file_scattered_read_total_timeouts	Scattered読み取りイベントで発生したタイムアウト回数です。	I/O
db_file_scattered_read_total_waits	Scattered読み取りイベントで発生した待機回数の合計です。	I/O
db_file_sequential_read_time_waited	シーケンシャル読み取りイベントで待機した時間です。	I/O
db_file_sequential_read_total_timeouts	シーケンシャル読み取りイベントで発生したタイムアウト回数です。	I/O
db_file_sequential_read_total_waits	シーケンシャルの読み取りイベントで発生した待機回数の合計です。	I/O
direct path read time waited	ダイレクトパスの読み取りで待機した時間です。	I/O
direct path read total timeouts	ダイレクトパスの読み取りで発生したタイムアウト回数です。	I/O
direct path read total waits	ダイレクトパスの読み取りで発生した待機回数の合計です。	I/O
direct path write time waited	ダイレクトパスの書き込みで待機した時間です。	I/O
direct path write total timeouts	ダイレクトパスの書き込みで発生したタイムアウトの回数です。	I/O
direct path write total waits	ダイレクトパスの書き込みで発生した待機回数の合計で	I/O

Metrics	Description	Category
	す。	
enq:_TX_-_row_lock_contention_time_waited	行ロック(row lock) 競争で待機した時間です。	I/O
enq:_TX_-_row_lock_contention_total_timeouts	行ロック(row lock) 競争で発生したタイムアウトの回数です。	I/O
enq:_TX_-_row_lock_contention_total_waits	行ロック(row lock) 競争で発生した、合計の待機回数です。	I/O
enqueue_waits	キューの読み取りが遅れてキューの変換または読み取り中に発生した待機回数の合計です。	I/O
execute_count	SQL文(statement)を実行した合計のコマンド数です。	Performance
fatal_count	クリティカルなエラーが発生した回数です。	Info
iowait_cpu	I/O待機状態で使用されたCPU時間の比率です。	Performance
latch free time waited	Latch free状態で待機した時間です。	I/O
latch free total timeouts	Latch free状態で発生したタイムアウトの回数です。	I/O
latch free total waits	Latch free状態で発生した待機回数の合計です。	I/O
library cache lock time waited	ライブラリキャッシュロックで待機した時間です。	I/O
library cache lock total timeouts	ライブラリキャッシュロックで発生したタイムアウトの回数です。	I/O
library cache lock total waits	ライブラリキャッシュロックで発生した待機回数の合計です。	I/O
library cache pin time waited	ライブラリキャッシュピンで待機した時間です。	I/O

Metrics	Description	Category
library cache pin total timeouts	ライブラリキャッシュピンで発生したタイムアウトの回数です。	I/O
library cache pin total waits	ライブラリキャッシュピンで発生した合計待機回数です。	I/O
lock_wait_sessions	現在の行ロック(row lock)を取得するために待機している数です。	Connection
logons_current	現在有効になっているログインの数です。	Connection
long_running_sessions	長時間実行中のセッションの数です。	Connection
memory	現在使用中のメモリ比率です。	MySQL Resource
opened_cursors_current	現在、開いているカーソル(cursor) の数です。	Connection
parse_count_(hard)	パースング呼び出しの合計数(実際のパースング)です。	Performance
parse_count_(total)	パースング呼び出しの合計数(hard, soft, describe)です。	Performance
physical_reads	ディスクで読み取ったデータブロックの数です。	I/O
physical_writes	データをディスクに記録したブロックの数です。 physical writes direct および physical writes from cache の合計です。	I/O
pq_sessions	並列クエリセッションの数です。	Connection
redo_entries	生成されたRedoエントリの数です。	I/O
rollback_changes_-_undo_records_applied	ロールバック変更により適用されたUndoレコードの数です。	I/O
session_logical_reads	メモリで読み込んだデータブロックの数です。 db block gets および consistent gets の合計です。	I/O

Metrics	Description	Category
sorts_(disk)	ディスクで発生したソート作業の数です。	I/O
sorts_(memory)	メモリで発生したソート作業の数です。	I/O
sys_cpu	セッションによって使用されたcpu timeの合計です。	Performance
total_elapse_time	データベースインスタンスが最後に再起動されてから、すべてのリクエストの累積合計経過時間です。	Performance
total_sessions	合計のセッション数です。	Connection
transaction_tables_consistent_reads_-_undo_records_applied	トランザクションテーブルで一貫した読み取り作業に適用されたUndoレコードの数です。	I/O
txn_sessions	トランザクションセッションの数です。	Connection
user_commits	トランザクション当たりに行われたユーザーコミットの作業数です。	I/O
user_cpu	ユーザー作業で使用されたCPUの比率(%)です。	Performance
user_rollback	ユーザーのロールバック作業の数です。	I/O
wait_sessions	イベント待機中のセッション数です。	Connection
warning_count	すべてのアラート数です。	Info

以下はエージェント設定で `ora_ratio` オプションを介して収集する指標の説明です。 エージェントオプション設定の詳細については、[次の文書](#)を参照してください。

Metrics	Description
buffer cache hit ratio	ディスクからデータを読み込まずにメモリから直接データを探せる回数。 この数値が高ければデータベースの性能が良いことを意味し、数値が低け

Metrics	Description
	ればディスクI/Oが増加して性能低下を招くことがあります。
soft parse ratio	SQL文が再利用される頻度。 Soft Parseは、すでにキャッシュにあるSQL文を再利用する場合です。この数値が高ければ、SQL文の再利用がうまくいっていることを意味します。
hard parse ratio	SQL文が最初から新しくパーシングされる頻度、 ハードパスは、SQL文を初めて実行するか、キャッシュに存在しない場合に発生します。この数値が高ければ、データベースのCPU使用率が増加し、性能が低下する可能性があります。
library cache hit ratio	Library CacheからSQL及びPL/SQL文を正常に検索する割合。 この数値が高ければ、SQL文とPL/SQLコードがキャッシュで効果的に再利用されていることを意味します。
library cache pin hit ratio	Library Cache Pinリクエストが成功した割合。 Library Cache Pinは、実行中のSQL文が他のセッションで変更されないように保護する役割をします。この数値が高ければ、これらの保護が効果的に行われていることを示します。
library cache get hit ratio	Library Cacheの個体を正常に取得する割合。 この数値が高ければ、SQL文とPL/SQLコードがライブラリキャッシュで効果的に再利用されていることを意味します。
latch hit ratio	Latchリクエストが成功した割合。 Latchは、データベースの共有メモリアクセスを制御するためのロックメカニズムです。この数値が高ければ、Latchの待ち時間が少なく、システムの同時性と性能が優れていることを意味します。
shared pool free mem ratio	共有プールの空きメモリ比率 共有プールはSQL文、PL/SQLコード、及びデータディクショナリーキャッシュを保存するメモリ領域です。適切な空きメモリ比率を維持することで、共有プールの効率的な運用と性能最適化を保証できます。

db_oracle_dma_tablespace

Oracleデータベースのテーブルスペース使用状況を1時間おきに収集したカテゴリです。このカテゴリのメトリクスは、テーブルスペースの状態と容量をモニタリングし、ストレージリソースを効率的に管理し、データベースの性能の問題を事前に予防するのに役立ちます。

Tags

次は、`db_oracle_dma_tablespace` カテゴリに収集されるタグに関する案内です。

Tag	Description	Remarks
<code>con_name</code>	接続されたコンテナのデータベース名です。マルチテナント環境で、各Pluggable Database(PDB)を区分するのに役立ちます。	-
<code>datfile</code>	各テーブルスペースに割り当てられたデータファイルのパスとファイル名です。	-
<code>oname</code>	エージェント名です。	一意の値
<code>tablespace</code>	テーブルスペースの名前です。	-

Field

次は、`db_oracle_dma_tablespace` カテゴリに収集されるフィールドの案内です。

Metrics	Description
<code>autoextensible</code>	<code>YES</code> または <code>NO</code> で、データファイルが自動拡張の可能有無を示します。
<code>increment_by</code>	データファイルが自動拡張された時に一度に拡張されるブロックの数です。
<code>maxsize</code>	データファイルが自動的に拡張される時に設定された最大サイズです。
<code>size</code>	テーブルスペースの現在割り当てられている容量の合計です。

Metrics	Description
<code>used</code>	現在使用中のテーブルスペースのサイズです。

XOS関連メトリクス

XOSエージェントから収集されるメトリクスについての説明です。

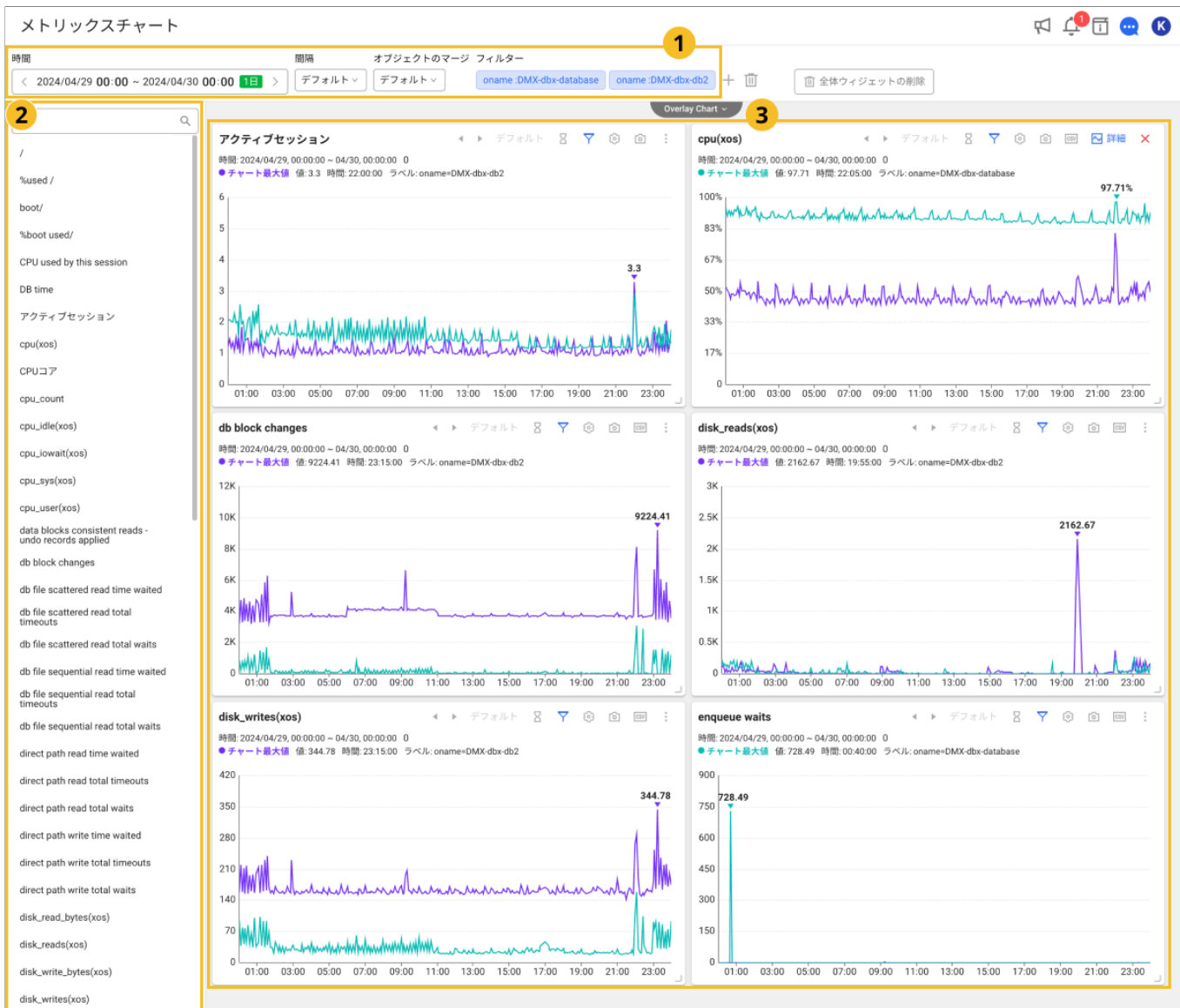
Metric	Unit	Description
<code>cpu(xos)</code>	pct	CPU使用率 <code>cpu_user</code> + <code>cpu_sys</code> + <code>cpu_iowait</code> の合計
<code>cpu_idle(xos)</code>	pct	CPUが遊休状態の時間
<code>cpu_iowait(xos)</code>	pct	CPUがI/O作業を待機した時間
<code>cpu_sys(xos)</code>	pct	CPUがカーネルコードの実行に使用された時間
<code>cpu_user(xos)</code>	pct	CPUがユーザーコードの実行に使用された時間
<code>disk_read_bytes(xos)</code>	バイト	ディスクから読み取ったデータ量
<code>disk_reads(xos)</code>	block	ディスクで読み取られたブロック数
<code>disk_write_bytes(xos)</code>	バイト	ディスクに記録されたデータ量
<code>disk_writes(xos)</code>	block	ディスクに記録されたブロック数
<code>mem(xos)</code>	pct	メモリ使用率
<code>mem_available(xos)</code>	MB	使用可能なメモリサイズ
<code>mem_buffcache(xos)</code>	MB	buff/cache用途に使用中のメモリサイズ

Metric	Unit	Description
<code>mem_free(xos)</code>	MB	使用していないメモリサイズ
<code>mem_swapfree(xos)</code>	MB	未使用のスワップ領域サイズ スワップはディスク領域を使用するため、使用時に性能が低下する可能性がある
<code>mem_swaptotal(xos)</code>	MB	全体スワップ(Swap)サイズ 物理メモリ不足時に使用
<code>mem_total(xos)</code>	MB	全体メモリのサイズ
<code>mem_used(xos)</code>	MB	使用中のメモリサイズ
<code>net_rcv_bytes(xos)</code>	バイト	ネットワークから受信したデータ量
<code>net_rcv_packets(xos)</code>	count	ネットワークから受信したパケット数
<code>net_send_bytes(xos)</code>	バイト	ネットワークで送信したデータ量
<code>net_send_packets(xos)</code>	count	ネットワークで送信したパケット数

メトリクスチャート

🔍 サイトマップ > 分析 > メトリクスチャート Old

メトリクスチャートメニューでは、モニタリング対象から収集されたメトリクスデータを次のようにチャートで照会できます。時間と指標の指定が必須です。



上部オプション

① 領域のメトリクスチャートの上部オプションからチャートの時間範囲とモニタリング対象エージェントを指定できます。

- **時間**：時間の範囲でX軸の開始と終了を指定できます。
- **間隔**：時間間隔でX軸データの間隔を指定できます。
- **時間マージ**：データマージ方式の1つを選んで、指定した時間間隔内のデータをマージできます。

例) 平均は、1時間のデータ平均値を意味します。

-  **エージェント**：照会するエージェントを指定できます。指定しない場合は、すべてを対象に照会します。

指標一覧

② 領域は、オプションを検索する指標のリストです。まず、**カテゴリ**を選択します。**カテゴリ**を選択すると、そのカテゴリに含まれる指標が表示されます。目的の指標を選択してください。**カテゴリ**と指標を選択すると、① 領域のトップメニューから指定した時間範囲のデータに基づいて③ 領域のチャートウィジェットからデータを照会できます。

① データ結合

データのマージは、**オブジェクトのマージ**と**時間マージ**方法を提供します。

- **時間マージ**は、ソースデータからフィールド値が同じデータを一定の間隔でマージします。
- **オブジェクトのマージ**は、異なるフィールド値を持つデータの中からタグが一致する場合、そのデータをマージします。

チャートウィジェット

③ 領域のチャートウィジェットの左上で指標名を確認できます。チャートウィジェットの右上で次のオプションを確認できます。

アクティブセッション



デフォルト






詳細


時間: 2024/04/30, 00:00:00 ~ 05/01, 00:00:00 0

● **チャート最大値** 値: 2.43 時間: 07:00:00 ラベル: oname=DMX-dbx-database

- タイムシフト：◀ ▶ 左矢印または右矢印ボタンを使用して、選択した時間範囲だけ-1または、+1ずつ移動することができます。

例えば、時間範囲が2月13日00:00~2月14日00:00の場合、◀ 左矢印を選択すると、2月12日00:00~2月13日00:00までのデータが表示できます。

- 間隔/時間のマージ：① トップメニューで指定した間隔と時間のマージを修正できます。
- モニタリング対象：▼ アイコンを選択して、モニタリング対象を指定できます。選択しない場合は、すべて表示されます。
- 時間比較：📊 アイコンを選択すると、同じ指標の以前の時間帯の推移を比較できます。
- スナップショット：📷 アイコンを選択すると、ウィジェットのオプションを除くチャートのスナップショットを取得できます。
- CSV：📄 アイコンを選択して、チャートを描画したデータをCSVファイルとしてダウンロードできます。
- 詳細：🔍 アイコンを選択して詳細を表示します。モニタリング対象が多い場合、モニタリング対象の指標推移を特定して確認できるため便利です。

❗ メトリクスチャートウィジェットの上部にオプションが表示されない場合は、⋮ アイコンを選択します。

ウィジェットを削除する

画面に配置したすべてのウィジェットを削除するには、🗑 全体ウィジェットの削除ボタンを選択します。

メトリクス照会

ホーム画面 > プロジェクト選択 > 分析 > メトリクス照会

メトリクス照会メニューからタグベースで特定のメトリクスを照会できます。

メトリクス照会

時間選択

< 2023/08/17 08:04 ~ 2023/08/17 09:04 1時 >

最大カウント

100

* カテゴリ

app_counter

🔍

タグ 全体

container

containerKey

host_ip

okindName

oname

onodeName

pid

type

8/8

フィルター

▼ タグ値でフィルタリング

フィールド 全体

_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

metering

resp10

resp95

resp_time

sql_count

sql_error

sql_fetch_count

sql_fetch_time

sql_time

tps

tx_count

tx_dbc_time

tx_error

tx_http_time

tx_sql_time

tx_time

tx_time_sqr_sum

5/29

🔍

📄 CSV

	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 08:04:30	-877561626	agent.service	1395943798	10.21.1.26	demo-okind-1	demo-8101	node-1	3284967	java	13	19.554815	9.995699	19.008858	1
2	2023-08-17 08:04:30	1387800924	agent.service	1395943798	10.21.1.26	demo-okind-0	demo-8102	node-0	3285018	java	10	21.37008	5.108556	19.723042	1

1 時間とカテゴリの選択

① 領域でメトリクスが収集された時間選択と最大カウントおよびカテゴリを指定できます。時間選択とカテゴリは、必ず指定してください。

- 時間選択：メトリクスが収集された時間を指定して照会できます。既定値は 1時間 です。基本オプションとして提供する照会時間以外に、ユーザーが時間選択タブを選択して日付と時間を指定できます。

① 時間選択オプションの利用方法の詳細については。

- 最大カウント：③ 一覧から参照するメトリクスの最大数を指定できます。10、50、100、200、300、1000、2000、3000 個まで設定できます。
- カテゴリ：関連指標の分類単位です。カテゴリタブを選択し、カテゴリを指定できます。
- 🔄 カテゴリ、タグ、フィールドオプションの更新：カテゴリ、タグおよびフィールドオプションを再読み込みできます。

2 タグとフィールドの選択

2 領域でタグとフィールドを選択します。ユーザーが個別に指定しない場合、デフォルト設定が完全な選択になります。

- **タグ**：収集された対象を区分できる固有情報データです。
- **フィールド**：モニタリング対象から収集された指標です。
- **フィルター**：▼タグ値でフィルタリングボタンを選択し、タグ値を設定してフィルタリングできます。

例) oname の値を demo-8101 に設定すると、フィルタリングしたデータを表示することができます。

- 🔍 **検索**：条件を設定して検索アイコンを選択すると、3 領域でそのメトリクスの原本データを検索できます。
- 📄 **CSV**：メトリクスの原本データをCSVファイルとしてダウンロードできます。

3 メトリクステーブル

収集されるメトリクスを事前に特定できないため、収集されるすべてのメトリクスの原本データを確認することが重要です。上記の条件領域で目的の条件を設定した後 3 領域で、そのメトリクスの原本データをテーブル形式で検索できます。ユーザーがタグとフィールドの各条件を指定することによって、テーブルのカラムが変更されます。

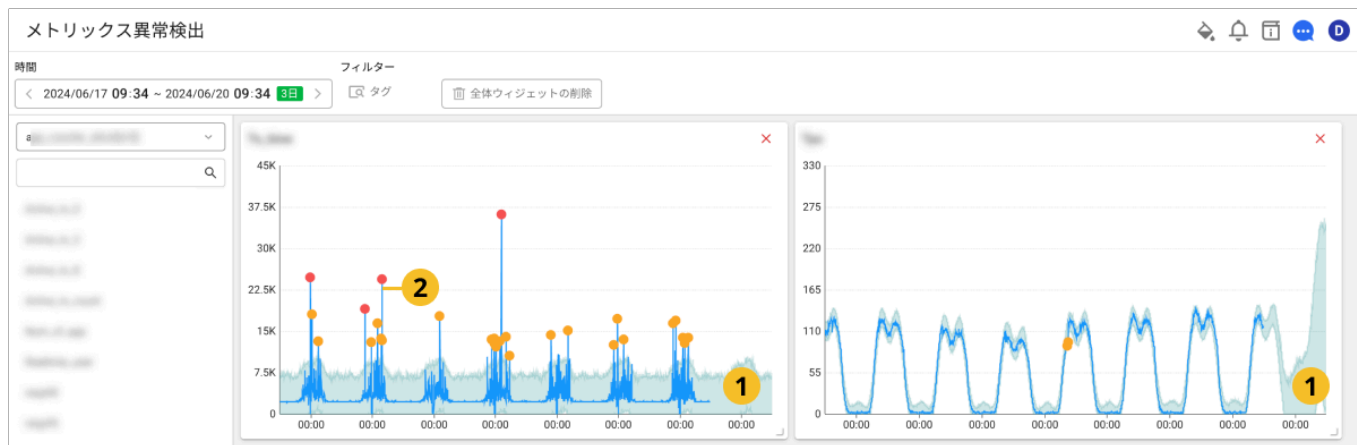
- ❗ • メトリクスの参照時、**時間選択**と**カテゴリ**を指定する必要があります。
- メトリクス参照時、**タグとフィールド**の指定はオプションです。

メトリクス異常検出

さまざまなメトリクスのパターンをAIが学習した予想パターンと比較できます。予想パターンを外れた異常探知をグラフチャートで確認できます。過去のデータに基づいて繰り返されるパターンを確認し、今後の指標値の予測に活用できます。

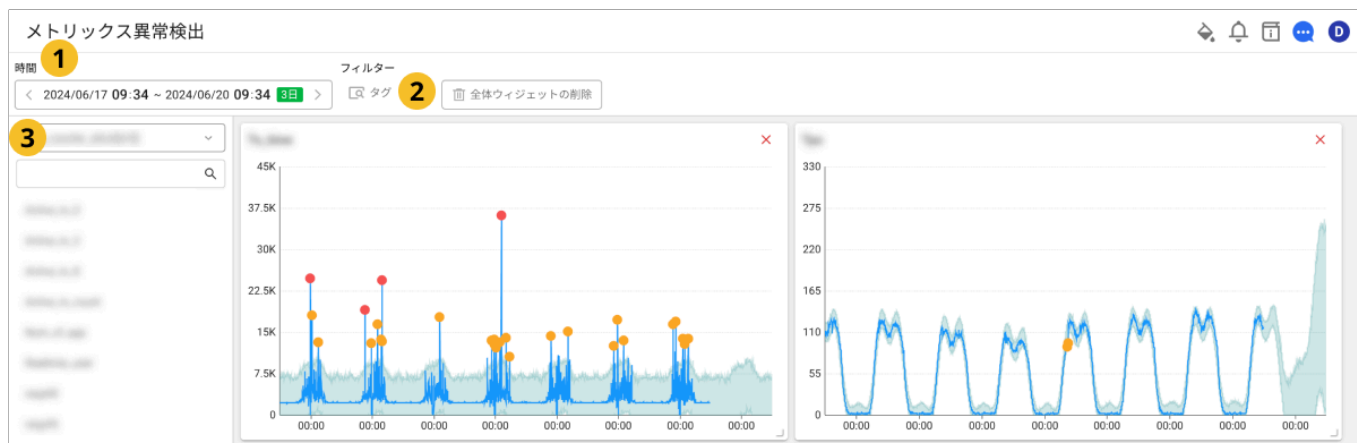
- ❗ 画面に配置したウィジェットは、別のメニューに移動すると保存されずに初期化されます。
- パターン分析と異常値検出機能を除くと分析>メトリクスチャートメニューに似ています。
- 異常値(Anomaly Detection)アラート通知機能の技術的根拠は、メトリクス異常値です。異常値アラート通知設定の詳細については、[次の文書](#)を参照してください。

ウィジェットを確認する



- ① 明るい色のグラフ領域は、AIによって分析した予想されるパターンです。
- ② 青色グラフは、プロジェクトのメトリクスの推移です。
- AIによって分析された予想パターンから外れるとオレンジ色と、赤色のステップでグラフにドットが表示されます。予想されるパターンの範囲から大幅に外れた値を赤色として表示します。

ウィジェットを配置する



1. ① 時間で、目的の時間間隔を設定してください。最大1か月間隔まで設定できます。
2. ② フィルターでメトリクスの範囲を選択します。
3. 以下の一覧でモニタリングするメトリクスを選択します。

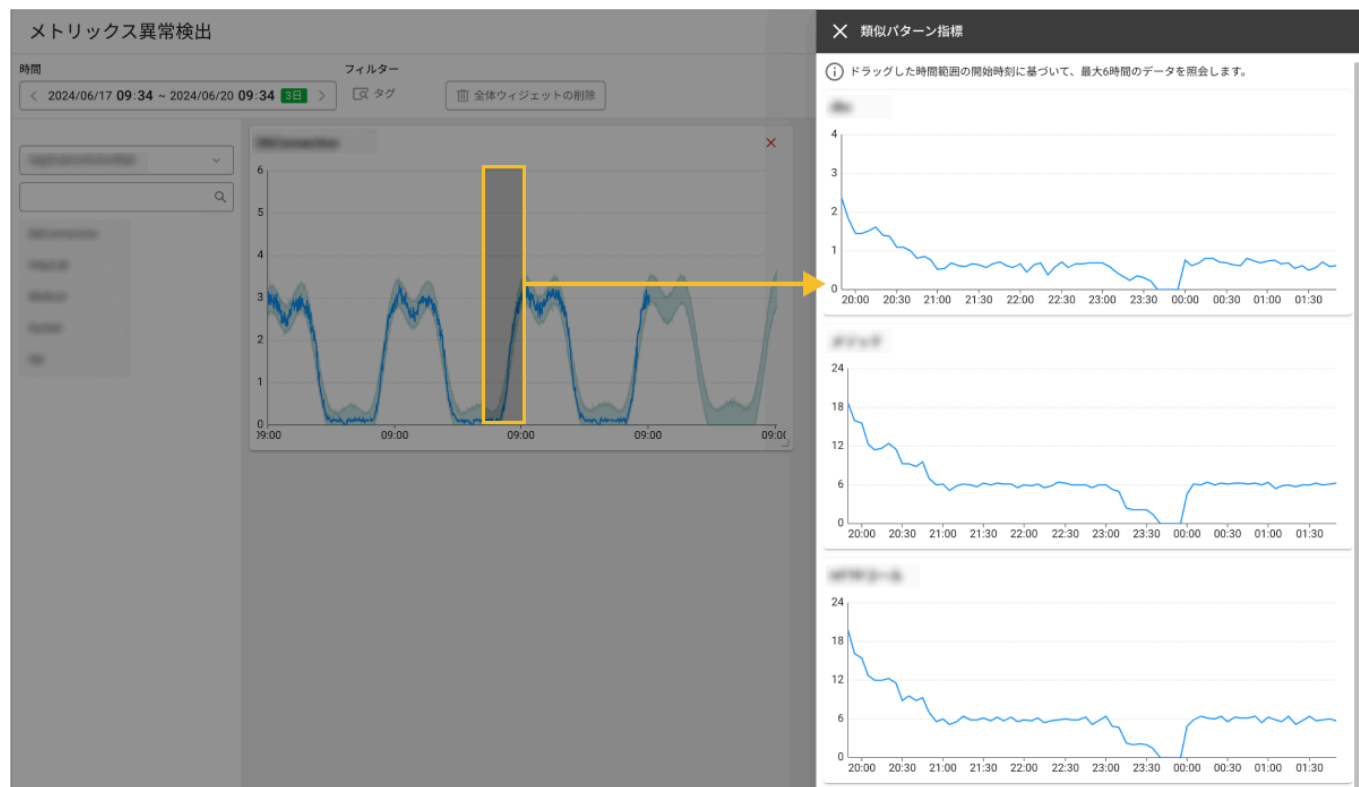
選択したメトリクスを画面の右側に配置します。

- ① • モニタリング対象を選択してメトリクスを絞り込んで確認するには、② タグを選択します。画面右側にタグ選択リストが表示されます。目的の項目を選択し、ウィジェットを追加します。
- 画面に配置したウィジェットは、時間またはタグ値を変更してもチャートに反映されません。
- 画面に配置したウィジェットを削除するには、ウィジェット右上の✕ボタンを選択します。
- ウィジェットの左上を選択した状態でドラッグすると、ウィジェットの位置を変更できます。
- ウィジェットの右下を選択した状態でドラッグすると、ウィジェットのサイズを変更できます。

ウィジェットを削除する

画面に配置したすべてのウィジェットを削除するには、 全体ウィジェットの削除ボタンを選択します。

関連メトリクスを確認する



ウィジェットでチャートの一部の領域をドラッグします。画面の右側に**類似パターン指標**画面が表示されます。ドラッグした時間範囲の開始時間に基づいて、最大6時間の関連メトリクスを照会できます。

テーブルスペースサイズ

テーブルスペースサイズは、データベーステーブルスペースの使用状況を監視・管理できる画面です。テーブルスペースごとのサイズ変化の傾向をラインチャートで簡潔に確認し、日次増減と**Remaining Day**（残余日数）によって空き容量不足を事前に検知できます。

- テーブルスペースサイズ推移ラインチャートを通じて、主要テーブルスペースの容量変化の傾向を把握できます。
- ディスク使用量では、マウントポイントごとのディスク使用状況と残余日数を確認できます。XOSエージェントがインストールされている場合に収集できます。
- テーブルスペースサイズ詳細では、個々のテーブルスペースの最新使用量、日次増減、残余日数を分析できます。

事前設定

テーブルスペースサイズ画面にデータを表示するには、[DB設定](#)と[エージェント設定](#)が必要です。

DB設定

エージェントがテーブルスペースおよびセグメント情報を照会できるよう、モニタリングアカウントに権限を付与します。

```
GRANT SELECT ANYDICTIONARY TO whatap;
```

または、個別ビューに対して権限を付与します。

```
GRANT SELECT ON dba_segments TO whatap;
```

エージェント設定

セグメントサイズ（TOP 50）データを収集するための設定です。以下の条件に基づき、エージェントが1日1回、設定した時刻にセグメントサイズの収集を実行します。 `segments_min_size` などの条件によってデータが収集されない場合があります。この設定を行わないと、セグメントサイズ情報を確認できません。

設定項目	タイプ	デフォルト値	説明
segments	boolean	true	セグメント情報収集の有効化
segments_hour	int	5	セグメント情報収集時刻。デフォルトは5で、午前5時に収集を開始します。
segments_min_size	int	10	収集対象の最小セグメントサイズ（MB）
segments_exclude_sys	int	1	sysアカウントのセグメントを除外するか否か（1：除外、0：含む）
segments_max_count	int	50	テーブルスペースごとに収集するセグメント数

① テーブルスペースサイズ収集オプション

- ① DB環境によっては、テーブルスペースサイズの収集時にパフォーマンスの問題が発生する場合があります。以下のオプションで収集方式と周期を調整できます。

設定項目	タイプ	デフォルト値	説明
tablespace	int	1	テーブルスペースサイズの収集方式を選択 - 0：収集を無効化 - 1：dba_free_spaceベースの照会（最も一般的） - 2：LMT（Locally Managed Tablespace）向け free_blocks ベースの照会 - 3：tablespace_usage_metrics ビューベースの照会（データファイル別照会不可、Auto Ext情報なし）
tablespace_split	boolean	false	tablespace オプションが1の場合に適用 - true：最初にデータファイルのリストを取得し、ファイルごとにfree spaceを個別照会して負荷を分散。大規模なデータファイル環境で収集クエリの負荷が高い場合に使用。



設定項目	タイプ	デフォルト値	説明
tablespace_interval	int	60	テーブルスペースサイズの収集周期（分） - デフォルト値：60分（1時間）

- ✓ DBサイズが大きく収集がうまくいかない場合は、tablespace=1 と tablespace_split=true を組み合わせて負荷を分散するか、tablespace=2 に変更してお試しください。

テーブルスペースサイズ

基本オプション

テーブルスペースサイズ画面の上部フィルターエリアで、照会する時間および対象データベースなどを選択できます。

- ・ **時間**：照会する時間範囲を選択。最大3週間まで照会可能で、3日以上以上の照会は1日平均で表示されます。
- ・ **インスタンス**：照会対象のDBインスタンスを選択します。
- ・ **データ単位**：MiB/GiB/TiB から選択。すべてのテーブルのサイズ値が選択した単位で統一して表示されます。
- ・ **データベース**：照会対象のデータベースを選択します。

テーブルスペースサイズ推移

テーブルスペースごとのサイズ変化をラインチャートで表示します。各テーブルスペースは色別に区分され、下部の凡例で確認できます。

- ・ マウスオーバーで時間帯ごとの詳細数値を確認できます。

項目	説明
Top 5 / 10 / 15 / 20	チャートに表示する上位テーブルスペース数を選択

項目	説明
Size / Used	Size（全体サイズ）またはUsed（使用量）基準に切り替え

ディスク使用量

❗ XOSエージェントがインストールされている場合に収集できます。

マウントポイントごとのディスク使用状況を表示します。**Latest Data**エリアには、時間選択器の照会範囲内で最後に収集されたデータを表示します。その時点のサイズ、Usage(%)、Increase AVG、Remaining Dayを確認できます。

Latest Data

カラム	説明
Mount Point	ディスクマウントパス
Size	ディスクの全体サイズ
Used	使用中のサイズ
Free Size	残りの空き容量
Usage(%)	使用率（プログレスバーで視覚化）
Increase AVG	照会期間中の日次増加量の平均値
Remaining Day	現在の増加速度基準のディスク利用可能残余日数

日次カラム

照会期間の各日付ごとに Size と前日比の増減（Inc）を表示します。

- 検索：LabelまたはPathでフィルタリングできます。

- **CSV**：CSV形式でエクスポートできます。

ASM環境の場合、ディスク使用量タブとASMディスク使用量タブが併せて表示されます。

ASMディスク使用量

日次カラム、検索、CSVの機能はディスク使用量と同様です。

カラム	説明
Label	ASMディスクラベル
Path	ASMディスクパス
Size	ディスクの全体サイズ
Used	使用中のサイズ
Free Size	残りの空き容量
Usage(%)	使用率（プログレスバーで視覚化）
Increase AVG	照会期間中の日次増加量の平均値
Remaining Day	現在の増加速度基準のディスク利用可能残余日数

テーブルスペースサイズ詳細

個々のテーブルスペースの詳細な使用量を表示します。最上部の**TOTAL**行で全体の合計を確認できます。**Latest Data** エリアには、時◆◆選択器の照会範囲内で最後に収集されたデータを表示します。その時点のサイズ、Usage(%)、Increase AVG、Remaining Dayを確認できます。

Latest Data

カラム	説明
Container	コンテナ（PDB）名。CDB環境でのみ表示され、Non-CDB環境ではこのカラムは非表示になります。
Tablespace	テーブルスペース名
Auto Ext	自動拡張の有無（YES/NO）。設定でカラムの表示/非表示を選択できます。
Inc By	自動拡張時の増加単位。設定でカラムの表示/非表示を選択できます。
Max Size	最大サイズ制限。設定でカラムの表示/非表示を選択できます。
Total Size	テーブルスペースの全体サイズ
Used Size	使用中のサイズ
Free Size	残りの空き容量
Usage(%)	使用率（プログレスバーで視覚化）
Increase AVG	照会期間中の日次増加量の平均値
Remaining Day	現在の増加速度基準の残余日数。 $\text{Free Size} \div \text{Increase AVG}$ で計算します。

日次カラム

照会期間の各日付ごとに **Size** と前日比の増減（ **Inc** ）を表示します。増加は正の値、減少は負の値で表示され、色で区分されます。

追加機能

- 🔍 虫眼鏡アイコン：テーブルスペースのセグメントサイズ（TOP 50）ポップアップを表示します。

- ▼ **展開**：テーブルスペースに属するデータファイルのリストを展開して確認できます。
- **検索**：コンテナ、テーブルスペースでフィルタリングできます。
- **CSVダウンロード**：CSV形式でエクスポートできます。ダウンロード時に**Tablespace**単位または**Datafile**単位を選択できます。

テーブル設定

右上の⚙️ **設定**ボタンをクリックすると、テーブルの表示オプションを設定できます。

項目	説明
Column: Auto Ext / Inc By / Max Size	Auto Ext、Inc By、Max Size カラムの表示有無をトグルします。
Warning(%)	警告レベルの使用率(%)。超過時はオレンジ色で表示。デフォルト値：80
Critical(%)	危険レベルの使用率(%)。超過時は赤色で表示。デフォルト値：100
設定の初期化	すべての設定をデフォルト値に初期化します。

セグメントサイズ (TOP 50)

テーブルスペース行の🔍 **虫眼鏡アイコン**をクリックすると、該当テーブルスペースで多くの容量を占める上位50件のセグメントをポップアップで表示します。

ポップアップ右上の⬇️ **ダウンロード**ボタンをクリックすると、ポップアップに表示された一覧をCSVファイルとしてエクスポートできます。画面に表示されているカラム構成と表示形式がそのままファイルに反映され、保存されるファイル名は **table_size_top50.csv** です。

- ⚠️ エージェント設定で segments オプションが有効になっている必要があります。エージェントは1日1回セグメントサイズを収集し、設定しない場合はデータを確認できません。

カラム	説明
テーブルスペース名	セグメントが属するテーブルスペース

カラム	説明
オーナー	セグメントを所有するスキーマ
セグメント名	セグメント名。クリックすると Object Detail ポップアップでオブジェクトの詳細情報を照会できます。
セグメントタイプ	TABLE、INDEX、LOBSEGMENTなどのセグメント種別
パーティション名	パーティション名（パーティションテーブルの場合）
サイズ	セグメントサイズ。日付ごとのカラムでサイズ変化を確認できます。

Object Detail

セグメントサイズ（TOP 50）ポップアップでセグメント名をクリックすると、エージェントにリアルタイムの照会リクエストを送信し、該当オブジェクトの詳細情報を表示します。

- **オブジェクト基本情報**：owner、table_name、num_rows、blocks、avg_row_len、tablespace_name、partitioned、last_analyzed、compression、statusなど
- **Column/Index**：カラムリスト（column_id、column_name、nullable、data_type、data_default、num_distinct、num_buckets）とインデックス情報を確認できます。

データ解釈ガイド

主要確認ポイント

- **Usage(%)** が高いテーブルスペース
空き容量不足のリスク → 不要なデータの整理またはテーブルスペースの拡張を検討
- **Increase AVG** が高いテーブルスペース
急速に増加中 → **Remaining Day** を確認して事前に対応
- **Remaining Day** が短い場合

現在の増加速度基準で、まもなく利用可能容量が枯渇 → 即時対応が必要

- **Auto Ext がNOのテーブルスペース**

自動拡張が無効化されており、容量枯渇時にエラーが発生する可能性あり → Usageが高い場合は手動拡張または Auto Extの有効化を検討

- **日次の Inc 値が急激に変動**

特定の日に大量のデータの投入または削除が発生 → 原因を確認

- **ディスクの Usage(%) とテーブルスペースの Usage(%) を合わせて確認**

テーブルスペースに空きがあっても、ディスクが満杯の場合は拡張不可

実践活用シナリオ

シナリオ1. 空き容量不足の事前検知

1. テーブルスペースサイズ詳細で Usage(%) を基準に降順ソート
2. 使用率が高いテーブルスペースの Remaining Day を確認
3. 残余地数が少ない場合は、テーブルスペースの拡張または不要なデータの整理を実施

シナリオ2. 容量急増の原因分析

1. テーブルスペースサイズ推移チャートで急激に増加しているテーブルスペースを確認
2. 該当テーブルスペースの日次 Inc カラムで増加が発生した日付を確認
3. 該当テーブルスペースの 🔍 虫眼鏡アイコンをクリックして、多くの容量を占めているセグメントを分析
4. セグメントの詳細を確認し、不要なインデックスなどの有無やデータサイズを削減する方法を検討。対応策がない場合はディスク増設を検討

シナリオ3. ディスク容量管理

1. ディスク使用量で Usage(%) が高いマウントポイントを確認
2. Remaining Day でディスクフルの予想時期を把握

3. テーブルスペースサイズ詳細と比較して、どのテーブルスペースがディスクを多く使用しているかを確認

シナリオ4. 自動拡張設定の点検

1. 設定でAuto Ext / Inc By / Max Sizeカラムを有効化
2. Auto Ext がNOのテーブルスペースのうち、Usage(%) が高いものを確認
3. 必要に応じて自動拡張を有効化、または手動でテーブルスペースを拡張

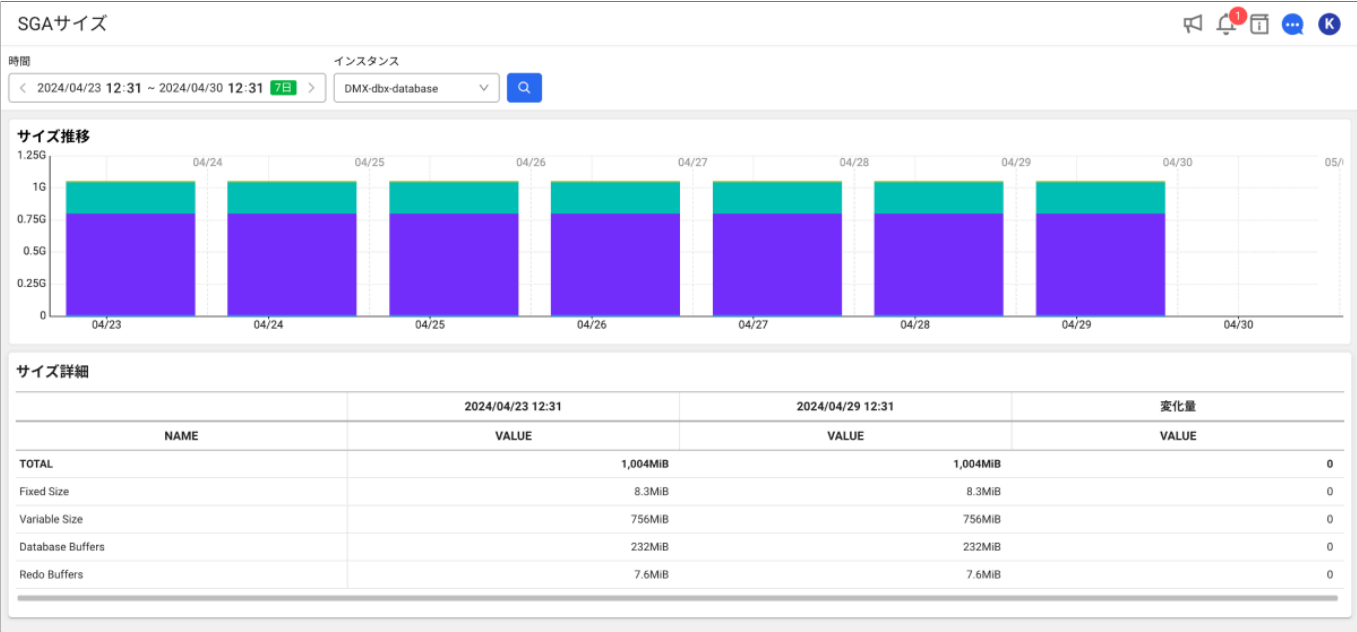
SGAサイズ

ホーム画面 > プロジェクト選択 > 統計/レポート > SGAサイズ

サーバーのメモリ領域であるSystem Global Area(SGA)の要約情報を時間ごとに保存し、時間別変化量をモニタリングできます。

- 性能最適化: SGA内の複数のメモリ領域の大きさおよび使用率をモニタリングすることで、性能の問題を事前に予防し、メモリ資源を最適に管理できます。
- ボトルネック検知: 各メモリ領域で発生するボトルネックを早期に特定し、システムの可用性と安定性を向上できます。
- リソース管理: メモリ資源の適正な割り当てを確認することで、不要なメモリの使用を減らし、メモリ関連資源を効率的に配分できます。
- 問題解決: SGA領域のどの部分が過剰に使用されているかを確認することで、性能低下の原因を迅速に解決できます。

基本的な使い方



1. 時間照会する時間を設定します。
2. インスタンスでトレンド情報を表示するインスタンス名を選択します。
3.  ボタンを選択します。

 • 照会時間は最大3週間まで選択できます。

データベースパラメータ

ホーム画面 > プロジェクト選択 > 分析 > パラメータ

データベース設定値の参照機能を提供し、過去の日付の時点でのデータベース設定を表示することができます。比較機能を使用すると、データベースのパラメータ値を日付別に比較できます。デフォルトでは、現在時刻のデータが表示されます。

データベースにパラメータ値が原因で問題が発生した場合、データベースパラメータ比較機能を使用すると、すばやく原因を特定できます。

パラメータ				
基準日	比較日	インスタンス		
2024/06/14	2024/06/13	DMX-dbx-database		
Key	2024/06/14 13:56:16 本日		2024/06/13 00:00:00	
	Value		Value	
shrd_dupl_table_refresh_rate	60		60	
pdb_lockdown				
default_sharing	metadata		metadata	
containers_parallel_degree	65535		65535	
disable_pdb_feature	0		0	
remote_recovery_file_dest				
cdb_cluster_name	ORA122		ORA122	
cdb_cluster	FALSE		FALSE	
max_pdb	4098		4098	
target_pdb	1		1	
common_user_prefix	C##		C##	
noncdb_compatible	FALSE		FALSE	
pdb_file_name_convert				
pdb_os_credential				
enable_pluggable_database	TRUE		TRUE	

1. 基準日と比較日を設定します。
2. インスタンス項目で目的のインスタンスを選択します。
3.  ボタンをクリックします。

現在と選択した過去の日付との変更点を一目で確認する場合は、**変更を表示**ボタンを選択します。

パラメータ			    	
基準日	比較日	インスタンス		
2024/06/14 	2024/06/13 	DMX-dbx-database 		変更を表示
		2024/06/14 13:56:16 	2024/06/13 00:00:00	
Key	Value		Value	
resource_manager_plan	SCHEDULER[0x4AC6].DEFAULT_MAINTENANCE_PLAN		SCHEDULER[0x4AC5].DEFAULT_MAINTENANCE_PLAN	

レポート

ホーム画面 > プロジェクト選択 > 統計/レポート > レポート

画面の上に**レポート**で▽ボタンをクリックすると、日刊、週刊、月刊に分類された各種分析レポートを確認できます。希望するレポート様式を選択し、照会したい日付と時間、フィルターオプションを設定してください。 ボタンを選択すると、選択したレポートを確認できます。

レポートは、個々のプロジェクトのサービスの使用状況と障害の発生履歴を報告する文書です。サービスモニタリング担当者は、レポートを通じて関連部署の担当者とステータスを共有します。

モニタリングデータ分析は、サービスの改善方向を定める指標となるため、極めて重要です。ただし、複数のダッシュボードからデータを収集して文書化するには煩雑な場合があります。

WhaTapの**レポートメニュー**は、**レポート作成業務の自動化**をサポートします。毎週報告書を作成すること、決まった時間に報告書を共有すること、様々な書式を管理することなど、すべて**レポートメニュー**から解決できます。

✔ レポートダウンロード

レポートのダウンロード、印刷またはメール送信の予約を希望する場合は ダウンロードボタンをクリックしてください。html形式でダウンロードできます。

レポートの種類

基本レポートは、大手企業、公共機関やITサービス企業などで実際に使用しているフォームです。ご要望のフォームを作成したい場合は、support@whatap.ioまでお問い合わせください。要望されたレポートは、**統計/レポート > レポート一覧**に追加されます。

メール送信予約

定期的にレポートをメールで受信する場合は、**レポートのメール送信予約**を選択します。出社直後行っていたサービスのチェック作業をメール確認で代わりにできます。

レポートのメール送信予約

×

① 登録した翌日の04~06時にレポートが送信されます。

② 最大5件までのレポート送信が予約できます。

レポートタイプ

レポート時刻

00:00

~

23:59

送信曜日

☒ Mon

☒ Tue

☒ Wed

☒ Thu

☒ Fri

☐ Sat

☐ Sun

メール

メールを入力してからEnterする

×

+ 保存

目次レポート：データベース

日次データベースレポートは、1日あたりのデータベースのパフォーマンスを数値とチャートで確認できるレポートです。レポート作成時に必要に応じて開始日、開始時間、終了時間を設定することができます。特定のインスタンスのみ選択して表示することができます。すべてを選択すると、すべてのインスタンスを一度に表示します。

性能推移チャート

データベースインスタンスごとに指標チャートを確認できます。各指標の最大値を円形で表示し、最大値が発生した時間をまとめて表示します。

- ① チャートで表現される指標は、Config設定によって異なる場合があります。
- 各チャートの右上にあるCSVボタンを使用して、データをエクセルファイルとしてダウンロードすることができます。

SQL TOP 10

検索期間中に全体または選択したインスタンスが実行したSQLクエリ統計を確認できます。クエリ実行時間の最大値を降順にソートし、上位10項目のみ表示します。

週次レポート：データベース

週次データベースレポートは、1週間のデータベースのパフォーマンスを数値とチャートで確認できるレポートです。開始日を直接設定することができ、参照日の7日前から表示できます。

性能推移チャート

データベースインスタンスごとに、週間指標を数値およびチャートで比較できます。検索期間の最大値を円形で表示し、その数値が発生した時間も表示されます。

- ① • チャートで表現される指標は、Config設定によって異なる場合があります。
- 各チャートの右上にある[CSV](#)ボタンを使用して、データをエクセルファイルとしてダウンロードすることができます。

SQL TOP 10

検索期間に全体または選択したインスタンスが実行したSQLクエリ統計を確認できます。クエリ実行時間の最大値(elapse max)を降順にソートし、上位10項目のみを表示します。

月次レポート：データベース

月間のデータベースの性能を数値とチャートで確認できるレポートです。開始日を設定でき、開始日から30日まで表示できます。月間の各指標別の変化量をインスタンス別に確認することができ、検索期間中に実行した上位10項目のSQLクエリ統計も確認できます。

SQL統計

ホーム画面 > プロジェクト選択 > 統計/レポート > SQL統計

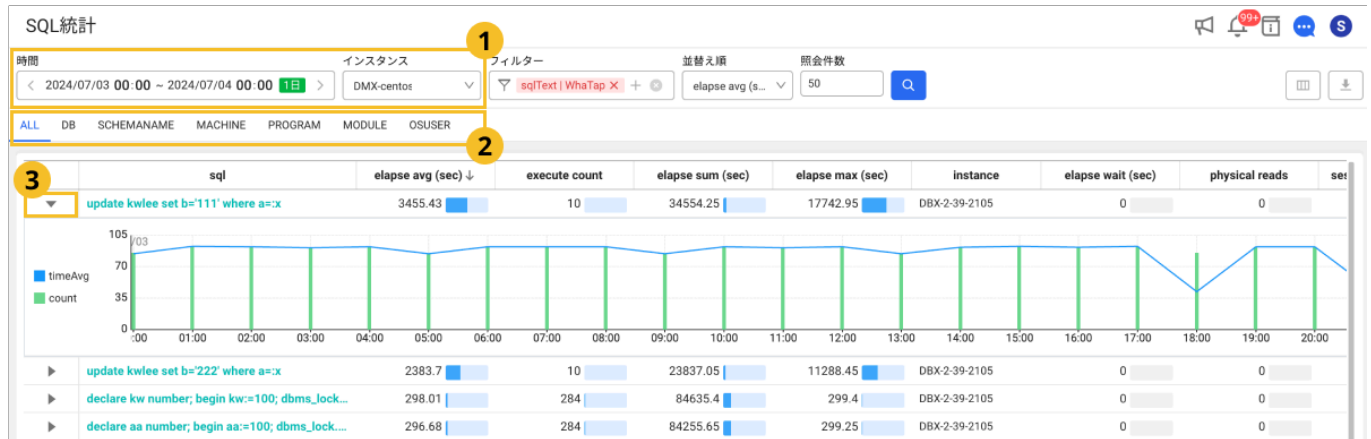
エージェントが収集したアクティブセッション内のSQLを実行回数、実行時間、待機時間などを基準に統計を算出し、表示期間中のTop SQL一覧を提供します。平均実行時間とピーク実行時間は、頻繁に実行されるSQL性能を分析できる根拠を提供します。

パフォーマンス測定でよくで使う分析カテゴリ(DB、USERNAME、MACHINE、PROGRAM)ごとにTop SQLを照会できます。主な照会基準は次のとおりです。

- **execute count**: SQLの実行回数
- **elapsed sum (sec)**: SQLの実行時間(秒) 合計
- **elapsed avg (sec)**: SQLの平均実行時間(秒)で**elapsed sum/execute count**の結果値
- **elapsed max (sec)**: SQLの最大実行時間(秒)
- **elapsed wait (sec)**: SQLの待機時間合計(秒)
- **instance**: インスタンス名
- **physical reads**: 該当SQLがディスクからデータを読み込んだ回数
- **session logical reads**: 該当SQLがメモリ内でデータを読み込んだ回数
- **cpu time**: 該当SQLの実行中に使用されたCPU時間
- **redo size**: 該当SQLが発生させたトランザクションログ(redo log) のサイズ

ⓘ **SQL統計機能**は、リテラル値を除いたSQLを基準に統計を算出します。

基本画面ガイド



1. ① 時間で、データを照会する時間と日付を選択します。緑色のボタンを選択して照会時間を選択できます。
2. ① インスタンスでデータを照会する対象を選択します。
3. 🔍 ボタンを選択します。

上位50件までの内容をテーブルに表示します。

- ① 任意の日付と時間を表示するには、時計の表示を選択します。日付と時間のテキスト領域をクリックすると、日付と時間を選択するオプションが表示されます。
- テーブルヘッダーの各カラムをクリックすると、選択したカラムをソートすることができます。
- 並替え順で選択した項目を基準に照会結果をソートできます。項目を選択した後 🔍 ボタンを選択します。
- 照会件数で、テーブルに表示する件数を設定できます。設定を完了した上で 🔍 ボタンを選択します。
- 照会した内容をCSV形式のファイルにダウンロードするには、📄 ボタンを選択します。
- テーブルヘッダーのカラムを非表示にしたり、目的の項目を追加したりできます。カラムの順序を変更することもできます。

クエリ別の実行推移を確認

使用区分ごとに希望するデータを分類して照会したり、ソート順を変更することができます。希望するタブをクリックしてください。

テーブルの項目別の左の▶ボタンを選択します。クエリ文の5分当たりの実行回数と平均実行時間が表示されます。各棒グラフにカーソルを合わせると、その時間の数値が表示されます。

ソート基準	説明
elapsed avg (sec)	平均実行時間が最も長い値からソート
execute count	実行回数の多い順にソート
elapsed sum (sec)	総実行時間が長い値からソート
elapsed max (sec)	最大実行時間が長い値からソート
elapsed wait	クエリの待ち時間が長い順にソート
physical read	ディスクから読み取ったデータブロックが多い順にソート
session logical reads	メモリから読み取ったデータブロックが多い順にソート

クエリプレビュー

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

照会された結果リストからマウスポインタをsqlカラム項目に移動すると、クエリ文をプレビューできます。

❗ テーブル一覧でsqlのカラム項目を選択するとSQL詳細画面が表示されます。

参照結果フィルタリング

フィルター条件を追加する

照会された結果からSQLテキスト値またはエージェント名、DBユーザー名などに基づいて希望するフィルタリングが可能です。

1. フィルターオプションで+ボタンを選択します。
2. フィルターキー項目で目的のフィルタリング基準を選択します。
 - 選択した項目の値が文字に該当する場合は含む(青)、含まない(赤)の条件を選択できます。

example

- 選択した項目の値が数字に該当する場合は == (等しい), >= (より大きい)か等しい), <= (より小さい)か等しい) 条件を選択できます。
3. 条件項目で条件を選択します。
 4. 条件に一致する文字列または数字を入力します。
 5. 適用ボタンを選択します。

❗ フィルター条件の追加と削除

- フィルタリング条件を追加するには、追加 ボタンをクリックし、手順1～5を繰り返します。追加された条件はAND (&&) 条件として適用されます。
- 条件追加中に一部の項目を削除する場合は、フィルター条件の右側にある 𐀀 ボタンをクリックします。すべての条件を削除するには、𐀀 完全削除 ボタンをクリックします。
- フィルター オプションに適用された条件をすばやく削除するには、× ボタンをクリックします。
- sqlText に「WhaTap」が含まれている場合は、除外するようにデフォルトで設定されています。



フィルター

🔍 sqlText | WhaTap ✕ + ✕

フィルター条件を修正する

1. 画面上部の **フィルター** オプションで、適用されている項目をクリックします。
2. **フィルター**の修正 ウィンドウで目的の項目を修正し、**適用** ボタンをクリックします。

ログ

WhaTapのログモニタリングは、システム統合を通じて、優れた利便性とログデータへのスムーズなアクセスを提供します。独自の技術力による確実なデータ収集を強みとし、ライブテール、ログトレンド、ログ検索、イベント通知、パーサーなどの主要機能をより効率的に活用いただけます。

ログメニューは、照会およびクエリの分析とオプション設定などの機能を提供します。イベント設定メニューでは、ログ関連のイベント通知が設定できます。

- ホーム画面 > プロジェクト選択 > **ログ**

WhaTapモニタリングサービスのメイン画面からプロジェクトを選択し、プロジェクトメニューにある**ログメニュー**を選択します。次の機能を活用すると、複雑なログにより簡単にアクセスができ、様々な条件でリアルタイムに確認および分析ができます。

- **ログTail**
- **ログトレンド**
- **ログ検索**

- ホーム画面 > プロジェクト選択 > **アラート通知** > **イベント設定**

WhaTapモニタリングサービスのメイン画面でプロジェクトを選択し、プロジェクトメニューにある**アラート通知**メニューを選択し、**イベント設定**メニューを開きます。イベント条件を設定し、メール、SMS、メッセージャー、App Pushなどの様々な方法で通知を受信できます。

WhaTapログモニタリングサービスの主要なメニューガイドを次のように提供します。

適用方法

データベースとAWS RDSのログを収集する方法をご案内します。

 ライブテール

ログモニタリングのライブテールについて説明します。

 ログエクスプローラー

ログモニタリングのログエクスプローラーメニューについて案内します。

 ログトレンド

ログモニタリングのログトレンドメニューについて案内します。

 ログの検索

ログモニタリングのログ検索メニューについて説明します。

 ログの設定

ログモニタリングの設定方法について説明します。

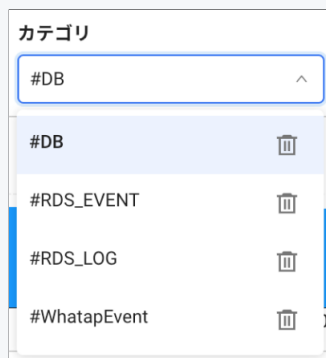
 ログパーシング

WhaTapが提供するログパーサについて案内します。

適用方法

WhaTapは、データベースのログモニタリングを提供します。サーバーインストール型データベースと、クラウドサービスであるAWS RDSのログを収集できます。これにより、WhaTapのログモニタリング機能である**ログTail**、**ログトレンド**メニューからデータベースの散在したログをひと目で確認できます。

- ✓ ログモニタリングを適用した後、各メニューの**カテゴリ**項目とログ一覧からインストール型DBは **#DB**、AWS RDSは **#RDS_LOG** および **#RDS_EVENT** と表示されます。



サーバーインストール型DBに適用する

サーバーにインストールされたデータベースのログをモニタリングするには、WhaTapの**xos.conf**ファイルに次のオプションを適用します。

xos.conf

```
file=/home/oracle/diag/rdbms/ora18/ORA18/trace/alert_ORA18.log
```

- ⓘ **Oracle Pro**商品はXOSエージェントが組み込まれているため、別途のインストール過程が必要ありません。

ログモニタリングを開始する

すべての設定を完了した後、ログ>ログ設定メニューにアクセスし、ログモニタリングを活性化します。

エージェントの設定とログモニタリング活性化

ログTail、ログトレンド、ログ検索機能で、アプリケーションに点在するログをまとめて確認することができます。

ログモニタリングの設定方法

[インストール型DB](#) AWS RDS

1. XOSエージェントのインストール

ORACLE PROの場合はxosが内蔵されており、xosのインストール過程は不要です。

2. xos.confファイルへのオプション適用

xos.confファイルに次のオプションを適用してください。

```
# 例
file=/home/oracle/diag/rdbms/ora18/ORA18/trace/alert_ORA18.log
```

3. ログモニタリング機能の活性化

設定を完了してから、ログ監視を有効にするトグルボタンをクリックしてください。



ライブテール

ⓘ ログ参照権限がない場合は、そのメニューにアクセスできません。

ホーム画面 > プロジェクト選択 > ログ > ログTail

ログTail メニューでは、サーバーコンソールに接続せずに、ログデータストリームをモニタリング画面上で直接確認できます。複雑で大量のログも、フィルター と ハイライト 機能を活用して絞り込み、素早く把握できます。ログデータの取得周期は2秒です。

主な用語は次のとおりです。

- **Category**：ログの収集および照会単位
- **Content**：ログメッセージ
- **Search Key**：ログパーサーの設定によって生成される検索キー
- **Tag**：収集されたログを検索できるタグ

ⓘ ログテーブルカラムの端をドラッグしてカラムの幅を修正できます。

エージェントオプション

エージェントオプションが設定されている場合、ログレベルを収集し、各ログの左側に色付きの縦線が次のように表示されます。

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

❗ エージェントのオプション設定

```
# whatap.conf  
weaving=log4j-2.17  
weaving=logback-1.2.8
```

- Javaエージェント2.2.22バージョンからウィービング設定にlog4j-2.17またはlogback-1.2.8設定時に使用できます。エージェントの再起動が必要です。
- ログレベルは、パーシングされたキーワードの中で level 、 type 基準で判別します。 level 、 type にパーシングされたキーが存在し、パーシングの値がFATAL、CRITICAL、ERROR、WARN、WARNING、INFOを含む場合は、ログレベルの色を表示します。

1 フィルター領域

フィルター適用

フィルターを適用すると、入力した条件に合ったログをフィルタリングします。複数のフィルターを入力できます。フィルターのタグが同じ場合、OR(||)、そうでない場合はAND(&&)が適用されます。

入力欄に文字を直接入力するか、**フィルター**入力欄をクリックしてフィルターを指定します。フィルタータグは **検索キー**、**オペレーター**、**検索値** の順に入力します。🔍 **検索** ボタンを選択すると、フィルターが適用されたデータをログ一覧で確認できます。

❗ ガイドUI

次のように入力ウィンドウの下に**ガイドUI**を提供します。入力画面またはフィルタータグにマウスカースールがある場合、**ガイドUI**を閉じる時はESCキーで操作できます。



検索キー、オペレーター、検索値の入力

- **検索キー** 入力時、一般インデックス、予約語インデックス、数字のみ入力できるインデックスを区分して推奨値を提供します。
- **オペレーター** 入力時、一般インデックス検索キーの場合は、`==`、`!=` オプションが下に表示されます。数字のみ入力できるインデックスの場合、`>`、`<`、`<=`、`>=`、`==`、`!=` オプションを提供します。
- **検索値** 入力時の一致検索(`>`、`<`、`<=`、`>=`、`==`)の時は青色で、除外検索(`!=`)の時は赤色でハイライトします。 **検索値** 入力時に大文字と小文字の区分オプションを活用して検索することができます。

- ❗
- フィルタータグが2行以上長くなる場合へ **折りたたみ**アイコンを選択して折りたたむことができます。
 - フィルタータグを入力した後、入力画面でShiftとTabキーを同時に使用して、以前のフィルタータグに移動できます。
 - フィルタータグを入力した後、入力画面でTABキーを通じて次のフィルタータグに移動できます。

フィルタータグ追加

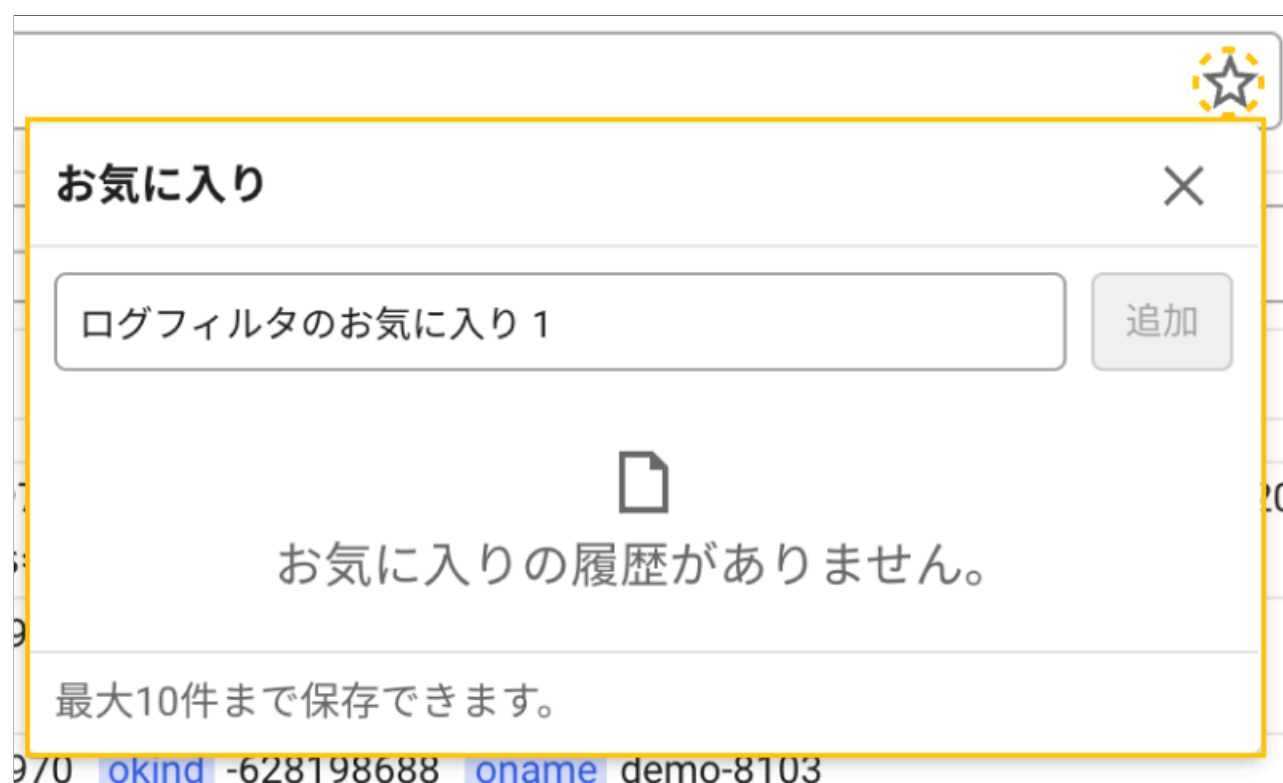
入力フィールドにテキストを入力し、Enter または Tab キーを押すか、入力欄下のガイド UI で推奨値をクリックまたは矢印キーで選択して追加できます。

フィルタータグの削除

単一のタグを削除するには、**Backspace** キーを押すか、タグ上の **X** アイコンをクリックします。すべてのタグを一度に削除するには、入力フィールド右側の **X** アイコンをクリックします。

フィルターのお気に入り

フィルターお気に入り機能を提供します。入力画面の右側にある ☆ お気に入り アイコンを選択して、目的のフィルター検索条件をお気に入りに追加、削除および既存のお気に入りを選択できます。お気に入りは最大50個まで保存されます。



フィルター適用の例外状況

- 数字のみ入力できるインデックス(.n で終わる 検索キー)を入力したタグで、 検索値 は数字のみ入力できます。
- 重複した 検索キー 、 検索値 は入力できません。
- 検索キー 、 検索値 いずれかのタグが存在しない場合は、検索できません。有効でないタグの場合、灰色と表示します。

❗ **ログTail** 検索キー として category は入力できません。

入力フィルター値の下にある数式(expression)は、ログデータを表示するときに適用されるフィルター式のプレビューです。

パースングされていないキーワードのフィルター適用

ログからパースングされていない、つまりインデックスが作成されていないキーワードを含むログを照会できます。この場合、指定範囲内のすべてのログをFull Scanします。そのため、インデックスが生成されたキーに比べて検索速度が多少落ちる可能性があります。定型化されたログデータの場合、**ログパーサの設定**を通じてインデックスキーを活用した検索をお勧めします。

フィルター ⓘ

▼

content

==

select

×

フィルターを入力してから「Enter」キーを押して追加

×

content == *select*

1. カテゴリを選択します。
2. フィルター入力欄に content を選択後、検索キーワードを入力します。
例) content *select*
3. 🔍 検索ボタンをクリックしてログを照会します。

- ❗ **ログTail**の場合、すべてのログ検索が可能であるため、カテゴリを指定する必要はありません。
- パーサ設定の詳細については、[次の文書](#)を参考にしてください。

フィルター修正

フィルターに値を入力し、入力した値をクリックすると、その値を修正できます。

フィルター ⓘ

🔍 oname == ote-account × key == aws.base:index × appender != fileAppender × フィルターを入力してから「Enter」キーを押して追加 ×

☐ 大小文字を区別する

fileAppender

searchValue*	検索値で始まるログを検索します。 ex key!=searchValue*
*searchValue	検索値で終わるログを検索します。 ex key!=*searchValue
searchValue	検索値を含むログを検索します。 ex key!=*searchValue*
*search*Value*	検索値を含むログを検索します。 ex key!=*search*Value*
re:searchValue	正規表現に一致するログを検索します。 ex key!=re:searchValue
**	検索語に該当するすべてのログを検索します。 ex key!=**

key == value 一致 key != value 除外 key == value 無効なタグです。 ⓘ ガイド表示

- ・ フィルター入力欄にテキストを書き換えて修正できます。
- ・ フィルター入力欄の下に表示されるガイドUIから推奨値を選択して修正できます。

検索キー(Search Key)

次の画像で青いボックス部分はパーシング(parsing)された検索キーです。 検索キーはログ設定のログパーサの設定タブでパーシングロジックを登録し、設定できます。

検索キー

フィルター入力の文法

タグは検索キーと検索値で構成されています。 次の例で、検索キーは `exception`、検索値は `UnknownHostException` となります。 この例は、収集したログデータのうち、IPアドレスとドメインアドレスが一致せず、サーバーをホストに接続できない場合に発生する例外(`UnknownHostException`)が含まれたログデータを照会

します。

exception == UnknownHostException

×

検索キーの種類

検索キーの種類	検索キーフォーマット	意味	検索キーと検索値の例	検索例
文字列キーワード	keyword	ファイル名	- キー: fileName - 値: /data/whatap/logs/yard.log	fileName:/data/whatap/logs/yard.log
数字キーワード	keyword.n	レスポンス時間	- キー: response_time.n - 値: 2945	response_time.n>=2945
予約語キーワード (辞書定義キーワード)	@keyword	TXID	- キー: @txid - 値: 85459614215434144	-
ログ本文キーワード	content	ログ本文	- キー: content - 値: ユーザー入力値	content: *ERROR*

ⓘ Content検索キー

- Content検索キーは、インデックスされていないログの本文を対象に検索します。例えば、`content *ERROR*` のように入力する場合は、ログ本文中で ERROR を含めたログを検索します。
- どのキーワードでインデックスをかけるべきかわからない場合は、Content検索キーを活用して問題となるキーワードを含むログを識別します。その後、**ログ設定メニュー**のログパーサ設定を通じて、該当キーワードでパーサを設定し、インデックスを生成する方式で、検索速度を早くできます。

共通文法

文法の種類	説明	例
<code>==searchValue</code>	検索値と一致するログを検索します。	<code>exception==RuntimeExceptionexception</code>
<code>!=searchValue</code>	検索値を除いたログを検索します。	<code>exception!=RuntimeException</code>
<code>*searchValue</code>	検索値で終わるログを検索します。	<code>word==*hello</code>
<code>searchValue*</code>	検索値で始まるログを検索します。	<code>word==hello*</code>
<code>*searchValue*</code>	検索値として中間に含まれるログを検索します。	<code>word==*hello*</code>
<code>*search*Value*</code>	検索値として含まれるログを検索します。	<code>word==*he*llo*</code>
<code>re:{正規表現}</code>	正規表現にマッチングされるログを検索します。	<code>caller==re:^(i\.w\.a\.w\.s\.v\.r\.</code>
<code>**</code>	検索語に該当するすべてのログを検索します。	

検索キーが数字キーワード(keyword.n)の場合の文法

次の文法は検索キーが、`keyword.n` 形式の場合のみに対応します。

- 検索値は数字のみ使えます。
- `.n` キーワードの値にはPrefixを付けません。`.n` 以外のキーワードは、すべてPrefixを付ける必要があります。

例) `+>searchValue` は有効ではありません。

文法の種類	説明	例
<code>>searchValue</code>	検索値よりも大きな値を含むログを照会します。	<code>response_time.n>3000</code>
<code>>=searchValue</code>	検索値よりも大きい、同じ値を含むログを照会します。	<code>response_time.n>=3000</code>
<code>==searchValue</code>	検索値と同じ値を含むログを照会します。	<code>response_time.n==3000</code>

文法の種類	説明	例
<code>!=searchValue</code>	検索値とは異なる値を含むログを照会します。	<code>response_time.n!=3000</code>
<code><searchValue</code>	検索値より小さい値を含むログを照会します。	<code>response_time.n<3000</code>
<code><=searchValue</code>	検索値より小さいか、同じ値を含むログを照会します。	<code>response_time.n<=3000</code>

個人情報の照会

ログの個人情報照会権限がある場合**個人情報の表示**  トグルボタンの活性化を通じて非識別化処理した個人情報を表示することができます。

- ① ログの個人情報照会権限についての詳細は[次の文書](#)を参照してください。
- ログ個人情報の非識別化機能についての詳細は[次の文書](#)を参照してください。

2 コンテンツハイライト領域

ログのコンテンツ内で目的のキーワードを簡単に識別できるよう、ハイライト機能を提供します。単一または複数のキーワードでフィルタリングできます。

ログTail

カテゴリー

AppLog

フィルター

content

+=select*

×

フィルターを入力してから「Enter」キーを押して追加します。

content == +=select*

act

🔍

⏸

🔄

📄

⚙️

	oname	タイムスタンプ	ログ
▶	demo-8102	2023-11-24 14:06:43.648	@txid 4785826165275137961 pcode 5490 oname demo-8102 onodeName node-0 oid 1387800924 okindName demo-okind-0 okind -1383513482 select distinct ename, deptno, sal, job from emp
▶	demo-8102	2023-11-24 14:06:43.654	@txid 4785826165275137961 pcode 5490 oname demo-8102 onodeName node-0 oid 1387800924 okindName demo-okind-0 okind -1383513482 select ename, 1000 from emp
▶	demo-8102	2023-11-24 14:06:43.659	@txid -240363117644553212 pcode 5490 oname demo-8102 onodeName node-0 oid 1387800924 okindName demo-okind-0 okind -1383513482 select distinct pp.lastname, pp.firstname
▶	demo-8102	2023-11-24 14:06:43.659	@txid 4785826165275137961 pcode 5490 oname demo-8102 onodeName node-0 oid 1387800924 okindName demo-okind-0 okind -1383513482 select 'total income is', ((orderqty * unitprice) * (1.0 - unitpricediscount)), ' for '

1. ハイライトしたいキーワードをキーワード入力欄に入力します。

- 例: `select`

2. 検索したキーワードがログリストのContent内でハイライト表示されます。

-  **拡大表示** アイコンをクリックすると、**ログ** と **タイムスタンプ** を全画面で確認できます。
- ハイライトされたキーワード間の移動はショートカットで操作できます。
 - `Enter` : 次の行へ移動
 - `Shift+Enter` : 前の行へ移動
 - `↑ / ↓` アイコン: 行の移動
 - `Cmd + F (Mac) / Ctrl + F (Windows)`: キーワード検索

複数キーワード条件

複数キーワードでハイライトする場合は、次のように作成します。

入力文字列	説明	結果
<code>a b c</code>	各キーワードをスペースで区切ります。	a, b, c
<code>"Whatap is good."</code>	キーワードにスペースを含める場合は、 <code>"</code> または <code>"</code> で囲みます。	Whatap is good.
<code>"Whatap\\ is good."</code>	<code>"</code> で囲まれたキーワードに、 <code>\</code> を含める場合は、 <code>\\</code> と入力する必要があります。	Whatap\ is good.

ハイライトカラー設定

ハイライトするキーワードおよび色を設定できます。

- デフォルトでは、ログレベルに応じたハイライト (**FATAL**、**ERROR**、**WARN**、**SEVERE**、**CRITICAL**) が適用されています。
- 設定した内容は**プロジェクト単位**で保存されます。

1.  アイコンをクリックします。

2. **ハイライト** ウィンドウで、色を設定するキーワードを入力欄に追加してください。
3. 入力欄の左側の色をクリックすると、色を選択できます。

テーブル設定

カラム設定

2 領域の右で  **カラム設定** ボタンを選択します。カラムを追加したり、順序を設定できます。

- **カラム追加**

タグを選択してテーブルにカラムを追加できます。**log**カラム選択を解除する場合**ログ表示詳細設定**が確認できません。必ず1つ以上のカラムを選択します。

- **カラム順序設定**

カラムを追加すると**表示項目**に該当カラムが追加されます。希望のカラムをドラッグしてカラムの順序を変更します。

ログ表示詳細設定

2 領域右で  **ログ表示詳細設定** ボタンを選択します。デフォルトで**content**、**タグ**すべてチェックされており、両方の項目を表示します。**content**と**タグ**どちらかは必ず選択します。

チェックが解除されたアイテムはテーブルに表示されません。次のように**タグ**を解除する場合、テーブルでログの**タグ**は表示されません。

ログテーブルコンテンツタグ非表示

タグ管理一覧にタグを追加すると、追加した順にログのタグがソートされます。タグの順序はドラッグして変更できます。追加したタグを無効にすると、無効にしたタグはログのタグに表示されません。

❗ **カラムおよびログ表示設定の適用範囲**

- **カラム設定**と**ログ表示詳細設定**は、**ログTail**、**ログ検索**、**ログトレンド**で使用できます。
- 同一プロジェクト内の**ログTail**、**ログ検索**、**ログトレンド**は、**カラム設定**と**ログ表示詳細設定**を共有します。

ログファイルのダウンロード

検索したログの結果を **CSV** または **TXT** ファイル形式でダウンロードできます。

1. 画面中央右側の **↓ ダウンロード** ボタンをクリックします。
2. **CSVダウンロード** または **Logダウンロード** をクリックして、希望の形式のファイルをダウンロードします。

ログエクスプローラー

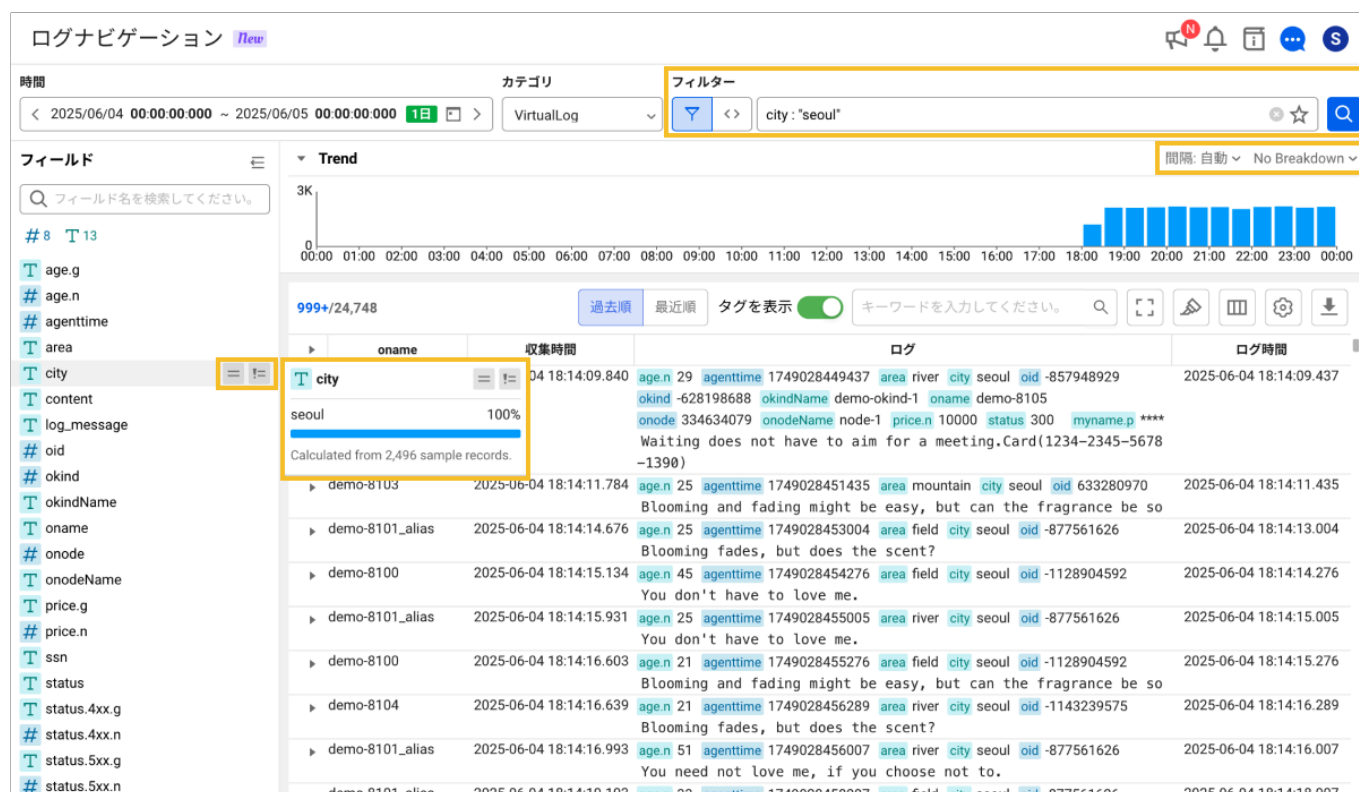
ⓘ ログ参照権限がない場合は、そのメニューにアクセスできません。

ホーム画面 > プロジェクト選択 > ログ > ログエクスプローラー *New*

ログエクスプローラー (Log Explorer) 機能は、ログデータを迅速に検索、フィルタリング、分析します。フィールド構造に関する情報を把握し、クエリベースのフィルタリングによって必要なログを効率的に検索し、結果をデータの可視化を通じて確認できます。本機能はLuceneインデックスをベースに動作し、主な特徴は以下のとおりです。

1. **ログトレンド分析および検索:** ログのトレンドと検索機能を同時にサポートし、時間経過によるログの変化傾向を把握できます。
2. **高速検索およびフィルタリング:** ユーザーは簡単なキーワード入力だけで、問題のあるログを迅速に検索できます。WhaTapログ検索クエリおよびLuceneクエリ言語に対応しており、さまざまな検索条件を設定できます。
3. **オートコンプリート機能:** クエリ入力時にオートコンプリート機能が提供され、ユーザーが簡単にクエリを作成し、習得できるようサポートします。
4. **データ可視化:** スタックグラフを活用して時系列のログ件数データを視覚的に表示し、ドリルダウン機能を通じて特定のログを深く分析できます。
5. **お気に入りと共有:** よく使うフィルターはお気に入りとして保存でき、URLを通じて他のユーザーと簡単に共有できます。

基本画面ガイド



時間設定

上部のオプションバーにある**タイムセレクター**を使用して、ログ検索の時間範囲を設定できます。

カテゴリ

上部のオプションバーにある**カテゴリ選択**で、探索したいログの種類を選択します。（例：AppLogなど）

フィルター

上部オプションバーの**フィルター**でクエリ言語を選択し、検索条件を入力してフィルターを適用します。
検索クエリ構文は、WhaTapログ検索クエリとLuceneクエリの両方をサポートしています。

-  : WhaTapログ検索クエリ
-  : Luceneクエリ

検索クエリ構文

WhaTap log search query

WhaTapのログ検索クエリはオートコンプリート機能を提供しており、簡単なキーワード入力だけで問題のログを迅速に検索できます。and、orなどのさまざまな構文を活用して、さまざまな検索条件を入力できます。

- 例、検索フィルターに `error` と入力するだけで、content（ログ本文）を検索できます。
 1. 上部オプションバーでフィルターアイコンをクリックした後、クエリ検索ウィンドウでフィールド（選択した時間とカテゴリに応じたリストが表示されます）を選択してください。
 2. 演算子を選択してください。
 - a. 基本: `:`、`*`、`and`、`or`
 - b. 選択したフィールドが `Number` 型の場合（基本 + `<=`、`>=`、`<`、`>`）
 - c. 選択したフィールドが `String` の場合（基本のみ）
 3. 値（選択した時間、カテゴリ、フィールドに一致するリストが表示されます）を選択してください。

検索インデックスフィールドタイプ

サポートされているフィールドタイプは、`Number`、`String`、`Boolean` の3種類のみです。

- `Number` : Long、double、float 型の数値
- `String` : メール本文のような全文テキスト
- `Boolean` : True または False の値

 WhaTapログ検索クエリの構文に関する詳細は[次の文書](#)を参照してください。

Lucene

Apache Lucene 8.11.1バージョンのオープンソースの構文がそのまま利用可能です。ただし、自動補完機能は提供していません。

❗ LuceneのQueryParserクラシックパッケージに関する詳細は[公式ドキュメント](#)を参照してください。

フィールド一覧

ドリルダウン分析に活用できます。

フィールド情報のフィルターを反映する

フィールド一覧でフィールド名を確認し、`=` または `!=` ボタンをクリックして条件をフィルターに適用してください。

- フィールドにマウスオーバーすると、`=`、`!=` ボタンが表示され、クリックするとフィルターに反映されます。
- フィールド検索機能を使って、目的のフィールドを素早く見つけることができます。

Trendチャート

Trendチャートでインターバルやブレイクダウンを適用して、ログをグラフで確認できます。

- グラフィインタラクション:** マウสดラッグや  ボタンで時間のズームイン、バークリックで該当領域のログを確認可能
- インターバル:** 自動、1秒、5秒、30秒、1分、5分、10分、30分、1時間から選択可能
- Breakdown:** 特定のフィールドを基準にTop 3 + Othersの値を分布として可視化し、ドリルダウンフィルターの適用が可能

breakdown

Breakdownは、ログ件数の推移を特定のフィールドで分析する可視化機能です。

- 特定のフィールドのTop 3の値とOthers（その他の値）を区別して、スタックグラフで分布を表示します。
- ドリルダウン分析に活用でき、凡例やスタックグラフの特定領域（部分グラフ）をクリックして、特定の値のみを

強調したり、フィルター条件を追加することができます。

ログの確認

検索条件に一致するログの一覧を確認でき、**フィールド一覧**ではフィールド情報とログタグが同じ色で表示されます。

コンテンツハイライト

ログのコンテンツ内で目的のキーワードを簡単に識別できるよう、ハイライト機能を提供します。単一または複数のキーワードでフィルタリングできます。

1. ハイライトしたいキーワードをキーワード入力欄に入力します。
 - 例: `select`
2. 検索したキーワードがログリストのContent内でハイライト表示されます。
 -  **拡大表示** アイコンをクリックすると、**ログ** と **タイムスタンプ** を全画面で確認できます。
 - ハイライトされたキーワード間の移動はショートカットで操作できます。
 - `Enter` : 次の行へ移動
 - `Shift+Enter` : 前の行へ移動
 - `^ / v` アイコン: 行の移動
 - `Cmd + F` (Mac) / `Ctrl + F` (Windows): キーワード検索

複数キーワード条件

複数キーワードでハイライトする場合は、次のように作成します。

入力文字列	説明	結果
<code>a b c</code>	各キーワードをスペースで区切ります。	a, b, c
<code>"Whatap is good."</code>	キーワードにスペースを含める場合は、 <code>"</code> または <code>"</code> で囲みます。	Whatap is good.

入力文字列	説明	結果
"Whatap\\ is good."	""で囲まれたキーワードに、\を含める場合は、\\と入力する必要があります。	Whatap\ is good.

ハイライトカラー設定

ハイライトするキーワードおよび色を設定できます。

- デフォルトでは、ログレベルに応じたハイライト（FATAL、ERROR、WARN、SEVERE、CRITICAL）が適用されています。
- 設定した内容はプロジェクト単位で保存されます。
 - 🔗アイコンをクリックします。
 - ハイライト ウィンドウで、色を設定するキーワードを入力欄に追加してください。
 - 入力欄の左側の色をクリックすると、色を選択できます。

カラム設定

☐ ボタンをクリックして、カラムを追加したり順序を設定することができます。

カラム追加

カラム追加：タグを選択して、テーブルにカラムを追加できます。

⚠ ログカラムの選択を解除すると、ログ表示詳細設定を確認できません。必ず1つ以上のカラムを選択します。

カラム順序設定

カラム設定でカラムを追加すると、表示項目に該当カラムが追加されます。希望するカラムをドラッグして、カラムの順序を変更してください。

ログ表示詳細設定

⚙ ボタンをクリックして、ログの表示を詳細に設定します。デフォルトでは **content**、**タグ** の両方がチェックされており、両方の項目が表示されます。

- **content**と**タグ**のいずれかを必ず選択してください。
- **タグ管理**リストにタグを追加すると、追加した順にログのタグが表示されます。タグの順序はドラッグで変更できます。追加したタグを無効にすると、無効にしたタグはログのタグに表示されません。
- チェックが解除された項目は、テーブルに表示されません。次のように**タグ**を解除する場合、テーブルからログの**タグ**は表示されません。

log table contents remove tag sc

ⓘ カラムおよびログ表示設定の適用範囲

- **カラム設定**と**ログ表示詳細設定**は、**ログTail**、**ログ検索**、**ログトレンド**で使用できます。
- 同一プロジェクト内の**ログTail**、**ログ検索**、**ログトレンド**は、**カラム設定**と**ログ表示詳細設定**を共有します。

ログファイルのダウンロード

検索したログの結果を **CSV** または **TXT** ファイル形式でダウンロードできます。

1. 画面中央右側の **↓ ダウンロード** ボタンをクリックします。
2. **CSVダウンロード** または **Logダウンロード** をクリックして、希望の形式のファイルをダウンロードします。

ログトレンド

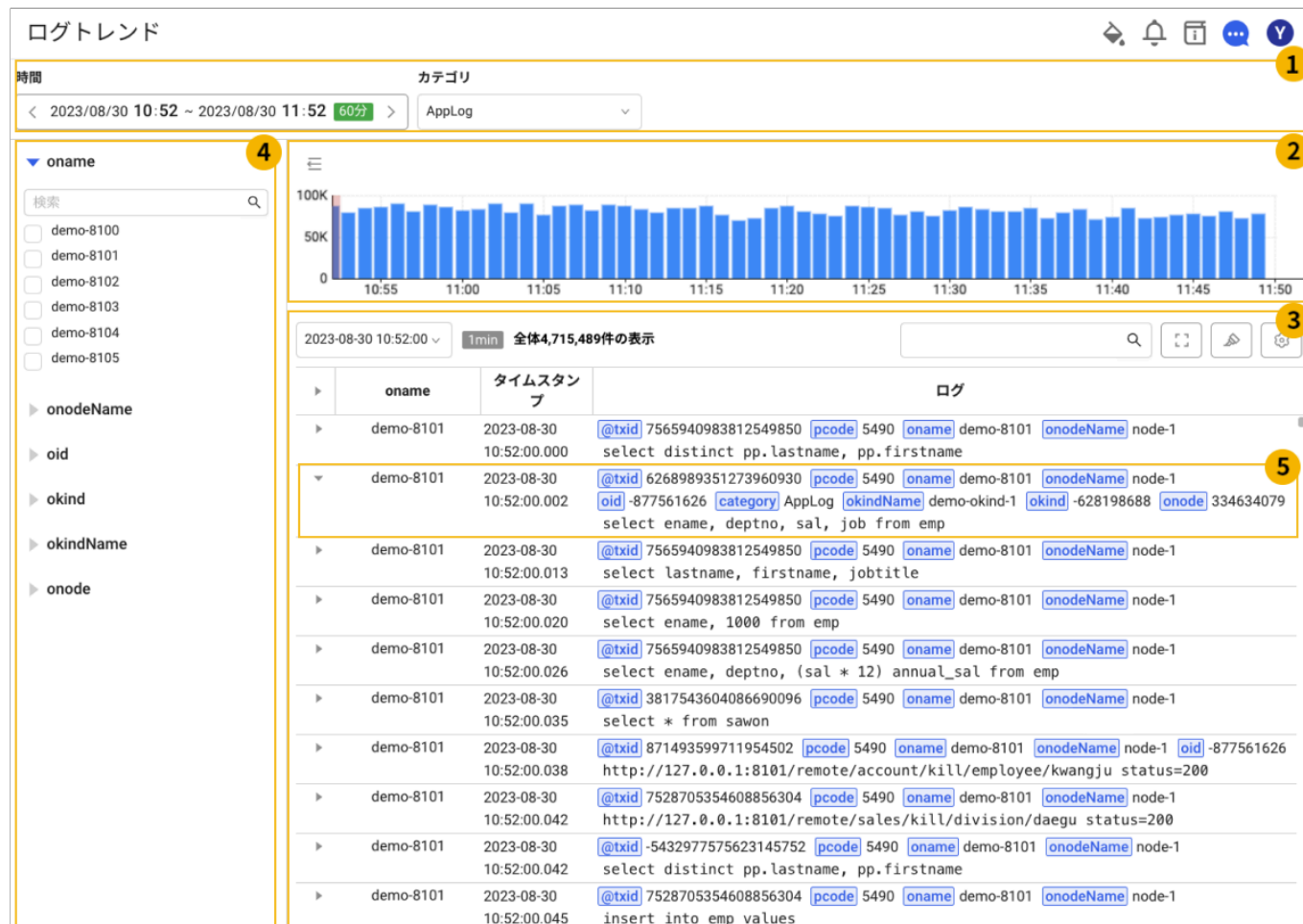
ⓘ ログ参照権限がない場合は、そのメニューにアクセスできません。

ホーム画面 > プロジェクト選択 > ログ > ログトレンド

ログトレンドでは、タイプ別に分類されたログの発生件数の推移を通じて、特定エラータイプの発生パターンを確認し、時間別の詳細ログデータを確認できます。ハイライト機能により、目的のログをすばやく識別できます。カテゴリ別に収集されたログの推移を照会できます。

主な用語は次のとおりです。

- **Category**：ログの収集および照会単位
- **Content**：ログメッセージ
- **Search Key**：ログパーサーの設定によって生成される検索キー
- **Tag**：収集されたログを検索できるタグ



データ参照

- 下部までスクロールすると、次のデータを表示します。
- 1で検索するログデータの時間と収集単位としてカテゴリを指定します。
- カテゴリを変更すると、選択されたカテゴリに対応するログを表示します。2バーチャートと3ログテーブルで確認できます。
- 2棒グラフのバーをクリックすると、バーの時間範囲に対応するログを検索します。
- 3ログテーブルの左上から照会されたログの総数を確認できます。
- 3 ログテーブル上段の右の「全画面」アイコンを選択すると、ログと収集時間を全画面で確認できます。
- 4 サイドメニューからタグでフィルタリングすることでログを確認できます。検索キーは最大2つまで選択でき、

検索値は複数選択できます。

- エージェントオプションが設定されている場合、ログレベルを収集し、各ログの左側に色付きの縦線が次のように表示されます。

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.err
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.err
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.err
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.err

❗ エージェントのオプション設定

```
# whatap.conf
weaving=log4j-2.17
weaving=logback-1.2.8
```

- Javaエージェント2.2.22バージョンからウィービング設定にlog4j-2.17またはlogback-1.2.8設定時に使用できます。エージェントの再起動が必要です。
- ログレベルは、パースされたキーワードの中で level 、 type 基準で判別します。 level 、 type にパースされたキーが存在し、パースの値がFATAL、CRITICAL、ERROR、WARN、WARNING、INFOを含む場合は、ログレベルの色を表示します。

- ❗ ログテーブルカラムの端をドラッグしてカラムの幅を修正できます。

個人情報の照会

ログの個人情報照会権限がある場合個人情報の表示  トグルボタンの活性化を通じて非識別化処理した個人情報を表示することができます。

- ❗ ログの個人情報照会権限についての詳細は次の文書を参照してください。
- ログ個人情報の非識別化機能についての詳細は次の文書を参照してください。

ログContentの確認

Content はログメッセージです。ログカラムの1行目には、パースされたキーと値が表示され、2行目にはContent（ログメッセージ）が表示されます。

- **3** ログテーブルの行(ログ)ごとに ▶ 詳細ボタンがあります。 ▶ 詳細ボタンを選択すると **5** のように、ログの全体 contentを確認できます。

ハイライト設定

目的のキーワードを簡単に識別できるように、ハイライト機能を提供しています。🔍 アイコンをクリックし、キーワードを入力してからEnterキーを押して追加してください。

コンテンツハイライト

ログのコンテンツ内で目的のキーワードを簡単に識別できるよう、ハイライト機能を提供します。単一または複数のキーワードでフィルタリングできます。

1. ハイライトしたいキーワードをキーワード入力欄に入力します。
 - 例: `select`
2. 検索したキーワードがログリストのContent内でハイライト表示されます。
 - **[] 拡大表示** アイコンをクリックすると、**ログ** と **タイムスタンプ** を全画面で確認できます。
 - ハイライトされたキーワード間の移動はショートカットで操作できます。
 - `Enter` : 次の行へ移動
 - `Shift+Enter` : 前の行へ移動
 - `^ / v` アイコン: 行の移動
 - `Cmd + F (Mac) / Ctrl + F (Windows)`: キーワード検索

複数キーワード条件

複数キーワードでハイライトする場合は、次のように作成します。

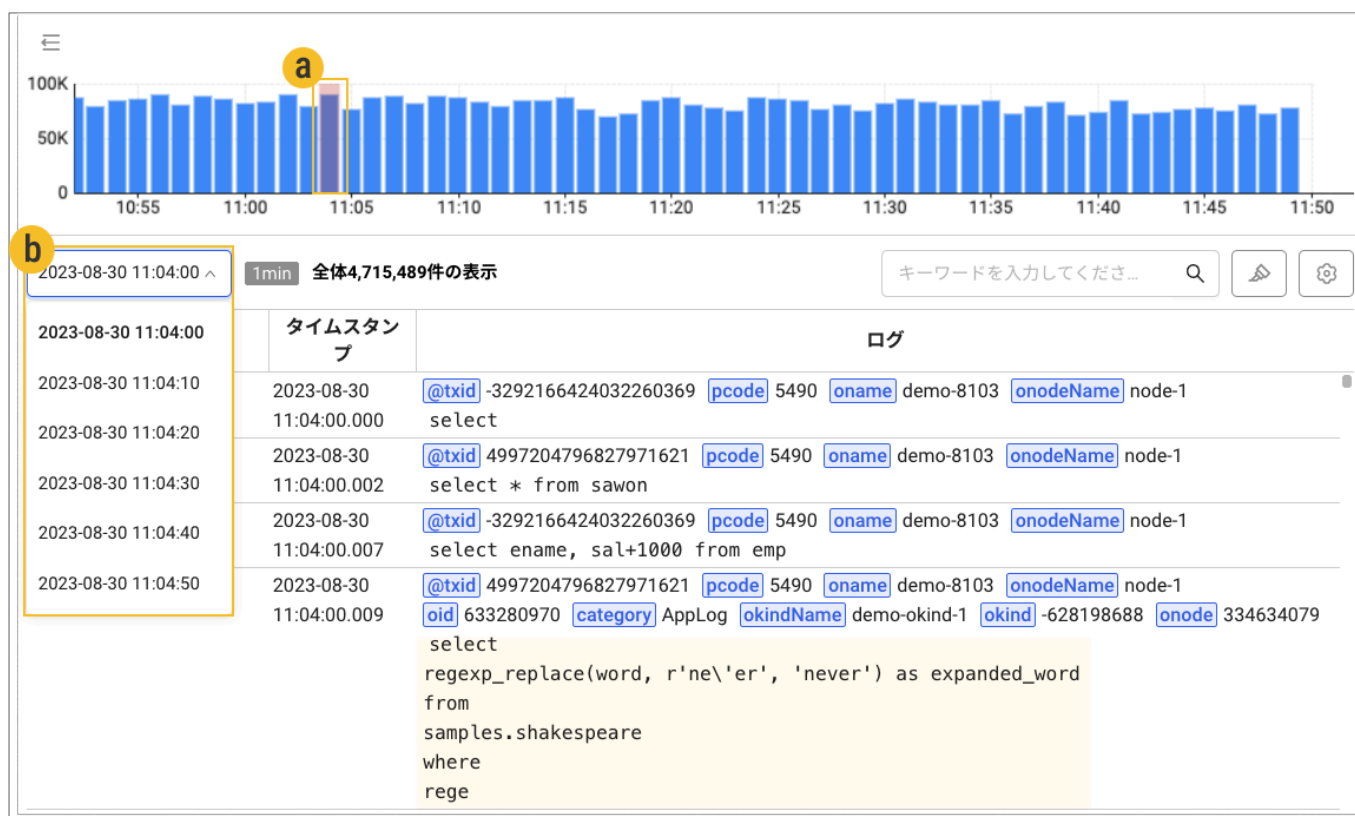
入力文字列	説明	結果
a b c	各キーワードをスペースで区切ります。	a, b, c
"Whatap is good."	キーワードにスペースを含める場合は、" または "" で囲みます。	Whatap is good.
"Whatap\\ is good."	""で囲まれたキーワードに、\ を含める場合は、\\ と入力する必要があります。	Whatap\\ is good.

ハイライトカラー設定

ハイライトするキーワードおよび色を設定できます。

- デフォルトでは、ログレベルに応じたハイライト（FATAL、ERROR、WARN、SEVERE、CRITICAL）が適用されています。
- 設定した内容はプロジェクト単位で保存されます。
 -  アイコンをクリックします。
 - ハイライト ウィンドウで、色を設定するキーワードを入力欄に追加してください。
 - 入力欄の左側の色をクリックすると、色を選択できます。

ログをチャートで検索



- バーチャートで **a** 目的の時間領域バーをクリックまたはドラッグして、その時間範囲のログを確認できます。
- 棒チャートの下ログテーブル上部左側の **b** 時間選択オプションを使用して、次のように選択した時間帯でより詳細なログ検索が可能です。
 - 1min : interval (チャート内の棒の間隔)
 - 時間選択オプション : 選択された時間範囲を6つの間隔に分割した時間帯

テーブル設定

カラム設定

3 領域の右で  **カラム設定** ボタンを選択します。カラムを追加したり、順序を設定できます。

- **カラム追加**

タグを選択してテーブルにカラムを追加できます。**log**カラム選択を解除する場合**ログ表示詳細設定**が確認できません。必ず1つ以上のカラムを選択します。

- **カラム順序設定**

カラムを追加すると**表示項目**に該当カラムが追加されます。希望のカラムをドラッグしてカラムの順序を変更します。

ログ表示詳細設定

3 領域右で  **ログ表示詳細設定** ボタンを選択します。デフォルトで**content**、**タグ**すべてチェックされており、両方の項目を表示します。**content**と**タグ**どちらかは必ず選択します。

チェックが解除されたアイテムはテーブルに表示されません。次のように**タグ**を解除する場合、テーブルでログの**タグ**は表示されません。

ログテーブルコンテンツタグ非表示

タグ管理一覧にタグを追加すると、追加した順にログのタグがソートされます。タグの順序はドラッグして変更できます。追加したタグを無効にすると、無効にしたタグはログのタグに表示されません。

ⓘ カラムおよびログ表示設定の適用範囲

- **カラム設定**と**ログ表示詳細設定**は、**ログTail**、**ログ検索**、**ログトレンド**で使用できます。
- 同一プロジェクト内の**ログTail**、**ログ検索**、**ログトレンド**は、**カラム設定**と**ログ表示詳細設定**を共有します。

ログファイルのダウンロード

検索したログの結果を **CSV** または **TXT** ファイル形式でダウンロードできます。

1. 画面中央右側の **↓ ダウンロード** ボタンをクリックします。
2. **CSVダウンロード** または **Logダウンロード** をクリックして、希望の形式のファイルをダウンロードします。

ログの検索

ⓘ ログ参照権限がない場合は、そのメニューにアクセスできません。

ホーム画面 > プロジェクト選択 > ログ > ログ検索

ログ検索メニューから収集された大量のログをさまざまな条件で検索し、ユーザーが必要なログを指定できます。複数の検索条件をパーシングされたキーとバリューで指定できるため、目的の条件に一致するログデータのみを抽出します。

動的ページで取得したログデータは、設定された行単位で取得され、スクロールで画面の下まで移動すると自動的に次のデータが取得されて表示されます。

主な用語は次のとおりです。

- **Category**：ログの収集および照会単位
- **Content**：ログメッセージ
- **Search Key**：ログパーサーの設定によって生成される検索キー
- **Tag**：収集されたログを検索できるタグ

ログ検索

時間: 2023/08/24 15:10 ~ 2023/08/24 16:10 60分

フィルター: フィルターを入力してから「Enter」キーを押して追加します。

DB (4,623)

- #DB (4,623)
- #RDS_LOG (85)
- #WhatapEvent (5)

全体4,713件の表示

Timestamp: 過去 最近

キーワードを入力してください

oname	タイムスタンプ	ログ
DBX-1-203-5432	2023-08-24 15:10:00.954	2023-08-24 15:09:57 KST [8053]: [1-1] user=kwlee,db=db1,app=,client=[local] LOG: duration: 17814.873 ms statement: s
DBX-1-203-5432	2023-08-24 15:10:06.990	2023-08-24 15:10:02 KST [8201]: [1-1] user=whatap-f,db=selong.sql,app=[unknown],client=[local] FATAL: role "whatap-f" does not exist
DBX-1-203-5432	2023-08-24 15:10:10.001	2023-08-24 15:10:08 KST [8055]: [9-1] user=kwlee,db=db1,app=,client=[local] LOG: duration: 15016.880 ms statement: select (15);
DBX-1-203-5432	2023-08-24 15:10:10.001	2023-08-24 15:10:08 KST [7986]: [11-1] user=kwlee,db=db1,app=,client=[local] CONTEXT: while locking tuple (1,129) in relation "kwlee"
DBX-1-203-5432	2023-08-24 15:10:10.001	2023-08-24 15:10:08 KST [7986]: [10-1] user=kwlee,db=db1,app=,client=[local] LOG: process 7986 acquired ShareLock on
DBX-1-203-5432	2023-08-24 15:10:10.001	2023-08-24 15:10:08 KST [7986]: [12-1] user=kwlee,db=db1,app=,client=[local] STATEMENT: update kwlee set b='67890' wh
DBX-1-203-5432	2023-08-24 15:10:10.001	2023-08-24 15:10:08 KST [8127]: [5-1] user=kwlee,db=db1,app=,client=[local] STATEMENT: update kwlee set b='67890' whe
DBX-1-203-5432	2023-08-24 15:10:10.001	2023-08-24 15:10:08 KST [8127]: [4-1] user=kwlee,db=db1,app=,client=[local] LOG: process 8127 acquired ExclusiveLock
DBX-1-203-5432	2023-08-24 15:10:10.001	2023-08-24 15:10:08 KST [7986]: [13-1] user=kwlee,db=db1,app=,client=[local] LOG: duration: 44040.104 ms statement:
DBX-1-203-5432	2023-08-24 15:10:10.001	2023-08-24 15:10:08 KST [8129]: [4-1] user=kwlee,db=db1,app=,client=[local] LOG: process 8129 acquired AccessExclusiv
DBX-1-203-5432	2023-08-24 15:10:10.001	2023-08-24 15:10:08 KST [8129]: [5-1] user=kwlee,db=db1,app=,client=[local] STATEMENT: select /* this is test */ "a",

データ参照

- 画面の下までスクロールすると、次のデータを検索します。一度に、10,000個のログを参照できます。
- ③ ログテーブルの左上に検索されたログの総数を確認することができます。
- ログデータを時間の順と逆順で照会できます。③ ログテーブルの右上の古い順と新しい順の中から照会方式を選択します。
- 時間範囲を指定した後適用ボタンを選択して照会時間を設定し、🔍 検索ボタンを選択してデータを照会します。
- ③ ログテーブルの右上の🔍 拡大表示アイコンを選択すると、ログとタイムスタンプを全画面で確認できます。
- エージェントオプションが設定されている場合、ログレベルを収集し、各ログの左側に色付きの縦線が次のように表示されます。

2023-12-18 14:31:02.563	INFO	1702877462042	1702877462042	AppLog	io.home.test.logback02starter.base.web.LogbackControllerGreeting
2023-12-18 14:31:02.563	WARN	1702877462043	1702877462043	AppLog	io.home.test.logback02starter.base.web.LogbackControllerError
2023-12-18 14:31:02.586	error	1702877462043	1702877462043	AppServer	io.home.test.logback02starter.base.errors.exception.ApiException: [2204] Process failure. Please try again.
2023-12-18 14:31:02.586	error	1702877462057	1702877462057	AppServer	Servlet.service() for servlet [DispatcherServlet] in context with path [] threw exception [Request processing failed: io.home.test.logback02starter.base.err
2023-12-18 14:31:02.586	error	1702877462081	1702877462081	AppServer	Servlet.service() for servlet [DispatcherServlet] in context with path [] threw exception [Request processing failed: io.home.test.logback02starter.base.err
2023-12-18 14:31:02.586	error	1702877462087	1702877462087	AppServer	Servlet.service() for servlet [DispatcherServlet] in context with path [] threw exception [Request processing failed: io.home.test.logback02starter.base.err

① エージェントのオプション設定

```
# whatap.conf
weaving=log4j-2.17
weaving=logback-1.2.8
```

- Javaエージェント2.2.22バージョンからウィーピング設定にlog4j-2.17またはlogback-1.2.8設定時に使用できます。エージェントの再起動が必要です。
- ログレベルは、パーシングされたキーワードの中で level 、 type 基準で判別します。 level 、 type にパーシングされたキーが存在し、パーシングの値がFATAL、CRITICAL、ERROR、WARN、WARNING、INFOを含む場合は、ログレベルの色を表示します。

① ログテーブルカラムの端をドラッグしてカラムの幅を修正できます。

ログContentの確認

Content はログメッセージです。ログカラムの1行目には、パースされたキーと値が表示され、2行目にはContent（ログメッセージ）が表示されます。

- ③ ログテーブルの行(ログ)ごとに ▶ 詳細ボタンがあります。 ▶ 詳細ボタンを選択すると④のように、ログの全体Contentを確認できます。
- ログのタグを選択すると、コピー, 検索, 除外検索, 隣接ログを検索できるドロップダウンメニューが表示されます。

フィルター

フィルター適用

① 左側の時間選択バーで時間範囲を指定できます。右側から、フィルターを適用すると、入力した条件に合ったログをフィルタリングします。複数のフィルターを入力できます。フィルターのタグが同じ場合、OR(||)、そうでない場合はAND(&&)が適用されます。

入力欄に文字を直接入力するか、**フィルター**入力欄をクリックしてフィルターを指定します。フィルタータグは **検索キー**、**オペレーター**、**検索値** の順に入力します。 **検索** ボタンを選択すると、フィルターが適用されたデータをログ一覧で確認できます。

① ガイドUI

次のように入力ウィンドウの下に**ガイドUI**を提供します。入力画面またはフィルタータグにマウスカーソルがある場合、**ガイドUI**を閉じる時はESCキーで操作できます。



検索キー、オペレーター、検索値の入力

- **検索キー** 入力時、一般インデックス、予約語インデックス、数字のみ入力できるインデックスを区分して推奨値を提供します。
- **オペレーター** 入力時、一般インデックス検索キーの場合は、**==**、**!=** オプションが下に表示されます。数字のみ入力できるインデックスの場合、**>**、**<**、**<=**、**>=**、**==**、**!=** オプションを提供します。
- **検索値** 入力時の一致検索(**>**、**<**、**<=**、**>=**、**==**)の時は青色で、除外検索(**!=**)の時は赤色でハイライトしま

す。 **検索値** 入力時に大文字と小文字の区分オプションを活用して検索することができます。

- ❗ • フィルタータグが2行以上長くなる場合へ **折りたたみ**アイコンを選択して折りたたむことができます。
- フィルタータグを入力した後、入力画面でShiftとTabキーを同時に使用して、以前のフィルタータグに移動できます。
- フィルタータグを入力した後、入力画面でTABキーを通じて次のフィルタータグに移動できます。

フィルタータグ追加

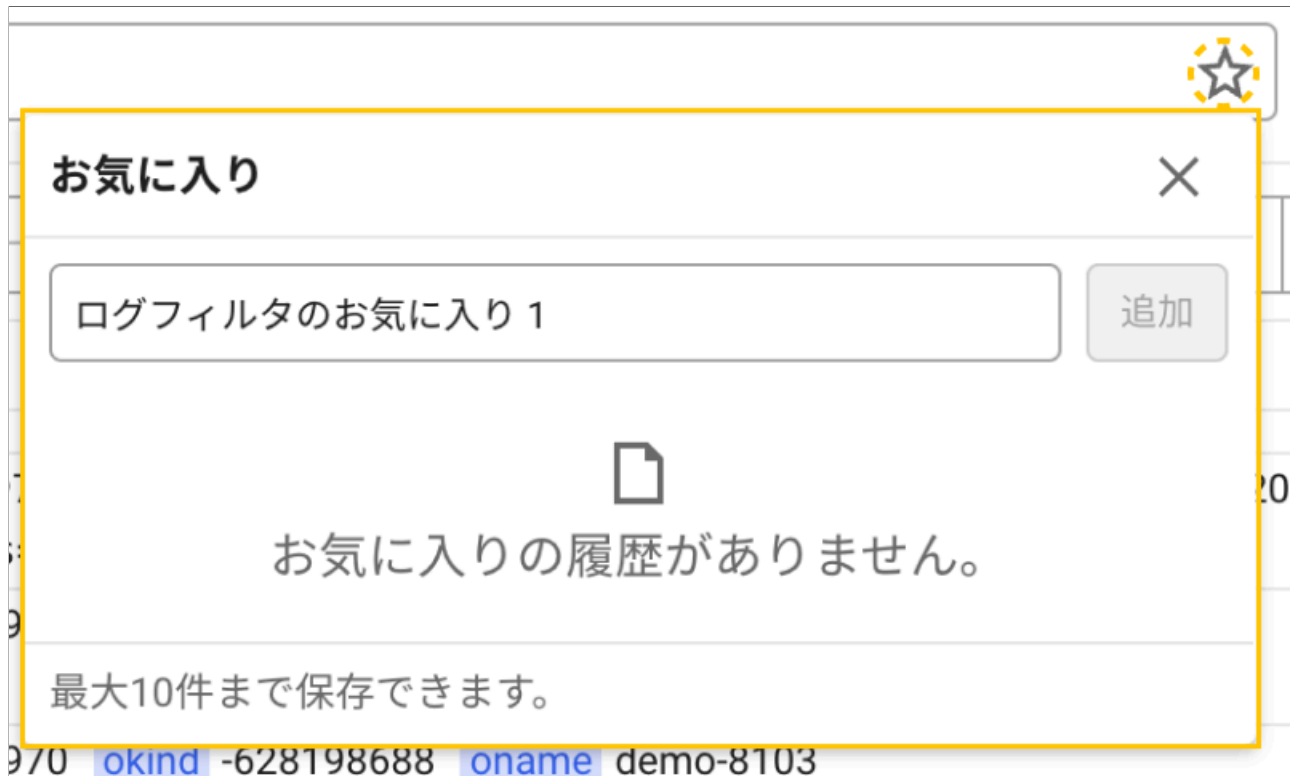
入力フィールドにテキストを入力し、Enter または Tab キーを押すか、入力欄下のガイド UI で推奨値をクリックまたは矢印キーで選択して追加できます。

フィルタータグの削除

単一のタグを削除するには、**Backspace** キーを押すか、タグ上の **X** アイコンをクリックします。すべてのタグを一度に削除するには、入力フィールド右側の **X** アイコンをクリックします。

フィルターのお気に入り

フィルターお気に入り機能を提供します。入力画面の右側にある☆ **お気に入り**アイコンを選択して、目的のフィルター検索条件をお気に入りに**追加**、**削除**および既存のお気に入りを**選択**できます。**お気に入り**は最大50個まで保存されます。



フィルター適用の例外状況

- 数字のみ入力できるインデックス(.n で終わる 検索キー)を入力したタグで、 検索値 は数字のみ入力できます。
- 重複した 検索キー 、 検索値 は入力できません。
- 検索キー 、 検索値 いずれかのタグが存在しない場合は、検索できません。有効でないタグの場合、灰色と表示します。

構文解析されていないキーワードのフィルター適用

ログからパーシングされていない、つまりインデックスが作成されていないキーワードを含むログを照会できます。この場合、指定範囲内のすべてのログをFull Scanします。そのため、インデックスが生成されたキーに比べて検索速度が多少落ちる可能性があります。定型化されたログデータの場合、**ログパーサの設定**を通じてインデックスキーを活用した検索をお勧めします。

フィルター ⓘ

content == *select* × フィルターを入力してから「Enter」キーを押して追加

content == *select*

1. カテゴリを選択します。カテゴリ指定が必須です。
2. フィルター入力欄に content を選択後、検索キーワードを入力します。
例) content *select*
3. 🔍 検索ボタンをクリックしてログを照会します。ログ全体の一部を先に照会します。1回あたりの検索結果は最大1万件です。
4. スクロールを下げて下段の追加照会ボタンを選択すると、つづけて照会できます。

ログ検索

時間 2024/01/15 14:50 ~ 2024/01/15 15:50

category == AppLog && oid == 397157872 && content == *select*

AppLog (41,513)

全体の41,513件の内、10,500件を照会し、検索された結果は9999+件です。

AppStdErr (2,355)

AppStdOut (56,731)

ログをもっと照会しますか？
ログパーサーを追加することで、追加の照会なしで検索ができます。
ログパーサーの設定 追加照会

- ① 全体ログのうち、サーバー照会範囲当たり1万件ずつ照会します。サーバー照会範囲の場合、基本20万件ですが、ログ全体の量によって比率が変わることがあります。
- パーサ設定の詳細については、[次の文書](#)を参考にしてください。

フィルター修正

フィルターに値を入力し、入力した値をクリックすると、その値を修正できます。

フィルター ⓘ

🔍 oname == ote-account × key == aws.base:index × appender != fileAppender × フィルターを入力してから「Enter」キーを押して追加 ×

☐ 大小文字を区別する

fileAppender

searchValue*	検索値で始まるログを検索します。 ex key!=searchValue*
*searchValue	検索値で終わるログを検索します。 ex key!=*searchValue
searchValue	検索値を含むログを検索します。 ex key!=*searchValue*
*search*Value*	検索値を含むログを検索します。 ex key!=*search*Value*
re:searchValue	正規表現に一致するログを検索します。 ex key!=re:searchValue
**	検索語に該当するすべてのログを検索します。 ex key!=**

key == value 一致 key != value 除外 key == value 無効なタグです。 ⓘ ガイド表示

- ・ フィルター入力欄にテキストを書き換えて修正できます。
- ・ フィルター入力欄の下に表示されるガイドUIから推奨値を選択して修正できます。

検索キー(Search Key)

検索キーは、ログデータ内で希望する特定の値にアクセスするための識別子を意味します。検索キーに該当する実際のデータが検索値です。左側の②領域のタグはカテゴリ別にパーシング(parsing)された検索キーです。タグを選択してフィルターを入力します。オレンジ色タグはカテゴリで、青色タグは検索キーです。

例えば②領域のAppLogとAppStdOutはカテゴリであり、その下のoidなどのタグは、パーシング(parsing)された検索キーです。検索キーはログ>ログ設定のログパーサの設定タブでパーシングロジックを登録し、設定できます。

フィルター入力文法

タグは検索キーと検索値で構成されています。次の例で、検索キーは `exception`、検索値は `UnknownHostException` となります。この例は、収集したログデータのうち、IPアドレスとドメインアドレスが一致せず、サーバーをホストに接続できない場合に発生する例外(`UnknownHostException`)が含まれたログデータを照会します。



exception == UnknownHostException X

検索キーの種類

検索キーの種類	検索キーフォーマット	意味	検索キーと検索値の例	検索例
文字列キーワード	keyword	ファイル名	- キー: fileName - 値: /data/whatap/logs/yard.log	fileName:/data/whatap/logs/yard.log
数字キーワード	keyword.n	レスポンス時間	- キー: response_time.n - 値: 2945	response_time.n>=2945
予約語キーワード (辞書定義キーワード)	@keyword	TXID	- キー: @txid - 値: 85459614215434144	-

共通文法

文法の種類	説明	例
==searchValue	検索値と一致するログを検索します。	exception==RuntimeExceptionexception
!=searchValue	検索値を除いたログを検索します。	exception!=RuntimeException
*searchValue	検索値で終わるログを検索します。	word==*hello
searchValue*	検索値で始まるログを検索します。	word==hello*
searchValue	検索値として中間に含まれるログを検索します。	word==*hello*
*search*Value*	検索値として含まれるログを検索します。	word==*he*llo*

文法の種類	説明	例
re:{正規表現}	正規表現にマッチングされるログを検索します。	caller==re:^i\.w\.a\.w\.s\.v\.r\.
**	検索語に該当するすべてのログを検索します。	

検索キーが数字キーワード(keyword.n)の場合の文法

次の文法は検索キーが、`keyword.n` 形式の場合のみに対応します。

- 検索値は数字のみ使えます。
- .n キーワードの値にはPrefixを付けません。 .n 以外のキーワードは、すべてPrefixを付ける必要があります。

例) `+>searchValue` は有効ではありません。

文法の種類	説明	例
>searchValue	検索値よりも大きな値を含むログを照会します。	response_time.n>3000
>=searchValue	検索値よりも大きいか、同じ値を含むログを照会します。	response_time.n>=3000
==searchValue	検索値と同じ値を含むログを照会します。	response_time.n==3000
!=searchValue	検索値とは異なる値を含むログを照会します。	response_time.n!=3000
<searchValue	検索値より小さい値を含むログを照会します。	response_time.n<3000
<=searchValue	検索値より小さいか、同じ値を含むログを照会します。	response_time.n<=3000

個人情報の照会

ログの個人情報照会権限がある場合**個人情報の表示**  トグルボタンの活性化を通じて非識別化処理した個人情報を表示することができます。

- ❗ ・ ログの個人情報照会権限についての詳細は[次の文書](#)を参照してください。

- ・ ログ個人情報の非識別化機能についての詳細は次の文書を参照してください。

ログのタグオプション

ログのタグを選択すると、ドロップダウンメニューが表示されます。検索、除外検索、隣接ログ、トランザクション情報オプションが確認できます。

≡

全体90,699件の表示

過去順

最近順

キーワードを入力してください。

🔍

🔗

📄

🗑️

⚙️

▶	oname	タイムスタンプ	ログ	
▶	java-demo3	2025-04-17 09:17:13.018	@txid 7759974649447934516 pcode 8 onodeName node-0 oid -278185929 okind 1152715164 oname java-demo3	eException: random exception
▶	java-demo3	2025-04-17 09:17:13.018	検索 demo3 onodeName node-0 agentime 1744849031927 oid -278185929 category AppStdErr okindName dev-okind-0	rtual.web.Simula.exec(Simula.java:79)
▶	java-demo3	2025-04-17 09:17:13.018	除外検索 demo3 onodeName node-0 agentime 1744849031927 oid -278185929 category AppStdErr okindName dev-okind-0	ion: java.lang.RuntimeException: random exception
▶	java-demo3	2025-04-17 09:17:13.018	隣接ログ demo3 onodeName node-0 agentime 1744849031928 oid -278185929 category AppStdErr okindName dev-okind-0	rtual.web.VExec.doFilter(VExec.java:16)

ログタグ オプション メニュー	説明
コピー	該当するタグのログをコピーします。
検索	フィルターには、該当のタグを含む ('==') 条件で入力されます。
除外検索	フィルターには、該当のタグを除外する ('!=') 条件で入力されます。
隣接ログ	隣接ログの詳細画面では、選択したログのサーバーを対象に、選択したログと隣接する時間帯のログを照会できます。時間を選択して、周辺の時間帯に発生したログを確認することも可能です。基準となるログは青い背景で表示されます。
トランザクション 情報	ログに @txid や @mtid タグが含まれている場合は、トランザクション情報をクリックすると、トランザクション情報の詳細画面で内容を確認できます。

コンテンツハイライト

ログのコンテンツ内で目的のキーワードを簡単に識別できるよう、ハイライト機能を提供します。単一または複数のキーワードでフィルタリングできます。

1. ハイライトしたいキーワードをキーワード入力欄に入力します。
 - 例: `select`
2. 検索したキーワードがログリストのContent内でハイライト表示されます。
 -  **拡大表示** アイコンをクリックすると、**ログ** と **タイムスタンプ** を全画面で確認できます。
 - ハイライトされたキーワード間の移動はショートカットで操作できます。
 - `Enter` : 次の行へ移動
 - `Shift+Enter` : 前の行へ移動
 - `↑ / ↓` アイコン: 行の移動
 - `Cmd + F` (Mac) / `Ctrl + F` (Windows): キーワード検索

複数キーワード条件

複数キーワードでハイライトする場合は、次のように作成します。

入力文字列	説明	結果
<code>a b c</code>	各キーワードをスペースで区切ります。	a, b, c
<code>"Whatap is good."</code>	キーワードにスペースを含める場合は、 <code>"</code> または <code>"</code> で囲みます。	Whatap is good.
<code>"Whatap\\ is good."</code>	<code>"</code> で囲まれたキーワードに、 <code>\</code> を含める場合は、 <code>\\</code> と入力する必要があります。	Whatap\ is good.

ハイライトカラー設定

ハイライトするキーワードおよび色を設定できます。

- デフォルトでは、ログレベルに応じたハイライト（**FATAL**、**ERROR**、**WARN**、**SEVERE**、**CRITICAL**）が適用されています。
- 設定した内容はプロジェクト単位で保存されます。
 -  アイコンをクリックします。
 - ハイライト ウィンドウで、色を設定するキーワードを入力欄に追加してください。
 - 入力欄の左側の色をクリックすると、色を選択できます。

テーブル設定

カラム設定

3 領域の右で  **カラム設定** ボタンを選択します。カラムを追加したり、順序を設定できます。

- カラム追加

タグを選択してテーブルにカラムを追加できます。logカラム選択を解除する場合**ログ表示詳細設定**が確認できません。必ず1つ以上のカラムを選択します。

- カラム順序設定

カラムを追加すると**表示項目**に該当カラムが追加されます。希望のカラムをドラッグしてカラムの順序を変更します。

ログ表示詳細設定

3 領域右で  **ログ表示詳細設定** ボタンを選択します。デフォルトで**content**、**タグ**すべてチェックされており、両方の項目を表示します。**content**と**タグ**どちらかは必ず選択します。

チェックが解除されたアイテムはテーブルに表示されません。次のように**タグ**を解除する場合、テーブルでログの**タグ**は表示されません。

ログテーブルコンテンツタグ非表示

タグ管理一覧にタグを追加すると、追加した順にログのタグがソートされます。タグの順序はドラッグして変更できます。追加したタグを無効にすると、無効にしたタグはログのタグに表示されません。

ⓘ カラムおよびログ表示設定の適用範囲

- カラム設定とログ表示詳細設定は、ログTail、ログ検索、ログトレンドで使用できます。
- 同一プロジェクト内のログTail、ログ検索、ログトレンドは、カラム設定とログ表示詳細設定を共有します。

ログファイルのダウンロード

検索したログの結果を CSV または TXT ファイル形式でダウンロードできます。

1. 画面中央右側の  ダウンロード ボタンをクリックします。
2. CSVダウンロード または Logダウンロード をクリックして、希望の形式のファイルをダウンロードします。

ログの設定

ホーム画面 > プロジェクト選択 > ログ > ログ設定

ログ設定 の上部タブからログモニタリングを設定できます。 エージェント設定の確認、ログモニタリングの有効化設定、ログデータの保持期間および照会パスワードの設定、ログパーサーの登録、クイックインデックスの設定、個人情報のマスキング管理などを行うことができます。

ⓘ 権限

- **ログモニタリング機能の活性化** を使用するには、**プロジェクト修正** 権限が必要です。
- **ログエディター** 権限があれば、**ログモニタリング機能の活性化** のほか、**ログ設定** メニューを編集できます。

ログモニタリングを開始する

ログ設定

ログモニタリングの開始

ログプライマリパーサ設定

ログセカンダリーパーサの設定

高速インデックス設定

ログ長期保存統計

1

エージェントの設定とログモニタリング活性化

ログTail、ログトレンド、ログ検索機能で、アプリケーションに点在するログをまとめて確認することができます。

ログモニタリングの設定方法

[インストール型DB](#)[AWS RDS](#)

- #### XOSエージェントのインストール

LinuxファイルシステムにインストールされたDBのログは、WaタブのXOSエージェントで収集します。
下記のガイドリンクからインストールを進めてください。

[エージェントインストールガイド](#)
- #### xos.confファイルにオプション適用

xos.confファイルに次のオプションを適用してください。

```
tail -f /var/log/audit/audit.log
tail -f /var/log/audit/audit.log
tail -f /var/log/audit/audit.log
```

コピー
- #### ログモニタリング機能の活性化

設定完了後、ログ監視を有効にするトグルボタンをクリックしてください。

☒

2

ログモニタリングデータの設定

初期化

保存

ログ使用量

0ライン

ログ照会パスワード

☐ ログ照会の際にパスワードを入力するするには、オンにしてください。

データ保存期間

1日

カテゴリ別のデータ保存期間

カテゴリ	データ保存日	日間ログ数 ①	昨日のログ数 ①	週間ログ数 ①	月間ログ数 ①	予想ログ数 ①
WhatapEvent.whatap	1日	▼	4	0	4	4
DB.whatap	1日	▼	0	0	0	0

上段でログモニタリングの開始タブを選択します。 ガイド表示アイコンを選択すると、関連の案内画面に移動します。

エージェント設定とログモニタリングの有効化

- 1** 領域でエージェント設定を確認し、ログモニタリングの有効化または無効化を設定できます。

エージェントの設定確認

ログモニタリングを開始するために、エージェントのバージョンと設定情報を確認してください。 エージェント設定

に関する詳細は[次の文書](#)を参照してください。

ログモニタリングの有効化

ログモニタリングを有効化または無効化できます。**2. ログモニタリング機能の活性化**でログモニタリングの有効／無効を設定してください。

エージェントの設定とログモニタリング活性化

ログTail、ログトレンド、ログ検索機能で、アプリケーションに点在するログをまとめて確認することができます。

ログモニタリングの設定方法

[インストール型DB](#) AWS RDS

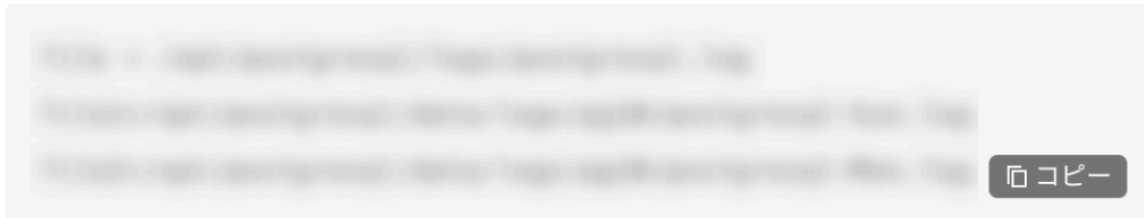
1. XOSエージェントのインストール

LinuxファイルシステムにインストールされたDBのログは、WaタブのXOSエージェントで収集します。
下記のガイドリンクからインストールを進めてください。

[エージェントインストールガイド >](#)

2. xos.confファイルにオプション適用

xos.confファイルに次のオプションを適用してください。



3. ログモニタリング機能の活性化

設定完了後、ログ監視を有効にするトグルボタンをクリックしてください。



-  ログモニタリングの有効化：トグルボタンをオンにするとログモニタリングが有効になり、有効化した日から

15日間は無料でお試しいただけます。

-  ログモニタリングの無効化：トグルボタンをオフにするとログモニタリングが無効になり、以降ログは保存されません。

ログモニタリングのデータ設定

2 領域でログ使用量が確認できます。またログ保存期間とログ照会パスワード設定を変更できます。

ログ参照パスワード

セキュリティを強化するため、**ログ照会パスワード**を設定してください。ログ検索パスワードの指定はオプションです。ログ検索パスワードを使用中の場合は、ログ画面に入るときは必ずパスワードを入力します。

パスワードを忘れた場合

ログ編集権限を持つユーザーは、**ログ設定** で新しいパスワードに変更できます。

ログ保存期間

共通に適用する基本(default) データ保持期間です。

- 最小1日から最大40日まで入力できます。未指定の場合、既定値は1日です。ログ保存期間の選択オプションの他に、ユーザーが希望する期間を入力できます。
- **ログ使用量**一覧で別に設定しない場合、このログデータの保存期間がデフォルトで適用されます。**ログ使用量** リストでカテゴリごとのデータ保持期間を設定し、**初期化** ボタンをクリックすると、デフォルトのデータ保持期間に初期化されます。

ログ使用量

ログ使用量一覧でカテゴリ別のログデータの保存期間を指定できます。ログ数は、当該期間中に蓄積されたログの件数を意味します。例えば、**日間ログ数**は、1日にたまったログ件数、**予想ログ数**は、データ保持日に今日のログ数を掛けたログ件数を意味します。

ログデータの保存期間を次のように指定できます。期間指定により、古いデータを削除してスペースを確保できます。

- **トライアルプロジェクト**

データ保存期間として1日、2日、3日を選択できます。

- **有料プロジェクト**

データの保存期間として1日、2日、3日、4日、5日、6日、7日、10日、30日、40日を選択できます。

- **保存量基準課金**

データの保存期間によってコストが異なります。

例、1日平均200万ログ件数が蓄積され、データ保存期間を3日に指定した場合、平均600万ログ件数が収集サーバーに保持され、課金の対象となります。

ログプライマリパーサ設定

ログ設定 上部の **ログプライマリパーサ設定** タブを選択して、収集されたログに対するパーサーを登録できます。

ログプライマリパーサは**GROK**と**JSON**パーサーを提供し、タイプ別ログ発生数の集計や特定ログの検索に必須のパーサーです。パターン条件に一致する検索キーと検索値を抽出し、パーシングされたキーはログのタイプ分類と検索インデックスに活用します。

- **GROK**：基本的には、正規表現ベースのパーシングに対応します。予約キーワードベースのパーシングを提供します。
- **JSON**：ログからJSONで出力された部分に対して一括パーシング機能を提供します。

✔ パーシングロジックが未登録時に検索可能なkey

category , oid , oname , okind , okindName , @txid , @login , httphost

パーサーで登録できない予約語

timestamp , message , pcode , category , content , logContent

- 次の予約語の場合、パーサーを登録してもインデックスは作成されません。

❗ ログパーサーに関する詳細は[こちらの文書](#)をご覧ください。

設定項目

設定値	説明	その他
カテゴリ	パターンを適用する対象のカテゴリです。	required
ログ検出条件	フィルターとして使用する検索キーと検索値を入力します。 - ログ検出条件に一致するログデータにのみパターンが適用されます。検出条件を入力しない場合、すべてのログにパターンが適用されます。	optional
パターン	ログをパースするためのパターンです。 - 作成したパターンに従ってログをパースし、インデックスを生成します。 GROKおよび正規表現の構文をサポートしています。	required

パーサー一覧

登録済みのパーサーを確認でき、追加や編集が可能です。適用順にパーサーが評価され、最初に一致したパーサーのみが適用されます。

- 右上の***追加するボタンをクリックすると、パーサ 追加画面が表示されます。
- 優先順位カラムの📌アイコンをドラッグしてパーサの設定手順を変更できます。
-  活性化トグルでパーサの活性化有無を指定できます。
-  修正及び  削除アイコンで登録したパーサを修正または削除できます。

パーサー登録

以下はパーサーを登録する共通の手順です。

1. ログプライマリパーサ設定 で + 追加する ボタンをクリックしてください。
2. パーサ 追加画面でパーサを選択してください。

❗ パーサーおよびパターンの登録方法

- [GROK パーサおよびパターン登録](#)
- [JSON パーサおよびパターン登録](#)

3. パーサーを選択すると、**パターン登録**ボタンが有効になります。**パターン登録**ボタンをクリックした後、**パーサーシミュレーション**ウィンドウでパターンとログを入力します。

a. **パターン**を入力してください。

b. **ログのカテゴリ**と**シミュレーション用のパースログ**を入力してください。**ログ検索**ボタンをクリックした後、検索結果の一覧から目的のログの右側にある**選択**ボタンをクリックすると、ログが自動的に入力されます。

❗ **ログ検索**を使用して、収集されたログの中からサンプルログを選択できます。選択したログはシミュレーション入力欄に自動的に反映され、コピーや貼り付けなしですぐにテストできます。

c. 登録するパターンが正しいかどうかを確認するには、**シミュレーション**ボタンをクリックして、シミュレーションとパターンのパフォーマンスを測定してください。

❗ シミュレーションと性能測定についての詳細は、[次の文書](#)を参照してください。

4. 正常にパースされるかどうかシミュレーション結果を確認した後、**パターン適用**ボタンをクリックしてください。

5. **追加**ボタンをクリックすると、パーサーが追加されます。

❗ 個人情報の非識別化

パーサー登録時に検索キー名を **検索キー.p** に指定すると非識別化の対象になります（例：**myname.p** **）。WhaTapは**個人情報の非識別化**タブで非識別化を管理する方式を勧めます。詳細は**個人情報の非識別化**を参照してください。

GROKパーサパターン登録

基本文法は、`%{SYNTAX:SEMANTIC}` です。GROKパーサの詳細は、[次の文書](#)を参照してください。パターン登録とシミュレーションが必須です。

- SYNTAX

GROK定義パターンです。

- SEMANTIC

パーシングされたデータに割り当てるキーです。

❗ SEMANTICでは、予約語ではなく、組み合わせでの使用を推奨します。

JSONフォーマットのパーサパターン登録

ログ全体または一部がJSON形式で出力される場合、JSONフォーマットのパーサを使用して、JSONに出力された部分をパーシングできます。ログのうちJSON形式として出力された部分を検出するために、**Prefix**、**Postfix**オプションを組み合わせでログのどの部分をJSONとして認識してパーシングするかを指定します。JSONパーサの詳細は、[次の文書](#)を参照してください。シミュレーションが必須です。

オプション	説明
Prefix	JSON文字列の先頭の前にある文字列を指定します。未指定時、ログ出力文の先頭からJSON文字列として識別します。
Postfix	JSON文字列の最後の部分の文字列を指定します。未指定時、ログ出力文の最後までをJSON文字列として識別します。
Ignore	JSON出力部分のうちキー抽出を除外するフィールドを指定します。

- 登録例

Log


```
[2022-10-25 10:15:34:145]...(改行)
Request: {"key1": "value1", "key2": "value2", ...}(改行)
Response: {"key3": "value3", "key4": "value4", ...}
```

例のように流入されるログがRequest JSON、Response JSONをすべてパーシングする場合、次の2つのパターンを登録します。

- Requestパーシング用のパターン

"Request : "と"Response"間の文字列 {"key1": "value1", "key2": "value2", ...} 対象

- Responseパーシング用のパターン

"Response : "からログの最後までまでの文字列 {"key3": "value3", "key4": "value4", ...} 対象

• JSONカスタムパターンの登録

ログの一部がJSON形式で出力される場合、JSONで出力された部分をカスタムパーサを使用してパーシングできます。パターンを次のように入力します。

```
io.whatap.logsink.parser.JsonFormatParser{}
```

ログのうちJSON形式として出力された部分を検出するために、**Prefix**、**Postfix**オプションを組み合わせることでログのどの部分をJSONとして認識してパーシングするかを指定します。

JsonFormatParser{} の {} にオプションを指定します。

- 登録例

Log

```
[2022-10-25 10:15:34:145]...(改行)
Request: {"key1": "value1", "key2": "value2", ...}(改行)
Response: {"key3": "value3", "key4": "value4", ...}
```

例のように流入されるログがRequest JSON、Response JSONをすべてパーシングする場合、次の2つのパターンを登録します。

- Requestパーシング用のパターン

"Request : "と"Response"間の文字列 {"key1": "value1", "key2": "value2", ...} 対象

```
io.whatap.logsink.parser.JsonFormatParser {prefix:"Request :",postfix:"Response"}
```

- Responseパースング用のパターン

"Response："からログの最後までまでの文字列 {"key3":"value3","key4":"value4",...} 対象

```
io.whatap.logsink.parser.JsonFormatParser {prefix:"Response:"}
```

パーサーシミュレーションと性能測定

パーサーシミュレーション後、パターンを登録できます。パフォーマンス測定は、シミュレーション実行対象文字列に対してパーサーの反復パースング所要時間を測定します。

1. パターンとログを入力します。ログ検索ボタンをクリックした後、検索結果の一覧から目的のログの右側にある選択ボタンをクリックすると、ログが自動的に入力されます。
2. シミュレーションボタンをクリックして、登録しようとするパターンでパースングに成功することを確認します。
3. シミュレーションの成功時シミュレーション結果と性能測定の結果を照会できます。
4. シミュレーション後、パターン適用ボタンをクリックすると、選択したパーサーに入力したパターンが適用されます。

パースングの成功

パースング ロジックを登録し、キー(key)が生成されたら、ログを照会する際に該当キーでパースングされた値が追加されます。次のログTailメニュー例のように、パースングされたキーと値が追加されます。

▶	Timestamp	ログ
▶	2022-08-17 14:28:00.612	oname dev949400-8093 onodeName node-1 oid 413390913 category AppStdOut okindName dev-okind-1 load 52

パースングされたキーはログTail、ログ検索、ログトレンドで確認できます。

ログセカンダリーパーサ設定

ログ設定メニューの上部にあるログセカンダリーパーサの設定タブを選択してログパーサを登録できます。

ログのセカンダリーパーサは、プライマリーパーサ(GROKまたはJSON)で抽出した値を加工して統計データや新しいフィールドを生成します。HTTPステータスコードベースの統計パーサとともに、キーワードフィルタ、フィールド加工、応答時間分類、GeoIP変換など、さまざまなプロセッサを提供します。

❗ ログのセカンダリーパーサは、1次でパースされた結果に対して、追加での2次パース機能を提供します。セカンダリーパーサを使用するには、**プライマリーパーサ**を登録する必要があります。

パーサー一覧

ログ設定

ログモニタリングの開始

ログプライマリーパーサ設定

ログセカンダリーパーサの設定

高速インデックス設定

ログ長期保存統計

1時間の統計ウィジェットのデータ設定のログ

ログセカンダリーパーサの設定

Grok、またはJSONパーサがすでに解析された場合に使用できるパーサです。

+ 追加する

保存

優先順位	パーサ	カテゴリ	ログ検出条件	パターン	活性化	
0	4xx、5xx状態コードパーサ	AppStdErr	すべてのログ		☒	
1	状態コード成功率パーサ	AppLog	すべてのログ		☒	

登録したパーサを確認でき、追加および編集が可能です。

パーサの追加、順序変更、活性化トグル、修正および削除などの操作方法は[プライマリーパーサー一覧](#)と同一です。

パーサ登録手順

次に、パーサ登録の共通の手順を説明します。

× パーサ 追加

パーサ*

WhaTap Monitoringには、サービスの信頼性に影響を与える可能性があるパーサーを無効にできる権限があります。

パーサを選択してください。



除外する状態コード

パーサを選択してください。

カテゴリ*

同じカテゴリに複数のパーサーが登録されている場合、最初に一致するパーサーのみが適用されます。

カテゴリを選択してください。

ログ検出条件

フィルタを入力しなかった場合、または検索キー／値を1つだけ入力した場合、そのパターンはすべてのログに適用される。

検索キー

検索値

追加

1. **+追加する**ボタンを選択すると**パーサ 追加**画面が表示されます。
2. **パーサ**選択画面からパーサを選択します。各パーサの設定項目の詳細については、次の文書を参照してください。
 - [4xx、5xx状態コードパーサ 設定項目](#)
 - [状態コード成功率パーサ 設定項目](#)
3. **除外する状態コード**を入力します。
4. **カテゴリ**選択画面からカテゴリを選択するか、直接入力します。
5. **ログ検出条件**を選択するか、直接入力します。
6. **追加**ボタンを選択してパーサを登録します。

4xx、5xxステータスコードパーサ設定項目

4xx、5xx状態コードパーサ はプライマリパーサでパースされたstatus値を基に4xx、5xx件数データを生成します。特定のステータスコードを除外する場合は、除外するステータスコードを入力してください。

設定項目

設定値	説明	その他
カテゴリ	4xx、5xxの件数データを生成するカテゴリです。	required
ログ検出条件	フィルターとして適用する検索キー、検索値を入力します。ログ検出条件を満たすログデータに対して、4xx、5xxのデータを生成します。ログ検出条件を入力しない場合は、すべてのログのデータを生成します。	optional
除外するステータスコード	統計データの作成時に除外するステータスコードです。入力がない場合は、4xx・5xxに該当するすべてのエラーステータスコードを対象に4xx、5xxの件数データを生成します。	optional

statusパーサの登録例

ログプライマリパーサ設定 収集したログに対してパーサを登録することができます。パーサは、順番通り適用され、最初にマッチしたパーサのみが適用されます。 + 追加する 保存						
優先順位	パーサ	カテゴリ	ログ検出条件	パターン	活性化	
0					☐	 
1					☒	 
2	GROK	AppLog	すべてのログ	%{NUMBER:status}	☒	 

流入されるログが `{"msg":"message","status":404}` で、例のようにGROKパーサでstatusをパースする場合、`status: 404` のようにパースされます。statusが正しくパースされることを確認したら、4xx、5xxのステータスコードのパーサから除外するステータスコードを登録します。

データ参照

パーサをすべて登録すると、統合Flexボードに移動してログ 4XX、5XX件数ウィジェットを作成します。

ウィジェットテンプレート

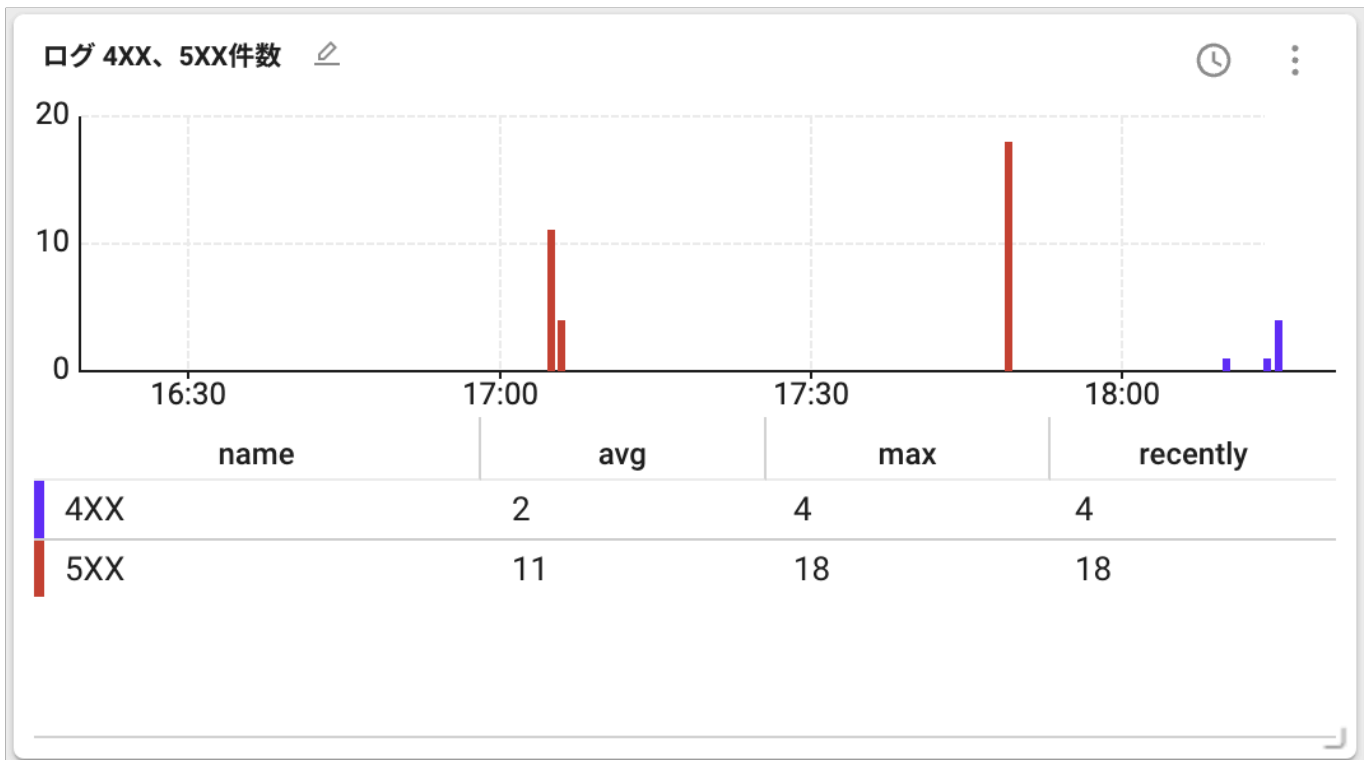
⇒ すべてのメトリックス

4xx

🔍

📊 ログ 4XX、5XX件数

ウィジェットを作成すると、次のようにデータを確認できます。



- **avg**：照会期間のデータで平均値です。
- **max**：照会期間のデータで最大値です。
- **recently**：照会期間のデータで最後の値です。

ステータスコード成功率パーサ設定項目

状態コード成功率パーサ はプライマリーパーサでパースングされたstatus値を基にHTTPリクエスト成功率データを生成します。 特定のステータスコードを除外する場合は、除外するステータスコードを入力してください。 statusパー

シングに関する内容は、[4xx、5xxステータスコードパーサ設定項目](#)を参照してください。

設定項目

設定値	説明	その他
カテゴリ	要請成功率データを生成するカテゴリです。	required
ログ検出条件	フィルターとして適用する検索キー、検索値を入力します。ログの検出条件を満たすログデータのみで要請成功率データを生成します。ログ検出条件を入力しない場合は、すべてのログのデータを生成します。	optional
除外するステータスコード	要請成功率データを生成する際に除外するステータスコードです。入力がない場合は、2xx・3xxに該当するすべての成功ステータスコードを対象に要請成功率のデータを生成します。	optional

データ参照

パーサをすべて登録したら、**統合Flexボード**に移動し、ログ要請の成功率ウィジェットを作成します。

ウィジェットテンプレート

⇌ すべてのメトリックス

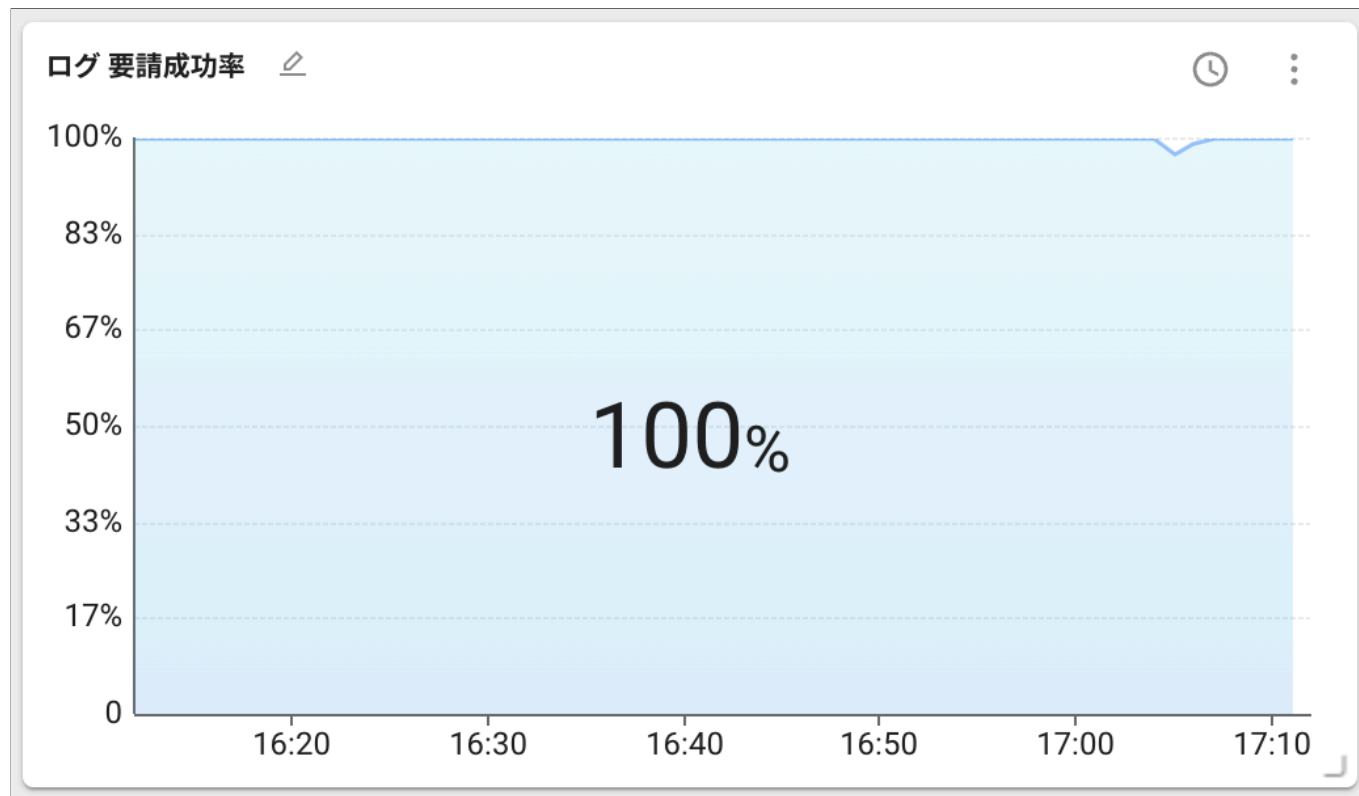
成功率

🔍

📉

ログ 要請成功率

ウィジェットを作成すると、次のようにデータを確認できます。



チャート上のデータは、照会期間に関する統計を示します。統計方法としては最近値、最大値、平均値などを選択できます。最近使った値がデフォルトで選択されています。

セカンダリーパーサプロセッサ

ログのセカンダリーパーサは、4xx、5xx状態コードパーサ と状態コード成功率パーサ のほかにも、次のようなフィールド加工プロセッサを提供します。

ログのセカンダリーパーサプロセッサ一覧

プロセッサ	説明
すべてのキーワード含有フィルタ	指定したすべてのキーワードが含まれるログにフラグフィールドを追加します。
一部キーワード含有フィルタ	指定したキーワードのうち1つでも含まれるログにフラグフィールドを追加します。

プロセッサ	説明
フィールド名変更	既存のフィールド名を別の名前に変更します。
ステータスコードグループ分類	HTTPステータスコードを2xx/3xx/4xx/5xxグループに自動分類します。
URL静的/動的分類	URLに拡張子(.)があれば静的コンテンツ、なければ動的サブリットに分類します。
URLパラメータ除去	URLから?以降のクエリパラメータを除去した値を保存します。
応答時間分類	応答時間(ms)をfast/normal/slow/critical区間に自動分類します。
JSONフィールド抽出	JSON文字列フィールドから特定のキーを抽出して新しいフィールドとして保存します。
GeoIP変換	IPアドレスを国/都市情報に変換します。
直接入力	FQNクラス名でプロセッサを直接登録します。

すべてのキーワード含有フィルタ

指定したすべてのキーワードが含まれるログにフラグフィールドを追加します。

設定値	説明	デフォルト値
入力フィールド	キーワードを検索するフィールド名	content
保存フィールド	結果を保存するフィールド名	has_str_all
keyword	検索するキーワード一覧(カンマで区切って入力)	required

活用例: 入力フィールドを content 、キーワードを error, timeout, database に設定すると、3つのキーワードがすべて含まれるログにのみ has_str_all フィールドが追加されます。

一部キーワード含有フィルタ

指定したキーワードのうち1つでも含まれるログにフラグフィールドを追加します。

設定値	説明	デフォルト値
入力フィールド	キーワードを検索するフィールド名	content
保存フィールド	結果を保存するフィールド名	has_str_any
keyword	検索するキーワード一覧(カンマで区切って入力)	required

活用例: 入力フィールドを content 、キーワードを reboot, shutdown, restart に設定すると、キーワードのうち1つでも含まれるログに has_str_any フィールドが追加されます。

フィールド名変更

既存のフィールド名を別の名前に変更します。対象のフィールド名がすでに存在する場合は上書きしません。

設定値	説明	デフォルト値
元のフィールド	変更する既存のフィールド名	required
新しいフィールド名	変更後のフィールド名	required

ステータスコードグループ分類

HTTPステータスコードを status_2xx 、 status_3xx 、 status_4xx 、 status_5xx グループに自動分類します。

設定値	説明	デフォルト値
入力フィールド	ステータスコードが入ったフィールド名	status
保存フィールド	グループ分類結果を保存するフィールド名	status_nxx

活用例: status 値が 404 の場合、 status_nxx フィールドに status_4xx が保存されます。

URL静的/動的分類

URLに拡張子(.)が含まれていれば静的コンテンツ、含まれていなければ動的サブレットに分類します。

設定値	説明	デフォルト値
入力フィールド	URLが入ったフィールド名	url
保存フィールド	分類結果を保存するフィールド名	url_type

URLパラメータ除去

URLから ? 以降のクエリパラメータを除去したパスのみを保存します。

設定値	説明	デフォルト値
入力フィールド	URLが入ったフィールド名	url
保存フィールド	パラメータ除去結果を保存するフィールド名	url_without_params

活用例: /api/v1/users?page=1&size=20 → url_without_params フィールドに /api/v1/users が保存されます。

応答時間分類

応答時間(ms)を区間別に自動分類します。区間のしきい値を直接設定できます。

設定値	説明	デフォルト値
入力フィールド	応答時間(ms)が入ったフィールド名	elapsed
保存フィールド	分類結果を保存するフィールド名	response_class
Fast上限(ms)	fast区間の上限値	100
Normal上限(ms)	normal区間の上限値	500

設定値	説明	デフォルト値
Slow上限(ms)	slow区間の上限値	2000

デフォルト値基準の分類結果は次のとおりです。

応答時間	分類
0 ~ 100ms	fast
101 ~ 500ms	normal
501 ~ 2000ms	slow
2001ms以上	critical

活用例: ログ探索で `response_class: critical` でフィルタリングすると、応答時間が2秒を超えるログのみを照会できます。

JSONフィールド抽出

プライマリーパーサで抽出したフィールドにJSON文字列が含まれる場合、該当のJSONから特定のキーを抽出して新しいフィールドとして保存します。

設定値	説明	デフォルト値
入力フィールド	JSON文字列が入ったフィールド名	payload
抽出するキー	抽出するJSONキー一覧(カンマで区切って入力)	required
保存プレフィックス	抽出されたキー名の前に付けるプレフィックス(例: <code>json_</code>)	optional

- ① ログ全体がJSONの場合、1次JSONパーサが自動的にパーシングします。このプロセッサは、**GROK 1次パーサ**使用時にテキストログ内に含まれるJSON部分を追加でパーシングする際に便利です。
- パフォーマンス保護のため、JSONフィールドサイズが10KBを超えるとパーシングをスキップします。

活用例

1次GROKパーシング結果

```
payload = {"user": "alice", "action": "purchase", "product": "laptop", "amount": 45000}
```

入力フィールド `payload`、抽出するキー `user, action`、保存プレフィックス `pay_` を設定すると、次のように抽出されます。

JSONフィールド抽出結果

```
pay_user = alice
pay_action = purchase
```

GeoIP変換

IPアドレスを国および都市情報に変換します。IP2Locationデータベースを活用して、アクセスログの地域別分析が可能です。

設定値	説明	デフォルト値
入力フィールド	IPアドレスが入ったフィールド名	<code>client_ip</code>
保存プレフィックス	結果フィールド名のプレフィックス	<code>geo</code>

結果として `{プレフィックス}_country`、`{プレフィックス}_city` の2つのフィールドが生成されます。

活用例: 入力フィールド `client_ip` の値が `223.130.195.200` で、保存プレフィックスが `geo` の場合:

GeoIP変換結果

```
geo_country = KR
geo_city = Seongnam
```

直接入力

一覧にないプロセッサをFQN(Fully Qualified Name)クラス名で直接登録できます。

直接入力形式

```
io.whatap.logsink.processor.ClassName{"param1":"value1","param2":"value2"}
```

- ❗ クラス名とパラメータ形式が正確である必要があります。誤った入力の場合、プロセッサが動作しないことがあります。

セカンダリーパーサのチェイニング

同一カテゴリに複数のセカンダリーパーサを登録してチェイニングできます。各プロセッサは登録順に実行されます。

元のログ

```
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
```

順序	パーサ	入力 → 出力
1次	GROK	timestamp, level, host, client_ip, url, status, elapsed, payload, content
2次-1	JSONフィールド抽出	payload → user, action
2次-2	応答時間分類	elapsed → response_class (slow)
2次-3	ステータスコードグループ分類	status → status_nxx (status_2xx)
2次-4	URLパラメータ除去	url → url_without_params (/api/v1/orders)
2次-5	GeoIP変換	client_ip → geo_country (KR), geo_city (Seoul)

- ❗ セカンダリーパーサの**カテゴリ**は、プライマリーパーサと同一に設定する必要があります。異なるカテゴリに設定すると、該当のログにセカンダリーパーサが適用されません。

高速インデックス設定

ログ設定メニューの上段から高速インデックス設定タブを選択します。大量のログを収集すると、ログ検索性能が低下する場合があります。よく使う検索条件は、インデックス(index)で生成すると、ログ検索がより速くなります。設定項目は次のとおりです。

設定値	必須の有無	説明
カテゴリ	必須	高速インデックスを設定するカテゴリ
検索キー	必須	高速インデックスを設定する検索キー
大文字と小文字を区別しない	オプション	大文字と小文字の区別するかどうか
ルール	必須	* を少なくとも 1 つ以上を含める必要あり
活性化有無	必須	活性化または非活性化の設定(既定値 true)

ログ設定のインポート・エクスポート

パーサ設定、高速インデックス設定、フィルター設定の内容をJSONファイル形式で保存し、他のプロジェクトからJSONファイルをインポートして適用できます。プロジェクトごとに何度も繰り返し設定を作成する手間を減らします。

- 1つのプロジェクトにパーサ設定、高速インデックス設定、フィルター設定を追加します。
- 各設定タブで画面右上にJSON  ボタンを選択します。
- JSONエクスポート画面が表示されると、画面右上にある抽出ボタンを選択します。
- JSON設定ファイルがユーザーPCに保存されます。
- 他のプロジェクトに移動してから、ログ > ログ設定メニューに移動してください。
- まず、JSON設定ファイルをエクスポートした設定タブを選択し、 ボタンを選択します。
- ファイル選択画面が表示されたら、ユーザーのPCに保存したJSON設定ファイルを選択します。

8. JSONインポート画面が表示されたら、設定内容を確認し、**リストに追加する**または**上書きボタン**を選択します。
9. 画面右上の**保存ボタン**を選択します。

❗ JSON設定ファイルをインポートして、**保存ボタン**を選択しないと、インポートされた設定は保存されません。

ログ長期保存統計の設定

ログ設定メニューの上段から**ログ長期保存統計**タブを選択します。ログデータは、非常に大容量のため、長期間保存することは難しいです。したがって、ログ統計データの設定機能を使用して**特定の条件を満たすログデータが5分ごとに何件収集されたかに関する情報を保存**できます。長期間にあたって、実際のログデータが削除されても、条件を満たすログがどれだけ収集されているかの推移は確認できます。

ログ長期保存統計の追加

×

ログ長期保存統計 追加

統計キー*

統計キーを入力してください。

データ保存期間
(ディスク使用量)

15日 (約 3 MB) ▼

カテゴリ*

カテゴリを選択してください。

ログ検出条件*

まず、カテゴリを選択してください。 ▼

まず、カテゴリを選択してください。

除外 ☐

大小文字を区別する ☒

+ 追加する

追加

ログ長期保存統計タブで+ 追加するボタンを選択するとログ長期保存統計 追加画面が表示されます。+ 追加するボタンでルールを追加したり、作成したルールを**-**アイコンで削除できます。

設定項目

フィールド	説明
カテゴリ	ルールが適用されるカテゴリです。
統計キー(key)	ルールを満たすログが発生した場合、保存するキーとして同じキーを重複で設定できません。
ログ検出条件	ログ統計データを生成する条件です。この条件を満たすログがどのくらい収集されたかで統計データを生成します。
除外	除外をチェックすると、入力した条件を満たさない値で統計データを作成します。
大小文字を区別する	入力したログ検出条件の値に対して大文字と小文字の区別有無を指定します。
活性化有無	活性化または非活性化の設定(既定値 true)

例

次のように設定を追加した場合、収集されたログのうち、statusが 200 , 300 のログを対象にTotalCountのキーで統計データを作成します。

ログ長期保存統計 指定した条件を満たすログがどれ程収集されたかの統計データが生成できます。 					
カテゴリ	統計キー	ログ検出条件	データ保存期間 (ディスク使用量)	活性化	
AppLog	TotalCount	status 200, 300	15日 (約 3 MB)	☐	 
AppLog	TotalCount	status 200, 300	15日 (約 3 MB)	☐	 
AppLog	TotalCount	status 200, 300	15日 (約 3 MB)	☒	 
AppLog	TotalCount	status 200, 300	15日 (約 3 MB)	☒	 

データ参照

1. 統合Flexボードのウィジェットテンプレートでログ長期保存統計を検索し、ウィジェットを作成します。



ウィジェットテンプレート

⇒ すべてのメトリックス

ログ長期保存統計 🔍

📈 ログ長期保存統計

2. データを照会するカテゴリと統計キーを指定して適用ボタンを選択します。



ログ長期保存統計 ✎

カテゴリ変更
AppLog ▼

統計キー
TotalCount ▼

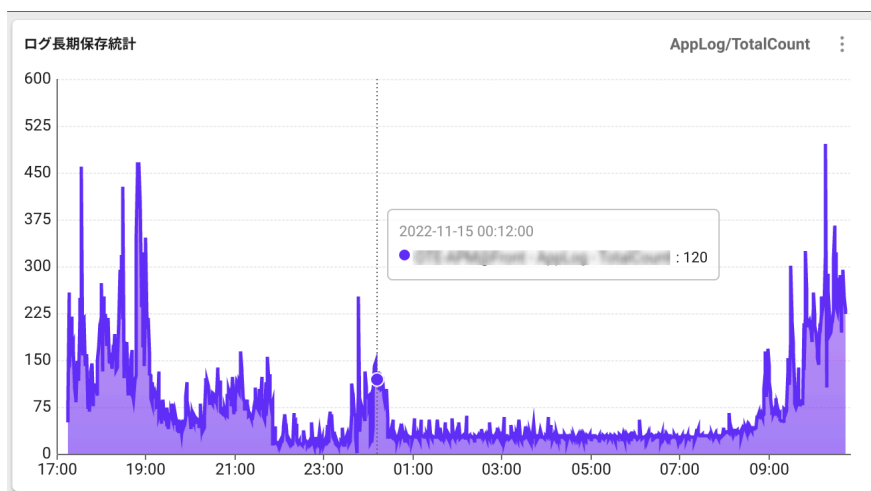
キャンセル 適用

ⓘ カテゴリ、キーを選択してください。

該当ウィジェットに表示するデータ収集のためにはログ長期保存統計 設定が必要です。

当該設定をしていない場合は、
[ガイド](#)を参考してログ長期保存統計 設定をしてください。
[ログ設定](#)

3. 追加した設定でログ長期保存統計データを次のように確認できます。



個人情報の非識別化

個人情報の非識別化ではログデータ内の個人情報をマスキング処理したり、安全な値に置換できます。

- 暗号化の保存：指定された機密な情報は暗号化され、データベースに保存されます。
- マスキング処理：ログ画面に表示される時、機密な情報は基本的にマスキング(***)処理されて表示されます。
- 置換処理：機密な情報をユーザーが指定した安全な値に変更します。

個人情報の分類

一般的に個人情報として分類する項目は次のとおりです。一般の個人情報以外にも機密な情報として処理すべき項目があれば個人情報の非識別化機能を通じて、ログ内で必要な項目をマスキング処理または置換処理してください。

- 個人識別番号
- パスポート番号
- 運転免許証番号
- 外国人登録番号
- クレジットカード番号
- 銀行口座番号

- 生体情報

要求権限

非識別化された個人情報はマスキング処理されて表示されます。ログの個人情報照会権限が付与されたユーザーのみ個人情報の表示トグルを通じて元のデータを確認できます。非識別化の対象検索キーの追加及び修正は、ログ編集権限が必要です。

❗ ログの個人情報照会権限及びログ編集権限の詳細は次の文書を確認してください。

個人情報の非識別化設定

新規のログデータにマスキング処理を適用して機密な情報を保護できます。

ログ設定

ログモニタリングの開始 ログプライマリパーサ設定 ログセカンダリーパーサの設定

個人情報の非識別化 個人情報が含まれた検索キーを追加すると、当該データを非識別化します。

活性化 カテゴリ

AppLog @txid agenttime @mt

ログ個人情報の非識別化チェック 以前に積み重ねられたログを非識別化します。

時間 < 2025/01/21 16:48 ~ 2025/01/21 17:48 1時 >

カテゴリ カテゴリを選択してください。

個人情報項目

個人情報パターン

置換を適用するかどうか

置換値

点検開始 チェックを中止

× 個人情報の非識別化 追加

カテゴリ * カテゴリを選択してください。

検索キー * 暗号化保存およびマスキング処理する検索キーを選択します。

暗号化する検索キーを入力してください。

追加

1. 個人情報の非識別化タブで追加するボタンを選択します。
2. 右側の個人情報の非識別化 追加画面でカテゴリを選択し、マスキング処理する検索キーを入力します。

3. 追加ボタンをクリックして設定を追加します。
4. 個人情報の非識別化一覧で非識別化項目を確認した後、右上の保存ボタンを押して現在の設定を保存してください。

個人情報の非識別化 個人情報が含まれた検索キーを追加すると、当該データを暗号化して保存し、マスキングします。 + 追加する 保存			
活性化	カテゴリ	検索キー	修正
☐	VirtualLog	age.n	編集 削除
☒	VirtualLog	status	編集 削除
☒	AppLog	agenttime	編集 削除

- 非識別化処理された検索キーの場合、ログコンテンツ内の **検索キー.pii** 形式で出力されます。
- 個人情報の非識別化一覧内の活性化カラムで非識別化項目を ☒ 活性化にできます。
- 個人情報の非識別化一覧内の編集カラムで [修正](#) 及び [削除](#) できます。

✔ **ログプライマリパーサ設定**タブで検索キー名を **検索キー.p** に指定すると、非識別化対象として自動登録されます。自動登録された項目は**個人情報の非識別化**タブで**カテゴリ**選択後、**指定された検索キー**で確認または解除できます。

パーサーで .p 接尾辞を指定する方式よりも、**個人情報の非識別化**タブで直接設定することを推奨します。

個人情報の表示

非識別化されたデータは**ログTail**、**ログトレンド**、**ログ検索メニュー**でマスキング状態で表示されます。**ログの個人情報照会権限**のあるユーザーはログ一覧内の**個人情報の表示** ☒ ボタンをクリックすると元のデータを確認できます。

▼	java-demo2	2024-12-16 16:06:49.988	area river lprivacy 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 lprivacy 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 lprivacy 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 lprivacy 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 lprivacy true status.pii 200 pcode 8 city daegu onodeName node-1 myname.p song Waiting does not have to aim for a meeting.

❗ **個人情報の非識別化タブ**で指定した検索キーは、ログメニュー内のログコンテンツから **検索キー.pii** 形式で表示されます。

例: **status.pii** 例 ***

ログの個人情報の非識別化点検

❗ **ログ個人情報の非識別チェック機能**を通じて既存のデータを置き換える場合、そのデータは元に戻すことができません。

ログ個人情報の非識別チェックは既存のログデータ内の個人情報を識別し、ユーザーが指定した**置換値**に置き換えます。

対象キー指定による置換

対象キーを指定すると、該当キーに保存された値のみを置換し、残りのログ内容はそのまま維持します。対象キーはログの **field** 値と **tag** 値の両方に適用されます。対象キーを指定しない場合、従来と同様にログコンテンツ全体から個人情報を識別して置換します。

❗ 置換は保存されたログデータにのみ適用されます。検索インデックスには、置換後も元の値が残っている場合があります。

対象キーは、ログ個人情報の非識別化点検フォームの**対象キー**入力欄で指定し、画面では **field** と **tag** を区別せず、1つの対象キーとして入力します。

ログ個人情報の非識別チェック

収集されてあるログを非識別化します。

📅

✓ チェック履歴の確認

時間

< 2025/03/21 12:26 ~ 2025/03/21 13:26 1時 📅 >

カテゴリ

VirtualLog

個人情報項目

個人識別番号

個人情報パターン

(?<!\w)\d(2)(0[1-9]|1[0-2])\{0-2\}[0-9]{3[0-1]}-[1-4]\d(6)(?\w)

置換の適用有無

☐

置換値 ①

*****_*****

カテゴリ

VirtualLog

検索開始時刻

2025/03/21 12:26

検索終了時刻

2025/03/21 13:26

個人情報項目

個人識別番号

個人情報パターン

(?<!\w)\d(2)(0[1-9]|1[0-2])\{0-2\}[0-9]{3[0-1]}-[1-4]\d(6)(?\w)

置換値

*****_*****

パターンのマッチング結果

0 ✓ 結果の確認

チェック作業の開始時刻

2025/03/21 13:28

チェック作業の終了時刻

2025/03/21 13:28

置換の適用有無

No

チェック作業の結果

SUCCESS

1. 開始時間と終了時間を設定してください。
2. カテゴリを設定します。
3. 個人情報項目を例の中から選択または**直接入力**を選択します。

① 直接入力選択時に個人情報を識別するための正規表現式パターンを直接入力します。

- 置換の適用有無トグルで個人情報が識別された場合、該当情報の置換有無を選択します。
- 既存のログで正規表現パターンに識別された文字列を変更する置換値を入力します。

- ❗ 置換値は元の値より長くしてはなりません。
- 置換の適用有無トグルの活性化時に、必ず置換値を入力する必要があります。

6. 条件を入力した後、**チェック開始ボタン**を選択します。

① 点検中の場合はいつでも**チェック中止**ボタンで点検過程を中止できます。

7. 置き換えを適用する時個人情報の非識別チェックの開始確認画面で置き換え入力後確認ボタンをクリックします。

個人情報の非識別チェックの開始



選択した個人情報パターンに一致する値を置き換えます。

①置き換え後はデータの復元ができません。

置き換えが必要な場合は "置き換え" を直接入力してください。

8. 点検作業の完了後、例の画像のように右側から**チェック作業の結果**を確認できます。

- **カテゴリ**：点検されたカテゴリ情報を表示します。
- **検索開始時刻及び検索終了時刻**：点検が実行された時間範囲を表示します。
- **個人情報項目**：点検された個人情報の項目を表示します。
- **個人情報パターン**：使用された正規表現式のパターンを表示します。
- **置換値**：適用された置換値を表示します。
- **パターンのマッチング結果**：識別された個人情報の数を表示します。 **結果の確認**をクリックするとサンプルログを確認できます。
- **チェック作業の結果**：個人情報識別された結果を表示します。

SUCCESS (点検成功)、 FAILED (点検失敗)、 STOP (点検中止)、 TIMED_OUT (時間超過による点検終了)

チェック履歴の確認

ログ個人情報の非識別化点検を行った後、過去に行われた点検結果を照会するには、上段右側で**チェック履歴の確認**ボタンをクリックします。置き換え履歴、点検時点、適用された個人情報項目などを追跡できます。また**チェック履歴の確認**画面の上段右側の **CSV** ボタンをクリックすると CSV ファイル形式でそのリスト情報をダウンロードできます。

 CSV

	パターンのマッチング ...	チェック作業の開始時刻	チェック作業の終了時刻	置換の適用有無	チェック作業の結果
***	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

個人情報のパターン

既存のログデータから個人情報を識別する際に使用される正規表現式のパターンの例は以下の通りです。

個人情報 の パターン	個人情報の例	正規表現パターン	置換される 値
個人 識別 番号	123456-1234567	(?!w)d{2}(0[1-9] 1[0-2])([0-2][0-9] 3[0-1])-[1-4]d{6}(?!w)	*****_ *****
パ	M1234567	([MSRODTC][0-9]{7}\b)	*****

個人情報のパターン	個人情報の例	正規表現式パターン	置換される値
スポーツ番号			
運転免許証番号	01-23-456789-00	(?<!w)d{2}-d{2}-d{6}-d{2}(?!w)	***_**_ *****_**
外国人登録番号	987654-1234567	(?<!w)d{6}-?[5-8]d{6}(?!w)	*****_ *****
クレジット	1234-5678-9012-3456	(?<!w)d{4}-d{4}-d{4}-d{4}(?!w)	****_****_ ****_****

個人情報のパターン	個人情報の例	正規表現パターン	置換される値
カード番号			
携帯電話番号	010-1234-5678	(?<!w)(+82-?1[016789]-?d{3,4}-?d{4} 01[016789]-?d{3,4}-?d{4})(?!w)	***_***_ ****
メールアドレス	example@domain.com	([a-zA-Z0-9][a-zA-Z0-9._%+~]*@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,})	***@***.***
銀行口座番号	123456-01-654321	(?<!w)d{6}-d{2}-d{6}(?!w)	*****_**_ *****

個人情報のパターン	個人情報の例	正規表現パターン	置換される値
事業者登録番号	111-11-01111	(?<!w)d{3}-d{2}-d{5}(?!w)	***_** *****
法人登録番号	111111-1111111	(?<!w)d{6}-?d{7}(?!w)	*****_ *****

ログパースング

ログパーサを使用すると、不規則な形のログをクエリが可能な構造化された形に変更できます。WhaTapログモニタリングは、次の2つのタイプのパーサを提供します。

- **GROKパーサ**：任意の形で収集されるログを正規表現とGROKを活用してパースングします。
- **JSONパーサ**：JSON形式で収集されるログをパースングします。

ⓘ 共通の注意事項

- 同じカテゴリに複数のパーサが登録されている場合、最初にマッチングされるパーサのみが適用されます。
- WhaTapはWhaTapサービスの安定性に影響を与えるパーサを無効にする場合があります。。

GROKパーサ

ログが不規則な形で収集される場合、GROKパーサを使用してログをパースングできます。GROKパターンは、named regular expressionsを提供し、正規表現をより簡単かつ便利に使用できます。

GROKを開始する

GROKは2種類の文法を提供します。

1. `%{SYNTAX:SEMANTIC}`：GROKライブラリーが提供するパターンです。**named regular expressions**を活用してタグを抽出できます。活用例は[次のリンク](#)を参照してください。
 - **SYNTAX**：GROKが提供するnamed regular expressionsを指定します。
 - **SEMANTIC**：マッチングされる値に付ける名前を指定します。

ⓘ named regular expressions

GROKが提供するパターンです。複雑な正規表現に名前を付けて使用できるよう、GROKが提供す

❗ る機能です。

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>

WhaTapで提供するすべてのnamed regular expressionsを確認したい場合は、次の[リンク](#)を参考にしてください。

2. `(?<SEMANTIC>REGX)` : 正規表現の**named capturing group**文法です。正規表現を活用して、ユーザーの意図に合わせてタグを抽出できます。活用例は[次のリンク](#)を参照してください。

- **SEMANTIC** : マッチングされる値に付ける名前を指定します。
- **REGX** : マッチングに使用する正規表現を入力します。

❗ named capturing group

正規表現の文法です。

- capturing group: 複数のトークンを1つにまとめて1つのマッチング単位で使用する機能を意味します。
- named capturing group: capturing groupに名前を付けたものです。
- 文字列マッチングの例を見てみましょう。 dev@whatap.io
 - 例1 : `(\w+)@(\w+\.\w+)`
 - 例2 : Eメール全体のマッチング、およびusernameとdomainを追加でマッチングする場合 `(?<username>\w+)@(?<domain>\w+\.\w+)`

%{SYNTAX:SEMANTIC} 活用の例

次は %{SYNTAX:SEMANTIC} を活用する例です。

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
```

サンプルログを見て、各単語が意味する内容を推測できます。各部分をsemanticな単語に置き換えると、次のように表現できます。

semantic replace

```
[date][logLevel][caller] projectCode logCategory dexBuildStartTime {area=areaEnum, city=cityEnum} dexBuildElapsed
```

semanticな単語すべて、正規表現に書き換えることができます。GROKパーサを使用すると、事前に定義されたnamed regular expressionsを活用できます。ここで使われた `TIMESTAMP_ISO8601`、`LOGLEVEL`、`DATA` は、GROKが提供するnamed regular expressionsです。これらの値は、それぞれ次の正規表現に置き換えられてマッチングされます。

- 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)?|[Ee]rr?(?:or)?|ERR?(?:OR)?|[Cc]rit?(?:ical)?|CRIT?(?:ICAL)?|[Ff]atal|FATAL|[Ss]evere|SEVERE|EMERG(?:ENCY)?|[Ee]merg(?:ency)?)`
- name: `DATA`
 - regular expression: `.*`

GROK parsing pattern

```
\[%{TIMESTAMP_ISO8601:date}\sGMT\]\[%{LOGLEVEL:level}\s\]\[%{DATA:caller}\]
```

上記の内容でパースングすると、次のようにタグを抽出できます。このようにGROKの `%{SYNTAX:SEMANTIC}` パター

ンは、複雑で長い正規表現を簡単に適用できる機能を提供します。

Tag extraction

```
- date : 2023-08-08 02:02:30,101
- caller : i.w.y.l.c.LogSinkDexScheduleThread.realProcess(159)
- level : LEVEL
```

(?<SEMANTIC>REGX) 活用の例

named regular expressionsでマッチングされない部分は (?<SEMANTIC>REGX) パターンを使ってパースできます。上記の[サンプルログ](#)から `%{SYNTAX:SEMANTIC}` パターンだけでパースされない領域は次の通りです。

Unparsed area

```
8 VirtualLog 20230808 02:01:00.000 {area=4, city=5} 56ms
```

該当ログの各部分をsemanticな単語に置き換えると、次のように表現できます。

semantic replace

```
projectCode logCategory dexBuildStartTime {area=areaEnum, city=cityEnum} dexBuildElapsed
```

このように不規則な形の文字列は、次のような (?<SEMANTIC>REGX) パターンを使ってパースできます。

サンプルログパースキーワード別にマッチングされる正規表現

パースキーワード	(?<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)

パーシングキーワード	(?<SEMANTIC>REGX)
56 ms	(?<dexBuildElapsed>\d{2})ms

正規表現の基本文法

文法	意味	別名
?	0 or 1	-
+	1 or more	-
*	0 or more	-
a{5}	exactly 5	-
\w	word character	[a-zA-Z_0-9]
\s	white space	-
.	any character except newline	-
[abc]	any of	-
[^abc]	not a,b, or c	-
[a-z]	character between a and z	-
[1-3[7-9]]	union (combining two or more character classes)	-
[1-6&&[3-9]]	intersection (共通集合)	-
[0-9&&[^2468]]	subtraction (差集合)	-
a{2,}	2 or more	-
a{1,3}	between 1 and 3	-



文法	意味	別名
<code>a+?</code>	match as few as possible	-
<code>{2,3}?</code>	match as few as possible	-
<code>(abc)</code>	capturing group (複数の文字列をsingle unitで処理する)	-
<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	-

このようにパージングされたキーワードをスペース(`\s`)と、特殊文字escape(`\{`、`\,`、`\}`)で接続すると、次のようにパターンを適用できます。

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
```

上記の内容でパージングすると、次のようにタグを抽出できます。

Tag extraction

```
- projectCode : 8
- logCategory : VirtualLog
- dexBuildStartTime : 20230808 02:01:00.000
- areaEnum : 4
- cityEnum : 5
- dexBuildElapsed : 56
```

GROKを適用する

ログ設定 > ログプライマリパーサ設定

1. GROKパターンパーサを適用するには、**ログ設定**メニューの**ログプライマリパーサ設定**タブに移動します。
2. **+ 追加する**を選択した後**パーサ**入力欄で**GROK**パーサを選択します。
3. **パターン登録**ボタンを選択すると、右側にパターン登録とシミュレーション画面が表示されます。
4. **パターン**と**ログ**入力後、**シミュレーション**ボタンをクリックして、適用するパターンでパーシングに成功することを確認します。

パターン 例:

```
[%{TIMESTAMP_ISO8601:date}\sGMT][%{LOGLEVEL:level}\s][%{DATA:caller}]\s(?:<projectCode>\d)\s(?:<logCategory>\d{2}:\d{2}:\d{2}\.\d{3})\s\s{area=(?:<areaEnum>\d),\scity=(?:<cityEnum>\d)\}\s(?:<dexBuildElapsed>\d{2})ms
```

ログ例:

```
[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. シミュレーションの成功時**シミュレーション結果**と**性能測定の結果**を照会できます。
6. シミュレーション後、**パターン適用**ボタンをクリックすると、選択したパーサーに入力したパターンが適用されます。
7. パターン適用後、**カテゴリ**と**ログ検出条件**、**パターン**を入力します。

×

パーサ 追加

パーサ*

WhaTap Monitoringには、サービスの信頼性に影響を与える可能性があるパーサーを無効にできる権限があります。

GROK

パーサーパターン*

入力したパターンでログが正常に解析されるかどうかをシミュレートし、パターンを登録します。

成功しました。

```

\[%{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})\s\{area=(?<areaEnum>\d),\scity=(?
<cityEnum>\d)\}\s(?<dexBuildElapsed>\d{2})ms

```

パーサーパターンの修正

カテゴリ*

同じカテゴリに複数のパーサーが登録されている場合、最初に一致するパーサーのみが適用されます。

AppLog

ログ検出条件

フィルタを入力しなかった場合、または検索キー／値を1つだけ入力した場合、そのパターンはすべてのログに適用される。

oname

demo-8100

追加

• カテゴリ

ログカテゴリを選択します。 **カテゴリ**は必須入力項目です。

• ログ検出条件

- 条件を満たすログのみパーサが適用されます。
- 検索キーと検索値を選択するか、直接入力します。
- ログ検出条件は、すべてのパーサが実行される前に適用されます。つまり、パーサの結果として追加されるタグを使用できません。

8. 追加ボタンを選択してパーサを登録します。

- ⓘ ログパーサー一覧から該当パーサの優先順位を変更するか、**活性化**および**修正**、**削除**できます。
- パーサーシミュレーション後、パターンを登録できます。

384

❗ GROKパーサの注意事項

- GROKパーサは `%{SYNTAX:SEMANTIC}` 、 `%{SYNTAX:SEMANTIC}` 二つのパターンをサポートします。
- `%{SYNTAX:SEMANTIC}` パターンを使用する時、 `SEMANTIC` を必ず入力します。
- `%{SYNTAX:SEMANTIC}` パターンを使用する時、 `SEMANTIC` は一つのパーサの中でユニークである必要があります。
- `(?<SEMANTIC>REGX)` パターンを使用する時、 `SEMANTIC` は文字(a-z、A-Z)、数字(0-9)、および指定された特殊文字(`.` 、 `_` 、 `-`)のみが使えます。
- `SEMANTIC` は文字(a-z、A-Z)で始めなければなりません。
- `SEMANTIC` は文字(a-z、A-Z)または数字(0-9)で終わる必要があります。

GROK追加パターンの選択

GROKパーサのシミュレーション時に、基本パターン(`WORD` 、 `NUMBER` 、 `IP` など)以外に特定のログ形式に合った追加パターンを選択できます。Java、Nginx、MySQLなど32個のパターンタイプが11個のカテゴリにグループ化されて提供されます。

追加パターンカテゴリのドロップダウン

1. パーサのシミュレーション画面で**追加パターン (選択)**領域のドロップダウンをクリックしてください。
2. カテゴリ別にグループ化されたパターンタイプ一覧から、必要なパターンタイプを選択してください。 検索入力欄にパターンタイプ名または説明を入力して検索することもできます。
3. 選択したパターンタイプは、シミュレーション時に基本パターンとともにロードされます。

❗ 基本パターン(Core)は常に自動的に含まれます。追加パターンの選択は任意です。

カテゴリ別パターンタイプ一覧

カテゴリ	パターンタイプ
Web Servers	haproxy, httpd, iis, nginx, squid

カテゴリ	パターンタイプ
Application Servers & Frameworks	java, log4j, rails, ruby, springboot
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

活用例

Tomcat Catalina 8のログをパースングするには、**java**パターンタイプを追加で選択した後、`%{CATALINA8_LOG}` パターンを使用してください。

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
```

パターン詳細表示

選択した追加パターンタイプにどのパターンが含まれているかを詳細画面で確認できます。パターン定義(正規表現)とサンプルログを照会し、シミュレーションにすぐに適用できます。

パターン詳細表示

追加パターンタイプを選択した後、選択されたパターンタイプのタグをクリックすると、右側に詳細画面が開きます。詳細画面はシミュレーションフォームを隠さずに並んで表示されるため、パターン定義を確認しながらシミュレーションを進めることができます。

各パターン項目で次の機能を使用できます。

ボタン	機能	説明
サンプル	サンプルログ入力	該当パターンにマッチングする例ログをシミュレーションパーシングログ入力欄に自動的に入力します。
選択	パターン挿入	<code>%{パターン名}</code> 形式でパターン入力欄に挿入します。該当パターンタイプも自動的に選択されます。
コピー	クリップボードコピー	<code>%{パターン名}</code> 形式をクリップボードにコピーします。

- ① パターン項目をクリックすると、パターン定義(正規表現)が展開されます。複数のパターンを同時に展開して定義を比較できます。
- パターン定義に含まれる `%{PATTERN:field}` 参照は、パターン名、フィールド名、リテラル文字がそれぞれ異なる色で区別されて表示されます。
- サンプルボタンで入力されたログでそのままシミュレーションを実行してパーシング結果を確認できます。

シミュレーションサンプルログの保存

シミュレーションに使用したログは、パーサ設定とともに自動的に保存されます。パーサを修正する際に、以前に使用したサンプルログがシミュレーション画面に自動的に復元され、毎回ログを再入力する必要がありません。

JSONパーサ

ログがJSONフォーマットで収集される場合、JSONパーサを使用して簡単かつ便利にパーシングできます。

JSONを適用する

ログ設定 > ログプライマリパーサ設定

1. JSONパターンパーサを適用するには**ログ設定**メニューの**ログプライマリパーサ設定**タブに移動します。
2. **+ 追加する**を選択した後**パーサ**入力欄で**JSON**パーサを選択します。
3. **パターン登録**ボタンを選択すると、右側にパターン登録とシミュレーション画面が表示されます。
4. **パターン**と**ログ**入力後、**シミュレーション**ボタンをクリックして、適用するパターンでパーシングに成功することを確認します。

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"} --
```

例のログで**Prefix**と**Postfix**を `--` に指定し、**Ignore**に `url` を指定時 `host`、`method`、`status` 3つ**タグ**のみ生成します。

- **Prefix** ログでJSONフォーマットが開始する場所を指定します。ログ全体がJSONフォーマットの場合は、空の値に設定します。
 - **Postfix** ログでJSONフォーマットが終わる位置を指定します。ログ全体がJSONフォーマットの場合は、空の値に設定します。
 - **Ignore** JSONフォーマットの中でタグを生成しないキーを指定します。
5. シミュレーションの成功時**シミュレーション結果**と**性能測定の結果**を照会できます。

×

パーサ シミュレーション

ガイド表示

パターン

agenttime, category, content, logContent, message, pcode, time, timestamp は予約語です。
予約語はログ検出条件、パターンで登録しても動作しません。

Prefix

-

Postfix

-

Ignore

url

ログ *

ログシミュレーションは必須ですする必要があり、
シミュレーションの結果に応じてパターン登録が必要になる場合があります。

2023-08-08 02:43:28.615 -
{"host":"10.21.3.24","method":"POST","status":"200","url":"http://devote.whatap.io/yard/api/flush"} -
-

シミュレーション

シミュレーション結果 成功しました。

キー	値	結果
method	POST	Ok
host	10.21.3.24	Ok
status	200	Ok

性能測定の結果 成功しました。

シミュレーション数	最小時間(ns)	最大時間(ns)	平均時
1	4,491	4,491	4,491
10	3,322	3,930	3,525
100	2,662	177,949	8,084
1,000	2,654	476,818	4,305
10,000	2,648	311,233	3,823
100,000	2,650	60,522,941	3,713
1,000,000	2,647	44,113,067	3,014

パターン適用

6. シミュレーション後、パターン適用ボタンをクリックすると、選択したパーサーに入力したパターンが適用されます。

7. パターン適用後、カテゴリとログ検出条件、パターンを入力します。

389

×

パーサ 追加

パーサ*

WhaTap Monitoringには、サービスの信頼性に影響を与える可能性があるパーサーを無効にできる権限があります。

JSON

パーサーパターン*

ログが正常に解析されるかどうかをシミュレートします。シミュレーションは必須でなければならず、シミュレーション結果によってはパターン登録がさらに必要になる場合があります。

✔ 成功しました。

Prefix --

Postfix --

Ignore url

シミュレーション検証とパターン修正

カテゴリ*

同じカテゴリに複数のパーサーが登録されている場合、最初に一致するパーサーのみが適用されます。

AppLog

ログ検出条件

フィルタを入力しなかった場合、または検索キー／値を1つだけ入力した場合、そのパターンはすべてのログに適用される。

oname

demo-8100

追加

• カテゴリ

ログカテゴリを選択します。 **カテゴリ**は必須入力項目です。

• ログ検出条件

- 条件を満たすログのみパーサが適用されます。
- 検索キー**と**検索値**を選択するか、直接入力します。
- ログ検出条件**は、すべてのパーサが実行される前に適用されます。つまり、パーサの結果として追加される**タグ**を使用できません。

8. 追加ボタンを選択してパーサを登録します。

- ⓘ ログパーサー一覧から該当パーサの**優先順位**を変更するか、**活性化**および**修正**、**削除**できます。
- パーサを登録する前に**シミュレーション**を通じて登録するパターンが正常であるか確認できます。GROK

390



パーサ登録シミュレーションプロセスと同じです。 [次の文書](#)を参考にしてください。

JSONフォーマット活用例

Sample log

```
{"host":"10.21.3.24","method":"POST","status":"200","url":"http://dev.whatap.io/yard/api/flush"}
```

上記のようなサンプルログが収集された場合パーサ 追加画面でJSONパーサを選択します。複雑なパーシングロジックを作成する必要なく、ログ分析に必要なタグを次のように抽出できます。

Tag extraction

```
- host : 10.21.3.24
- method : POST
- status : 200
- url : http://dev.whatap.io/yard/api/flush
```

JSONフォーマットを一部構成する時の活用例

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"}
--
```

もし、例のようにログの一部のみがJSONフォーマットで構成されている場合はPrefixとPostfixを指定してください。WhaTapログモニタリングはPrefixとPostfix間の領域をJSONフォーマットで認識してからパーシングします。

Tag extraction

```
- host : 10.21.3.24
- method : POST
- status : 200
- url : http://dev.whatap.io/yard/api/flush
```

警告アラート

監視対象のシステムまたはアプリケーションで問題が発生した際に、あらかじめ設定した条件に基づいてユーザーへ通知を送信する機能です。通知は、メール、SMS、メッセージャー、アプリのプッシュ通知など、さまざまなチャネルを通じて配信されます。

たとえば、CPU 使用率、メモリ使用量、ディスク I/O、ネットワークトラフィック、トランザクション数、エラー発生、レスポンスタイムの遅延などの主要なパフォーマンス指標にアラート基準を設定しておく、その基準を超過した場合や異常な兆候が検出された際に通知が送信されます。

WhaTap の AI ベースのアラート機能を使用すると、複雑な条件を手動で設定する必要はありません。AI がアプリケーションの正常な動作パターンを学習し、異常パターンを自動的に検知して通知します。また、タグカウントアラート機能を活用することで、特定のタグを基準に集計された値に条件を設定でき、複雑な状況でも高精度な検知が可能です。

主な機能

- ・ **イベント設定:** 閾値ベースのアラート条件および通知チャネルの設定
- ・ **通知設定:** ユーザーごとの通知チャネルおよび時間帯設定
- ・ **イベント履歴:** アラート発生履歴の確認

権限

イベントを追加または設定するには、**アラート設定権限** が必要です。

- ・ **閲覧:** メトリクスイベント設定を確認するには、**閲覧・分析権限** が必要です。
- ・ **追加／編集／削除:** 設定を変更するには、**アラート設定権限** が必要です。
- ・ **通知受信:** 通知を受信するには、**通知受信権限** が必要です。

📌 参考資料

詳細については、[メンバー権限](#) を参照してください。

イベント設定

ホーム画面 > プロジェクト選択 > 🔔 アラート通知 > イベント設定

イベント設定では、監視対象のシステムやアプリケーションで発生する可能性のある異常兆候を基準に、イベントを設定・管理できます。設定した条件が満たされると、メール、SMS、メッセージャー、アプリのプッシュ通知など、希望するチャンネルで通知を受信できます。監視の目的に応じて、イベントを柔軟に構成できます。

📄 メトリクス

メトリクスイベントは、リアルタイムデータを基にシステムおよびアプリケーションのパフォーマンス指標を監視できます。

📄 複合メトリクス

複合メトリクス機能では、複雑なルールを活用してイベントを作成し、警告アラートを送信できます。

📄 異常値

異常値機能では、想定外のパターンを検出して警告アラートを送信するよう設定できます。

📄 リアルタイムログ

リアルタイムで収集されたログから設定したイベント条件が検出された場合、警告アラートを送信します。

複合ログ

ログの内容を基準に警告通知を発生させる複合ログイベントを設定します。

ホーム画面 > プロジェクト選択 >  **アラート通知** > **イベント設定**

イベント設定では、監視対象のシステムやアプリケーションで発生する可能性のある異常兆候を基準に、イベントを設定・管理する機能を提供します。設定した条件を満たすと、メール、SMS、メッセージャー、アプリプッシュなど、希望するチャンネルで通知を受信できます。監視目的に応じてイベント構成を設計できます。

メトリクス

メトリクスイベントは、リアルタイムデータを基にシステムおよびアプリケーションのパフォーマンス指標を監視できます。

複合メトリクス

複合メトリクス機能では、複雑なルールを活用してイベントを作成し、警告アラートを送信できます。

異常値

異常値機能では、想定外のパターンを検出して警告アラートを送信するよう設定できます。

リアルタイムログ

リアルタイムで収集されたログから設定したイベント条件が検出された場合、警告アラートを送信します。

📄 複合ログ

ログの内容を基準に警告通知を発生させる複合ログイベントを設定します。

📌 参考

- イベント設定時は、不要なイベント発生を防ぐために適切なしきい値を設定してください。
- 通知設定では、問題解消時に通知される回復通知もあわせて検討すると、運用対応に役立ちます。

共通機能

すべてのイベント設定は **JSON** または **Excel 形式** でダウンロードできます。同じ種類のプロダクト設定を共有することで、繰り返し発生するイベント設定作業を削減できます。

- **JSON 一括編集**: すべてのイベント条件を JSON 形式で一括編集できます。
- **JSON 一括ダウンロード**: 現在登録されているイベント条件を JSON ファイルとして保存できます。
- **Excel 一括ダウンロード**: イベント一覧を Excel ファイルとしてダウンロードし、外部レポートやバックアップに活用できます。

JSON 一括編集

1. 画面上部の [JSON 一括編集] ボタンをクリックします。
2. **JSON 一括ダウンロード** 機能でダウンロードした JSON ファイルを読み込むか、直接編集します。
3. [上書き] ボタンをクリックして保存します。
4. 確認メッセージが表示されたら、[上書き] ボタンをクリックします。

⚠️ 上書き後は、以前のイベント設定に戻すことはできません。

- `productType` または `platform` プロパティの値を任意に変更すると、他のプロジェクトでイベントが正常に動作しなくなる可能性があります。



- JSON 編集画面には検証機能が組み込まれています。形式や値が正しくない場合、編集画面下部にエラーメッセージが表示され、[上書き] ボタンは無効になります。

```

    },
    "BASIC": [
      {
        "criticalEnabled": "false",
        "warningEnabled": true,
        "silentPeriod": 300000,

```

① 「criticalEnabled」には、「string」タイプではなく、「boolean」タイプの値が必要です。

読み込み

キャンセル

適用

JSON データ構造



```

{
  "productType": "{productType}",
  "platform": "{platform}",
  "events": {
    "HITMAP": {
      ...
    },
    "LOG_REALTIME": [
      {...}
    ],
    "METRICS": [
      {...}
    ],
    "COMPOSITE_LOG": [
      {...}
    ],
    "BASIC": [
      {...}
    ],
    "COMPOSITE_METRICS": [
      {...}
    ],
    "ANOMALY": [
      {...}
    ]
  }
}

```




```

  ]
}
}

```

JSON フィールド	説明
productType	プロダクトタイプ（例: APM、Infra など）
platform	プラットフォーム情報（例: java、node など）
events	イベントデータ
└ HITMAP	ヒットマップパターン
└ LOG_REALTIME	リアルタイムログイベント
└ METRICS	メトリクス
└ COMPOSITE_LOG	複合ログイベント
└ BASIC	基本イベント
└ COMPOSITE_METRICS	複合メトリクス
└ ANOMALY	異常値

JSON 一括ダウンロード

画面上部の [JSON 一括ダウンロード] ボタンをクリックします。

- JSON ファイル名は `integrated-event-YYYYMMDD.json` 形式です。
- JSON データの構造や `events` 属性配下のサブ属性は、プロダクト（`platform`）またはプロダクトタイプ（`productType`）によって異なる場合があります。

⚠ JSON データ内の `productType` または `platform` プロパティの値を任意に変更すると、他のプロジェクトでイベントが正常に動作しなくなる可能性があります。

Excel 一括ダウンロード

画面上部の [Excel 一括ダウンロード] ボタンをクリックします。

メトリクス

ホーム > プロジェクト選択 >  アラート通知 > イベント設定 > メトリクス タブ

プロジェクトに設定されているすべてのメトリクスベースのイベント一覧を確認できます。設定済みのメトリクスイベントを確認、修正、削除したり、新しいイベントを追加したりできます。

基本オプション

- **イベントレベル:** イベントの深刻度を示し、正常（Info）、警告（Warning）、危険（Critical）の3段階に分類されます。
- **エージェント重複アラート抑止:** 同じ名前のエージェントが同時に2つ以上動作している場合、選択した時間（30秒、1分、3分、5分、10分）の間はイベントが発生しません。
- **検索:** イベント名を基準に検索します。
- **JSON ダウンロード／アップロード:** JSON エディタでイベントルールを直接編集するか、ファイルとしてダウンロードして編集後に再アップロードできます。
- **+ イベント追加:** 新しいメトリクスイベントを追加します。
- **カラムサイズ調整:** 各カラムの境界をドラッグして幅を調整できます。

表 | メトリクスイベント一覧構成

項目	説明
No.	イベントの番号
	イベントの有効化状態 – 有効: 条件を満たすとイベントが発生 – 無効: イベントは動作せず、条件も評価されません
	イベント設定の編集および削除
イベント名	ユーザーが指定したイベント名 – 最大255文字

項目	説明
ルール	イベント発生条件
対象	イベントが適用される監視対象
発生回数	イベント発生の基準となる期間と回数 – 選択した時間内に設定回数分条件を満たすとイベントが発生
一時停止	アラート発生後、同一イベントの発生を一定時間停止 – 選択可能時間: 使用しない、5分、10分、15分、20分、30分、1時間、2時間、3時間、6時間、12時間、1日 – 回復通知が有効な場合、RECOVERED（正常）通知後、選択した時間は同一イベントが発生しても通知されません
回復通知	Critical および Warning レベルのイベントが解消された際に、RECOVERED（正常）通知を受信するかどう – 有効: 進行中から正常状態に遷移すると RECOVERED 通知を送信し、イベント履歴で進行中／解消状態を区別して確認可能 – 無効: しきい値を超えるたびに単発イベントが発生
イベント受信	該当イベント通知を受信するユーザーまたはグループ（受信タグ）

イベント設定

メトリクスイベントは、監視対象で発生する事象を条件に基づいて定義し、効率的に通知を受信できるよう設定するための機能です。

イベント追加

メトリクスイベントを追加するには、**イベント設定** > **メトリクス** タブに移動し、右上の [+ イベント追加] ボタンをクリックします。

メトリクスイベント追加手順
Step 1. テンプレート選択: イベントの基本構成を選択します。
Step 2. イベント条件定義: どの条件でイベントを発生させるかを設定します。
Step 3. イベント対象選択: イベントが適用される監視対象を選択します。
Step 4. 基本情報および通知設定: イベント名、メッセージ、受信対象を設定します。

テンプレート選択

1. イベントの構成方法に応じて、画面左側の3つのテンプレートのいずれかを選択します。
 - **新規作成:** カテゴリやメトリクスを一から自由に構成できます。複雑な条件や詳細な監視ルールが必要な上級ユーザーに適しています。
 - **クイック設定:** あらかじめ定義されたカテゴリ、メトリクス、基本しきい値が提供され、しきい値のみを調整して素早く適用できるため、初心者に適しています。
 - **高度な設定:** あらかじめ定義されたカテゴリやメトリクスを基に、条件を追加したり自由に変更したりできます。

❗ メトリクスイベントでは、さまざまな設定方法をサポートしています。利用する監視プラットフォームによって、提供されるテンプレートは異なる場合があります。

イベント条件定義

イベント条件定義のステップでは、イベント発生の判断に使用するメトリクス設定、発生回数、一時停止、回復通知、イベント動作時間を設定します。

指標設定

1. 検索欄にカテゴリを入力するか、カテゴリー一覧からカテゴリ（指標）を選択します。
指標設定を進めるには、カテゴリの選択が必要です。

- カテゴリ一覧では、カテゴリ名、データ収集間隔、キーを確認できます。直近3時間にプロジェクトで収集されたメトリクスデータが表示されます。

2. 一覧にない指標を使用する場合は、**直接入力**を選択してカテゴリを手動で入力します。

❗ カテゴリや指標が不明な場合は、メトリクス照会を活用してください。

初めて指標を設定する場合、メトリクス照会を利用すると効率的に指標を見つけられます。

- メトリクス照会で指標を検索すると、関連するカテゴリや詳細情報を確認できます。
- 検索結果から必要なメトリクスの[選択]ボタンをクリックすると、選択した値がイベント設定画面に自動的に反映されます。

メトリクス検索

All 13件 / 全 149件

カテゴリ	名前	単位	タイプ	詳細	マージ	
ApplicationHostResource	CpuSys	pct	field	System領域のCPU使用率	OM: avg TM: avg	選択
ApplicationHostResource	CpuIrq	pct	field	Interrupt Request (IRQ)を使用したCPU使用率	OM: avg TM: avg	選択
ApplicationHostResource	CpuSteal	pct	field	実際にCPUを待機する時間を%で表示した値	OM: avg TM: avg	選択
ApplicationHostResource	CpuCores	count	field	CPUコア数	OM: avg TM: avg	選択
ApplicationHostResource	CpuUser	pct	field	User領域のCPU使用率	OM: avg TM: avg	選択
ApplicationHostResource	CpuProc	pct	field	Proc領域のCPU使用率	OM: avg TM: avg	選択
ApplicationHostResource	Cpu	pct	field	CPU使用率	OM: avg TM: avg	選択
ApplicationHostResource	CpuWait	pct	field	ディスク入出力を待つ間(wait)に使用したCPU使用率	OM: avg TM: avg	選択
ProcessPrivate	Private	na		Number of Bytes Private		

< 1 > 50件 / ページ

3. イベントレベル（Critical、Warning、Info）を選択します。1つの指標に対して複数のレベルを設定できます。

❗ イベントレベルの動作

複数の条件を満たしても、最も優先度の高いレベル（Critical → Warning → Info）のみがイベントとして発生します。詳細は [レベル動作ガイド](#) を参照してください。

- イベントレベルは、危険（Critical）、警告（Warning）、正常（Info）に分類されます。
- 同一指標に対して段階的なしきい値でレベルを設定することを推奨します。

4. 選択したイベントレベルのフィールド、演算子、値を入力します。

a. **+** 追加 をクリックすると条件を追加できます。

- **&&** : すべての条件を同時に満たした場合 にイベントが発生
- **||** : いずれか 1 つ以上の条件を満たした場合 にイベントが発生

b. **-** をクリックすると、その条件が削除されます。

❗ 条件式の書き方や対応する演算子については、[条件設定ガイド](#) を参照してください。

発生回数

条件が何回満たされた場合にイベントを発生させるかを設定します。継続的な問題のみを検知したい場合に有効です。

1. 発生頻度を選択します。

- **連続**: 設定した回数分、イベントが連続して発生した場合にイベントを発生
- **直近**: 選択した時間内に、設定した回数分イベントが発生した場合にイベントを発生
 - 選択可能時間: 5秒, 10秒, 15秒, 30秒, 1分, 2分, 3分, 4分, 5分, 10分, 30分, 1時間

2. 発生回数を入力します。

- **Interval** の値は、選択したカテゴリのデータ収集間隔を示します。（Interval: 10s）

一時停止

過剰なイベント発生を防ぐため、イベント動作を一時停止する時間を選択してください。イベント発生後、設定した時間が経過するまで同一イベントは発生しません。

- 選択可能時間: 使用しない, 5分, 10分, 15分, 20分, 30分, 1時間, 2時間, 3時間, 6時間, 12時間, 1日

❗ 解消通知が有効な場合

正常状態の通知（**RECOVERED**）を受信した時点から、設定した時間が経過するまで、同一イベントが再発しても通知は送信されません。

解消通知

解消通知機能を有効にすると、イベント発生後の状態変化（進行中 → 解消）を追跡できます。**イベント履歴** メニューでは、進行中のイベントとして表示されます。

- 解消通知は **Critical** および **Warning** レベルにのみ適用されます。Info レベルは常に単発イベントとして動作します。
- **Critical** または **Warning** レベルのイベントが解消されると、正常（RECOVERED）状態の通知が送信されます。
- トグルボタンをクリックして **有効化** または **無効化** してください。
 -  **有効**: イベントは状態ベースで動作します。しきい値を超えると進行中となり、値がしきい値未満に回復すると解消状態に遷移します。
 -  **無効**: しきい値を超えるたびに単発イベントが発生します。

イベント動作時間

イベントが特定の時間帯（勤務時間／非勤務時間／点検時間など）にのみ動作するよう、タグを設定してください。タグを設定しない場合、イベントが有効な状態では 24 時間常時動作します。

1. **+ 追加** をクリックします。
2. イベント動作時間で **+ 新しいタグを作成** をクリックします。
3. **イベント動作時間タグ作成** で **タグ名**、**曜日**、**時間**、**色** を選択し、**[適用]** ボタンをクリックします。
4. 作成したタグはタグ一覧で確認でき、チェックボックスを選択すると適用されます。
5. タグを修正または削除するには、 をクリックし、**イベント動作時間タグ修正** で修正または削除してください。

 タグを削除すると、そのタグが適用されているすべてのユーザーからタグが削除されます。ただし、イベントルールで使用中のタグは削除できません。

シミュレーション

シミュレーション機能では、設定したイベント条件を過去データに基づいて事前にテストできます。想定されるイベント件数が多いすぎる、または少なすぎる場合は、条件を調整して最適なしきい値を見つけることができます。

- **ルール検証:** 新しく作成したイベントルールを過去データに適用した場合の結果を確認できます。
- **しきい値チューニング:** イベントが頻発する、またはまったく発生しない場合に、適切なレベルへしきい値を調整します。

シミュレーション開始

1. イベント条件の設定が完了したら、[シミュレーション] ボタンをクリックします。
2. 想定イベント件数と可視化された結果を確認します。

シミュレーション結果の理解

1. 時系列データ

指標の時間帯別変化と各レベルのしきい線がグラフで表示され、イベント発生タイミングを直感的に確認できます。

2. レベル別発生状況

画面右側に Critical、Warning、Info レベルごとのイベント発生回数が表示され、全体の分布や頻度を把握できます。

3. 対象別詳細情報

グラフ上の特定ポイントにマウスを合わせると、その時点の正確な指標値と監視対象が表示されます。複数の対象がある場合は色で区別され、対象ごとにパターンが異なる場合はフィルタリング機能で個別分析が可能です。

4. ピーク区間の特定

指標が急激に上昇する時間帯を確認し、問題が頻発する時間帯やパターンを把握できます。

🟢 シミュレーション活用のヒント

• 複数時間帯でのシミュレーション

平日／週末、ピーク／非ピーク時間帯をそれぞれシミュレーションし、時間帯ごとの最適なしきい値を設定してください。



- 実際の障害期間を含める

過去に障害が発生した期間を含めてシミュレーションを実行し、現在のルールが障害を正しく検知できるか確認しましょう。

- 対象フィルタリングの追加

監視対象が多い場合、結果が複雑になります。フィルタリング機能を使って特定の対象のみを選択し、明確な結果を確認してください。

- しきい値を段階的に調整

一度に大きく変更するのではなく、10 パーセント単位で段階的に調整しながら最適値を探すことをおすすめします。

イベント対象選択

イベント対象を指定しない場合、プロジェクト内のすべての対象に対してイベントが発生し、多数のイベントが生成される可能性があります。そのため、イベントを適用する対象を明確に設定することを推奨します。

❗ イベント対象を変更すると、イベント件数が変わる場合があります。シミュレーション ボタンをクリックして再実行し、想定イベント件数を確認してください。

対象選択

イベントを適用する対象は、**選択入力** または **直接入力** の方法で指定できます。

- **選択入力** 直接入力

1. タグ、演算子、値を選択または入力してください。
 - a. タグが見つけない場合は、**メトリクス照会** を使ってタグを検索できます。
2. **+ 追加** をクリックして、条件（**&&**、**||**）でイベント対象を追加してください。

監視対象のタグ、演算子、値を直接入力してください。

- 直接入力を選択すると、入力欄の下に使用可能なタグ一覧（**タグ選択入力**）が表示されます。

```
ex. endsWith(okindName, 'example_name') && container == 'prod.billing'  
ex. ${4xxErrorType} == '401'
```

❗ 入力方法を変更すると、作成中の内容は初期化されます。

基本情報および通知設定

イベント名、メッセージ、受信対象を設定します。設定した内容はイベント履歴や実際の通知メッセージにそのまま表示されるため、識別しやすく分かりやすい名称にすることが重要です。

イベント有効化

設定した条件を満たした際にイベントを発生させるかどうかを設定します。

- ☒ **有効**: 設定した条件を満たすとイベントが発生します。
- ☐ **無効**: 条件を判定しません。

イベント名

イベントのタイトルとして使用するイベント名を入力してください。設定したイベント名は **イベント一覧** および **イベント履歴** メニューに表示され、この名前で検索することができます。

メッセージ

イベントメッセージを入力してください。入力したメッセージは、イベント履歴および通知に使用されます。🕒 をクリックすると、過去に作成したメッセージ履歴を確認できます。メッセージには変数を使用して、イベント発生時点の実際の値を含めることができます。

- **変数の使用ルール**
 - 変数形式: `${指標名}`
 - 設定した **カテゴリ**と同一**カテゴリ内**の指標のみを変数として使用できます。
 - 複数の変数を組み合わせて、より詳細なメッセージを作成できます。
 - 使用可能な変数一覧は **メトリクス照会** メニューで確認できます。

- 変数の使用例

	イベント設定	イベント発生時
タイトル	Disk 使用量増加	Disk 使用量増加
メッセージ	Disk = \${disk}%	Disk = 89.9%

イベント受信

イベント通知を受信するメンバーを指定します。

- 全体受信** タグ選択受信

プロジェクトで通知設定が有効になっているすべての対象（メンバー、チャンネル）に通知が送信されます。

指定したタグを持つプロジェクトメンバーに通知が送信されます。

1. **タグ選択受信** を選択します。
2. **受信タグ** 項目で [+ タグ追加] または [+] ボタンをクリックします。
3. **イベント受信タグ** 画面の下にある + **新しいタグを作成** をクリックします。
4. **タグ作成** 画面でタグ名を入力し、色を選択した後、[タグ作成] ボタンをクリックしてタグを作成します。
5. 作成したタグは、**タグ一覧**の  をクリックして修正または削除できます。
6. **イベント受信タグ** 画面の **タグ一覧** から希望するタグを選択すると適用されます。

ⓘ 受信タグ

受信タグは、通知を受け取るユーザーやチャンネルを **グループ単位で管理するための機能** です。イベントに受信タグを設定すると、そのタグが付与されたユーザーにのみ通知が送信されます。受信タグを設定しない場合、プロジェクト内のすべてのユーザーに通知が送信されます。

イベント修正／削除

1. **警告アラート** > **イベント設定** の **メトリクス** タブに移動します。

2. イベント一覧で、修正または削除したいイベントの  をクリックします。
3. イベントルール修正 画面でオプションを修正し、[保存] ボタンをクリックします。
 - a. 選択したイベントを削除する場合は、イベントルール修正 画面右上の [削除] ボタンをクリックします。

イベント共有

メトリクスイベント設定を JSON ファイルとして保存し、他のユーザーと設定を共有したり、他のユーザーの設定を取り込んだりすることができます。

- JSON ファイル名: `event-rules-YYYYMMDD.json`

エクスポート

1. 画面右上の [JSON ] ボタンをクリックします。
2. JSON 編集画面が表示されたら、[ エクスポート] ボタンをクリックします。
 - イベントを検索した状態でエクスポート機能を使用すると、検索結果のみが JSON ファイルとしてダウンロードされます。
3. JSON ファイルがダウンロードされたら、共有したい他のユーザーに渡してください。

インポート

1. 画面右上の [] ボタンをクリックします。
2. エクスポート 機能でダウンロードした JSON ファイルを選択します。
3. JSON 編集画面が表示されたら、[一覧に追加] または [上書き] を選択します。

 同じ種類の製品間での利用を推奨します。異なる製品のプロジェクトからイベント設定を取り込むことは可能ですが、正常に動作しない場合があります。

JSON 形式で編集する

1. 画面右上の [JSON ] ボタンをクリックします。

2. 編集画面が表示されたら、JSON 形式に従って内容を修正してください。
3. 修正が完了したら、画面下部の [保存] ボタンをクリックします。

❗ 修正した内容が JSON 形式に合っていない場合、画面下部にエラーメッセージが表示され、保存できません。表示されるエラーメッセージは、形式エラーの内容によって異なる場合があります。

JSON error

JSON データ構造

```
{
  "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"
  ]
}
}

```

表 | JSON データ構造

JSON フィールド	型	説明	備考
version	Integer	イベントルールのバージョン	変更制限
eventId	String	イベントルールの一意識別子	
enabled	boolean	イベントの有効／無効	
stateful	boolean	状態ベースイベントかどうか	
title	String	ユーザーが設定したイベントルール名	
message	String	ユーザーが設定したイベントルールメッセージ	
category	String	データカテゴリ	
alertLabel	List<String>	イベント状態管理のため、カテゴリごとに指定される基本識別子（プライマリキー）値	変更制限
repeatCount	Integer	繰り返し回数	
repeatDuration	Long	繰り返し期間	
silent	Integer	一時停止時間	
receiver	List<String>	受信タグのキー値一覧	

JSON フィールド	型	説明	備考
timeTag	List<String>	動作時間タグのキー値一覧	
selectString	String	対象選択	
conditions	List<Object>	レベル別の発生条件一覧	
└ level	byte	レベル値	Critical: 30、Warning: 20、Info: 10
└ enabled	boolean	レベルの有効／無効	
└ rule	String	レベル発生条件	
createTime	Long	イベントルールの初回作成時刻	
lastModifiedTime	Long	イベントルールの最終修正時刻	
lastModifiedUser	String	イベントルールを最後に修正したアカウント	
basic	boolean	クイック設定で作成されたイベントかどうか	変更制限
metald	String	クイック設定テンプレートの一意識別子	変更制限: この値はクイック設定で作成されたイベントにのみ存在します (basic=true)。
selectCondition	Object	クイック設定の対象選択値	この値はクイック設定で作成されたイベントにのみ存在します (basic=true)。

❗ 「変更制限」と表示されたフィールドはイベント動作に影響を与える可能性があるため、値を変更しないことを推奨します。

付録

発生条件・対象選択ガイド

メトリクスアラートにおけるイベントの発生条件とイベント対象の選択は、同一の構文を使用します。ただし、イベント発生条件ではフィールド（Field）のキーを変数として使用し、イベント対象の選択ではタグ（Tag）のキーを変数として使用します。

レベル動作ガイド

イベントレベルは、危険（Critical）、警告（Warning）、正常（Info）の3段階に分類されます。

1. 優先順位に基づくイベント発生

複数のレベル条件を同時に満たした場合、最も優先度の高いイベントのみが発生します。

例

```
設定:  
- Warning: CPU > 70%  
- Critical: CPU > 90%  
  
現在の状態: CPU 95%  
→ 結果: Critical イベントのみ発生（Warning は抑制される）
```

2. レベル上昇

進行中状態を持つイベントにおいて、低いレベルのイベントが進行中の状態で高いレベルの条件を満たすと、両方のレベルが進行中状態になります。

例

```
設定:  
- Warning: CPU > 70%  
- Critical: CPU > 90%  
  
シナリオ:
```

- 1) CPU 80% → Warning 発生 (進行中)
- 2) CPU 95% → Warning (進行中)、Critical (進行中)

3. レベル下降

進行中状態を持つイベントで、高いレベルから低いレベルへ遷移した場合、高いレベルは解消され、低いレベルが維持されます。

例

設定:

- Warning: CPU > 70%
- Critical: CPU > 90%

シナリオ:

- 1) CPU 97% → Critical 発生
- 2) CPU 85% → Critical 解消、Warning 維持
- 3) CPU 65% → Warning 解消

4. 動作フロー

- Warning: CPU > 70%
- Critical: CPU > 90%

メトリクス値: 60% → 75% → 92% → 85% → 60%

状態: 正常 Warning Critical Warning 正常

 (発生) (発生) (維持) (解消)

 Warning

 (維持)

- ① 進行中状態を持つイベントは、条件が満たされている間は継続して有効となり、条件が解除されるまで維持されます。

条件設定ガイド

❗ メトリクス名が数字で始まる場合、または特殊文字を含む場合は、`${メトリクス名}` の形式で記述する必要があります。

- 括弧 `()`, `[]`
- 演算子 `+`, `-`, `*`, `/`, `%`
- 区切り文字 `:`, `@`, `#`, `,`, `,`, (空白)
- その他の特殊文字 `!`, `^`, `&`, `|`, `~`, ````, =`

`${}` 使用例:

```
${cpu(xos)} > 50
${mem[0]} >= 100
${cpu-usage} > 80
${namespace:cpu} > 70
${metric name} > 60
${4xx_error} > 10
```

`${}` なしで使用可能: 英字、アンダースコア (`_`)、ドット (`.`) のみを含む場合

```
cpu > 80
cpu_usage > 50
CPUUtilization.Average > 0.8
```

1. 比較演算子

数値比較

```
より大きい:  cpu > 80
以上:        cpu >= 80
より小さい:  memory < 1000
以下:        memory <= 1000
等しい:      status == 200
等しくない:  error != 0
```

文字列比較

```
status == 'OK'
region == "us-east-1"
```

2. 算術演算子

- 基本的な算術演算

```
加算:   cpu + 10 >= 90
減算:   memory - 100 >= 500
乗算:   cpu * 2 >= 100
除算:   disk / 1024 >= 100
剰余:   value % 10 == 0
```

- 括弧による優先順位の制御

```
(cpu + memory) * 2 >= 200
(disk - used) / total >= 0.2
((cpu + memory) / 2) >= 50
```

- 負の値の表現

```
cpu > -100
```

3. 論理演算子

- AND 演算 (&&)

```
cpu > 80 && memory > 1000
${cpu(xos)} > 50 && ${mem(xos)} > 60
```

- OR 演算 (||)

```
cpu > 90 || memory > 90
disk < 10 || network > 1000
```

- 複合論理演算

```
(cpu > 50 && memory > 50) || disk < 20
cpu > 80 && (memory > 1000 || disk > 500)
```

4. パターンマッチング演算子

- LIKE 演算子

```
oname like 'prod-*'
url like '*error*'
message like 'WARN%'
```

- NOT LIKE 演算子

```
oname not like 'test-*'
url not like '*debug*'
```

5. 組み込み関数

- Null チェック関数

```
isNull(value)      # null かどうかを確認
isNotNull(value)   # null でないかを確認
nvl(value, 0)      # null の場合に既定値を返す
isEmpty(str)       # 空文字列かどうかを確認
isNotEmpty(str)    # 空文字列でないかを確認
```

- 集計関数

```
sum(cpu, memory, disk)    # 合計
avg(cpu, memory)          # 平均
max(cpu, memory, disk)    # 最大値
min(cpu, memory, disk)    # 最小値
count(value1, value2, value3) # 件数
```

- 数学関数

```
round(cpu, 2)    # 四捨五入（小数点以下2桁）
```

- 文字列関数

```
length(str)      # 文字列の長さ
startsWith(str, 'prefix') # 接頭辞の確認
endsWith(str, 'suffix') # 接尾辞の確認
indexOf(str, 'search') # 文字列の位置
substring(str, 0, 10) # 部分文字列
trim(str)        # 空白の削除
replace(str, 'old', 'new') # 文字列の置換
hasStr(str, 'search') # 文字列の包含判定
```

- 条件関数

- **if:** 条件式が `true` の場合は `true値` を返し、`false` の場合は `false値` を返す条件関数

形式

```
if(条件, true値, false値)
```

例

```
if(value >= 90, 'High', 'Medium') == 'High'
```

- **decode:** 値と条件を比較し、一致した条件の結果を返します。一致しない場合は既定値を返します

形式

```
decode(値, 条件1, 結果1, 条件2, 結果2, ..., 条件N, 結果N, 既定値)
```

例

```
decode(value, 1, 'Low', 2, 'Medium', 3, 'High', 'Unknown') == 'Unknown'
```

- **in:** 値が指定した候補値のいずれかと一致した場合は `true` 、それ以外は `false` を返します

形式

```
in(値, 候補値1, 候補値2, ..., 候補値N)
```

例

```
in(status, 200, 201, 204) == false
```

❗ 記述時の注意事項

- 推奨事項
 - 簡単な変数名は `${}` を使わずに記述（例: `cpu`, `memory`）
 - 特殊文字を含む場合は必ず `${}` を使用
 - 複雑な演算は括弧で明確に区切る
 - 関数名は正確に入力（大文字と小文字を区別）
- 避けるべき事項
 - 不完全な式の記述（例: `cpu >` , `memory &&` ）
 - 括弧の不一致（例: `(cpu > 80 && cpu > 70)` ）
 - 演算子の連続使用(例: `cpu >> 80` , `cpu >< 80`)

複合メトリクス

ホーム > プロジェクト選択 >  アラート通知 > イベント設定 > 複合指標 タブ

複雑なルールを利用してイベントを作成し、警告通知を送信できます。

基本オプション

- **検索:** イベント名またはイベント受信タグを選択して、目的のイベントを検索します
- **JSON ダウンロード／アップロード:** JSON エディタでイベントルールを直接編集するか、ファイルとしてダウンロードして編集後に再アップロードできます
- **通知メッセージ設定:** 受信するメッセージの詳細項目を選択し、項目ごとに内容を編集します
- **+ イベント追加:** 新しい複合メトリクスイベントを追加します

 権限

複合指標 イベントを設定するには **アラート設定** 権限が必要です。この権限は **ホーム > プロジェクト選択 > 管理 > 統合メンバー管理** で、対象メンバーの  をクリックして付与できます。

表 | 複合メトリクスイベント一覧の構成

項目	説明
イベントタイトル	ユーザーが指定したイベント名 - 最大 255 バイトまで入力可能
条件	イベント発生条件
イベント状態が解消された場合の追加通知	Critical および Warning レベルのイベントが解消された際に、RECOVERED（正常）状態の通知を受信するかどうか - 有効: 進行中 → 正常状態に切り替わると RECOVERED 通知を送信し、イベント履歴で進行中／解消状態を区別して確認できます - 無効: しきい値を超えるたびに単発のイベントが発生します

項目	説明
イベント受信タグ	該当イベントの通知を受信するユーザーまたはグループ
	イベントの修正および削除
	イベントの有効／無効状態 - 有効: 条件を満たすとイベントが発生 - 無効: イベントは動作せず、条件も評価されません

イベント設定

複合指標 イベントは、メトリクスデータのクエリ言語である **MXQL** を基にイベント条件を作成します。**チャートから作成** 機能では、MXQL の自動補完を支援するコンボボックス機能を提供します。

イベント追加

1.  **アラート通知 > イベント設定** から **複合指標** タブをクリックします。
2. 画面右側の **[+ イベント追加]** ボタンをクリックします。
3. **複合指標** 画面で **チャートから作成** をクリックします。
 - a. 提供されているテンプレートを利用できます。
4. **イベント設定** 画面で、先にイベントデータを取得し、その後イベント条件を設定します。

イベントデータ照会

イベントデータを照会してチャートを構成します。**イベント設定** 画面右側の **イベントデータ照会** で、**ウィジェット** または **テキスト** オプションを選択してイベントを設定します。

• ウィジェット テキスト

時系列チャートを構成するオプションを使用して、イベント設定に利用する **MXQL** を自動補完できます。

1. データを照会する日付を選択します。
2. データを照会する **カテゴリ** (対象) を選択します。

3. **フィルター** 項目の **+フィルター追加** をクリックしてイベント条件を作成します。
 - a. フィルター画面で、演算式、タグ、フィルター値を選択します。
 - b. **+追加** をクリックして条件を追加できます。
 - c. 条件を削除する場合は、条件の横にある **-** をクリックします。
4. **グループ化** 項目で、グループ化されたメトリクスデータを選択します。複数選択が可能です。
5. **タイムユニット** 項目で、**秒**、**分**、**時間** 単位のいずれかを選択し、データを分割する時間基準を設定します。
6. **フィールド** 項目で、イベント発生条件に使用するフィールドを選択します。複数選択が可能です。

 テキストモードで入力した MXQL は、ウィジェットモードと互換性がありません。

MXQL を直接編集できるエディタが表示されます。

1. **MXQL** を直接修正し、**照会** ボタンをクリックします。
2. 日付を選択すると、該当内容をチャートまたはグラフで確認できます。

通知

警告通知設定の基本情報を入力します。

- **イベント有効化**: トグルボタンを有効にすると、イベントが動作します。
- **レベル**: イベントレベル (Critical、Warning、Info) を選択します。
 - **イベント状態が解消された場合の追加通知**: トグルを有効にすると、イベントが解消され、正常 (RECOVERED) 状態の通知が送信されます。
- **タイトル**: イベントルールのタイトルを入力します。
- **メッセージ**: イベント発生時に受信する通知メッセージの内容を入力します。メッセージ内で変数を使用すると、イベント発生時点の実際の値を含めることができます。

Message example

 変数使用ルール



- 変数形式: \${メトリクス名}
- 設定した **カテゴリ**と同一**カテゴリ** 内のメトリクスのみ変数として使用できます。
- 複数の変数を組み合わせて、詳細なメッセージを構成できます。
- 使用可能な変数一覧は **メトリクス照会** で確認できます。

イベント発行条件

- **データ照会範囲**: リアルタイムデータの照会範囲を選択します。
 - 選択可能な範囲: 30秒, 1分, 2分, 5分, 10分, 30分, 1時間, 2時間, 3時間
- **条件**: イベント発行条件を入力します。イベントデータ照会に含まれるフィールドのみ使用できます。

付加情報

- **インターバル**: 選択した時間間隔（1分, 5分, 10分, 30分, 1時間, 12時間）ごとにアラート条件を確認します。
- **サイレンス**: イベント発生後、選択した時間（unset, 1分, 5分, 10分, 30分, 1時間, 12時間）の間、同一イベントは発生しません。
- **イベント動作時間**: 勤務時間、非勤務時間、点検時間など、特定の時間帯にイベントが動作するようタグを設定します。タグを設定しない場合、イベントは有効な状態で24時間常時動作します。
 1. **+追加** をクリックします。
 2. **イベント動作時間で+新しいタグ作成** をクリックします。
 3. **イベント動作時間タグ作成** で **タグ名**、**曜日**、**時間**、**色** を選択し、**[適用]** ボタンをクリックします。
 4. 作成したタグはタグ一覧で確認でき、チェックボックスを選択すると適用されます。
 5. 修正または削除するタグの をクリックし、**イベント動作時間タグ修正** で編集または削除できます。



タグを削除すると、そのタグが適用されているすべてのユーザーから削除されます。ただし、イベントルールで使用中のタグは削除できません。

- **イベント受信タグ**: 受信タグを設定すると、該当タグを持つプロジェクトメンバーに通知が送信されます。受信タグを設定しない場合、プロジェクト内のすべてのユーザーに通知が送信されます。

1. **+ タグ追加** または **+** をクリックします。
2. **イベント受信タグ** 画面下部の **+ 新しいタグ作成** をクリックします。
3. **タグ作成** 画面でタグ名を入力し、色を選択した後、**[タグ作成]** ボタンをクリックします。
4. 作成したタグは、**タグ一覧** の  をクリックして修正または削除できます。
5. **イベント受信タグ** 画面の **タグ一覧** から適用したいタグを選択します。

✔ イベント設定時にイベント受信タグを選択すると、該当タグを持つプロジェクトメンバーおよび 3rd-party プラグインに通知を送信できます。**アラート通知 > 通知設定** で、それぞれにタグを割り当てることができます。

イベントルールテスト

設定した条件を過去データに基づいてテストし、どの程度のイベントが発生するかを予測できます。

1. イベントルールをテストする日付を選択し、**[確認]** ボタンをクリックします。
2. **[実行]** ボタンをクリックして、イベント発生件数を確認します。

イベント修正／削除

1. **アラート通知 > 通知設定** の **複合指標** タブに移動します。
2. 複合メトリクス一覧で、修正または削除したいイベントの  をクリックします。
3. **通知設定** 画面でオプションを修正し、**[保存]** ボタンをクリックします。
 - a. 選択したイベントを削除する場合は、**通知設定** 画面右上の **[削除]** ボタンをクリックします。

イベント共有

メトリクスイベント設定を JSON ファイルとして保存し、他のユーザーと設定を共有したり、他のユーザーの設定を取り込んだりすることができます。

- JSON ファイル名: **event-rules- YYYYMMDD .json**

エクスポート

1. 画面右上の [JSON ] ボタンをクリックします。
2. JSON 編集画面が表示されたら、[ エクスポート] ボタンをクリックします。
 - イベントを検索した状態でエクスポート機能を使用すると、検索結果のみが JSON ファイルとしてダウンロードされます。
3. JSON ファイルがダウンロードされたら、共有したい他のユーザーに渡してください。

インポート

1. 画面右上の [] ボタンをクリックします。
2. エクスポート 機能でダウンロードした JSON ファイルを選択します。
3. JSON 編集画面が表示されたら、[一覧に追加] または [上書き] を選択します。

 同じ種類の製品間での利用を推奨します。異なる製品のプロジェクトからイベント設定を取り込むことは可能ですが、正常に動作しない場合があります。

JSON 形式で編集する

1. 画面右上の [JSON ] ボタンをクリックします。
2. 編集画面が表示されたら、JSON 形式に従って内容を修正してください。
3. 修正が完了したら、画面下部の [保存] ボタンをクリックします。

 修正した内容が JSON 形式に合っていない場合、画面下部にエラーメッセージが表示され、保存できません。表示されるエラーメッセージは、形式エラーの内容によって異なる場合があります。

JSON error

通知メッセージ設定

受信するメッセージの詳細項目を選択し、項目ごとに内容を編集できます。

付録

MXQL 文法ガイド

複合指標 は MXQL クエリを基に動作します。詳細な文法については、[MXQL 文法ガイド](#) を参照してください。

異常値

ホーム > プロジェクト選択 >  **アラート通知** > **イベント設定** > **異常検知** タブ

異常検知 は、メトリクスデータから異常な上昇または下降パターンを検出し、通常とは異なる動きが検知された場合にイベントを発生させる機能です。

基本オプション

- **検索:** イベント名またはイベント受信タグを選択して、目的のイベントを検索します。
- **+ イベントルール追加:** 新しい異常値イベントを追加します。

表 | 異常値イベント一覧の構成

項目	説明
イベントメッセージ	イベント発生時に送信されるメッセージ
カテゴリ	メトリクスデータを分類する単位で、メトリクスイベント設定の基準となるカテゴリ
フィールド	異常を検知する指標（メトリクスフィールド）
フィルター	異常値の対象となるデータ範囲を制限する条件
上昇／下降パターン	上昇または下降パターンの感度
イベント受信タグ	該当イベントの通知を受信するユーザーまたはグループ
	イベント設定の修正
	イベントの削除
	イベントの有効／無効状態 - 有効: 条件を満たすとイベントが発生 - 無効: イベントは動作せず、条件も評価されません

イベント設定

メトリクスデータを基に、値の動きが通常とは異なり上昇または下降した場合に、警告通知を送信するよう設定できます。

イベントルール追加

1. 画面右側の [+ イベントルール追加] ボタンをクリックします。
2. イベント基本情報, メトリクス, 異常値, 付加情報 を順に設定します。

イベント基本情報

1. イベントレベル (Critical, Warning, Info) を選択します。
 - イベントの重要度および通知レベルが決定されます。
2. イベント発生時に受信する通知メッセージを入力します。
 - メッセージ内で変数を使用すると、イベント発生時点の実際の値を含めることができます。

❗ 使用可能な変数

- \$pcode : プロジェクトコード
- \$category : カテゴリ
- \$field : フィールド
- \$value : 発生値
- \$oname : イベント対象

メトリクス

1. カテゴリ を選択します。
 - 異常を検知するメトリクスデータの基準となるカテゴリです。

2. イベント発生条件に使用する **フィールド** を選択します。複数選択が可能です。
3. イベント対象を絞り込むためのフィルターを選択します。複数選択が可能です。

異常値

1. **上昇パターン** と **下降パターン** の感度を選択します。
2. 各パターンのトグルボタンをクリックして、有効または無効を切り替えます。
 - 上昇パターンのみを検知する場合: 上昇パターン **ON**, 下降パターン **OFF**
 - 下降パターンのみを検知する場合: 下降パターン **ON**, 上昇パターン **OFF**

付加情報

イベント動作時間

勤務時間、非勤務時間、点検時間など、特定の時間帯にイベントが動作するようタグを設定します。タグを設定しない場合、イベントは有効な状態で 24 時間常時動作します。

1. **+ 追加** をクリックします。
2. **イベント動作時間** で **+ 新しいタグ作成** をクリックします。
3. **イベント動作時間タグ作成** で **タグ名**、**曜日**、**時間**、**色** を選択し、**[適用]** ボタンをクリックします。
4. 作成したタグはタグ一覧で確認でき、チェックボックスを選択すると適用されます。
5. 修正または削除するタグの  をクリックし、**イベント動作時間タグ修正** で編集または削除できます。

 タグを削除すると、そのタグが適用されているすべてのユーザーから削除されます。ただし、イベントルールで使用中のタグは削除できません。

イベント受信タグ

イベント受信タグを設定すると、該当タグを持つプロジェクトメンバーに通知が送信されます。受信タグを設定しない場合、プロジェクト内のすべてのユーザーに通知が送信されます。

1. **+ タグ追加** または **+** をクリックします。
2. **イベント受信タグ** 画面下部の **+ 新しいタグ作成** をクリックします。

3. タグ作成 画面でタグ名を入力し、色を選択した後、[タグ作成] ボタンをクリックします。
4. 作成したタグは、タグ一覧の  をクリックして修正または削除できます。
5. イベント受信タグ 画面の タグ一覧 から適用したいタグを選択します。

✔ イベント設定時にイベント受信タグを選択すると、該当タグを持つプロジェクトメンバーおよび 3rd-party プラグインに通知を送信できます。アラート通知 > 通知設定 で、それぞれにタグを割り当てることができます。

イベント修正

1. アラート通知 > イベント設定 の 異常検知 タブに移動します。
2. 異常値 一覧で、修正するイベントの右側にある  をクリックします。
3. 異常値 画面でオプションを修正し、[保存] ボタンをクリックします。

イベント削除

1. アラート通知 > イベント設定 の 異常検知 タブに移動します。
2. 異常値 一覧で、削除するイベントの右側にある  をクリックします。
3. 削除確認画面で [削除] ボタンをクリックします。

リアルタイムログイベント

ホーム > プロジェクト選択 >  アラート通知 > イベント設定 > リアルタイムログ タブ

収集したログデータを条件に合わせてフィルタリングして警告通知を設定できます。ログイベントを使用するには、ログモニタリングを有効化する必要があります。**ログモニタリングの有効化**については、各製品のログドキュメントを参照してください。

- ・ **リアルタイムログイベント**: リアルタイムで収集したログで、設定したイベント条件が満たされるとイベントが発生します。

基本オプション

- ・ **検索**: イベント名またはイベント受信タグを選択して目的のイベントを検索
- ・ **JSONダウンロード/アップロード**: イベントルールはJSONエディタで直接編集するか、ファイルにダウンロードして修正後に再アップロード可能
- ・ **通知メッセージ設定**: 受信するメッセージの詳細項目を選択し、項目ごとに内容を編集
- ・ **+ イベント追加**: 新しいログイベントを追加

表 | リアルタイムログイベント一覧構成

項目	説明
イベント名	通知タイトルとして使用されるイベント名 - 最大255バイトまで入力可能
カテゴリ	ログ分類名（ログフォルダ名）
検索キー	ログデータで検索したい項目 - 例: HTTPレスポンスステータスコードの値にアクセスする場合、検索キー <code>status</code>
検索値	検索キーに対応する実際のデータ。ログに入力した文字列が含まれる場合に警告通知を送信 - 例: 検索キー <code>status</code> 、検索値 <code>200</code> を設定した場合、HTTPステータスコード200を含むログデータ収集時に警告通知が発生

項目	説明
イベント発生一時停止	イベント通知発生後、同じイベントの発生を一定時間停止 - 選択可能時間: 使用しない、5分、10分、15分、20分、30分、1時間、2時間、3時間、6時間、12時間、1日 - 解消通知が有効な場合、正常（RECOVERED）状態の通知受信後、選択した時間中は同じイベントが発生しても通知されない
イベント対象フィルタリング	イベントが適用される対象の条件 - 対象条件を設定しない場合、収集されるすべてのデータに対して通知を判断 - 選択入力: タグ、演算子、値を選択して条件を構成 - 直接入力: 条件を直接入力して対象を指定
イベント受信タグ	イベント通知を受け取るユーザーまたはグループ（受信タグ）
	イベントの編集と削除
	イベントの有効化状態 - 有効: 条件を満たすとイベント発生 - 無効: イベントが動作せず、条件をチェックしない

イベント設定

ログイベントは、リアルタイムで収集されるログ、または特定条件を満たすログが一定数蓄積された際に警告通知を設定できる機能です。

リアルタイムログイベントの追加

1. アラート通知 > イベント設定のリアルタイムログタブをクリックしてください。
2. 画面右側の [+ イベント追加] ボタンをクリックしてください。
3. イベントルール追加ウィンドウでログイベント条件を設定してください。
 - イベント名: イベントタイトルとして使用する名前を入力してください。

- **イベント有効化:** トグルボタンを有効にすると、条件を満たしたときにイベントが発生します。
- **レベル:** イベントレベル (**Critical**、**Warning**、Info) を選択してください。
- **メッセージ:** イベント発生時に受け取る通知メッセージを入力してください。メッセージに変数を使用して、イベント発生時点の実際の値を含めることができます。

✔ 変数使用ルール

- 変数形式: \${指標名}
- 設定したカテゴリと同じカテゴリ内の指標のみ変数として使用できます。
- 複数の変数を組み合わせて詳細なメッセージを構成できます。
- 使用可能な変数一覧は**メトリクス照会**で確認できます。

- **カテゴリ:** ログ分類名（ログフォルダ名）をリストから選択するか、直接入力してください。カテゴリを選択してから検索キーと検索値を入力できます。
- **検索キー:** ログデータで検索したい項目を入力してください。
 - 例: HTTPレスポンスステータスコードの値にアクセスする場合、**検索キー** status
- **検索値:** 検索キーに対応する実際のデータを入力してください。ログに入力した文字列が含まれる場合に警告通知を送信します。
 - 例: **検索キー** status、**検索値** 200 を設定した場合、HTTPステータスコード200を含むログデータ収集時に警告通知が発生
- **イベント対象フィルタリング:** イベントが適用される対象の条件を入力してください。入力値がない場合、収集されるすべてのデータに対して通知を判断します。
 - **選択入力:** タグ、演算子、値を選択して対象条件を設定してください。

❗ 条件式の記述方法とサポートされる演算子については、[条件設定ガイド](#)を参照してください。

- **直接入力:** 条件を直接入力してください。
- **イベント発生一時停止:** 通知を一時停止する時間を選択してください。イベント発生後、設定した時間中は同じイベントが発生しません。

- 選択可能時間: 使用しない、1分、2分、3分、5分、10分、20分、30分、1時間、3時間、6時間、12時間
- イベント動作時間: 特定の時間帯（業務/非業務/メンテナンス時間など）にイベントが動作するようにタグを設定してください。タグを設定しない場合、イベントが有効な状態で24時間常時動作します。
 1. + 追加をクリックしてください。
 2. イベント動作時間で**+新しいタグを作成**をクリックしてください。
 3. イベント動作時間タグの作成でタグ名、曜日、時間、色を選択し、[適用] ボタンをクリックしてください。
 4. 作成したタグはタグリストで確認でき、チェックボックスを選択すると適用されます。
 5. 変更または削除するタグの  アイコンをクリックして、イベント動作時間タグの編集でタグを編集または削除できます。

 タグを削除すると、そのタグが適用されているすべてのユーザーのタグが削除されます。ただし、イベントルールで使用中のタグは削除できません。

- イベント受信タグ: イベントに受信タグを設定すると、そのタグを持つプロジェクトメンバーに通知が送信されます。受信タグを設定しない場合、プロジェクト内のすべてのユーザーに通知が送信されます。
 1. + タグ追加または+をクリックしてください。
 2. イベント受信タグウィンドウ下部の**+ 新しいタグを作成**をクリックしてください。
 3. タグ作成ウィンドウでタグ名を入力し、色を選択してから、[タグ作成] ボタンをクリックしてタグを作成してください。
 4. 作成されたタグはタグリストの  アイコンをクリックして編集または削除してください。
 5. イベント受信タグウィンドウのタグリストから目的のタグを選択すると適用されます。

 イベント設定時にイベント受信タグを選択して、そのタグを持つプロジェクトメンバーとサードパーティプラグインに通知を送信できます。アラート通知 > 通知設定でプロジェクトメンバーとサードパーティプラグインにそれぞれタグを指定できます。

4. イベントルール追加ウィンドウで設定が完了したら、[保存] ボタンをクリックしてください。

イベントの編集/削除

1. アラート通知 > イベント設定のリアルタイムログタブに移動してください。
2. ログイベントリストで編集または削除するイベントの  アイコンをクリックしてください。
3. イベント設定画面でオプションを変更し、[保存] ボタンをクリックしてください。
 - a. 選択したイベントを削除するには、イベント設定画面の右上の [削除] ボタンをクリックしてください。

イベントの共有

メトリクスイベント設定を JSON ファイルとして保存し、他のユーザーと設定を共有したり、他のユーザーの設定を取り込んだりすることができます。

- JSON ファイル名: `event-rules-YYYYMMDD.json`

エクスポート

1. 画面右上の [JSON ] ボタンをクリックします。
2. JSON 編集画面が表示されたら、[ エクスポート] ボタンをクリックします。
 - イベントを検索した状態でエクスポート機能を使用すると、検索結果のみが JSON ファイルとしてダウンロードされます。
3. JSON ファイルがダウンロードされたら、共有したい他のユーザーに渡してください。

インポート

1. 画面右上の [] ボタンをクリックします。
2. エクスポート 機能でダウンロードした JSON ファイルを選択します。
3. JSON 編集画面が表示されたら、[一覧に追加] または [上書き] を選択します。

 同じ種類の製品間での利用を推奨します。異なる製品のプロジェクトからイベント設定を取り込むことは可能ですが、正常に動作しない場合があります。

JSON 形式で編集する

1. 画面右上の [JSON ] ボタンをクリックします。
2. 編集画面が表示されたら、JSON 形式に従って内容を修正してください。
3. 修正が完了したら、画面下部の [保存] ボタンをクリックします。

❗ 修正した内容が JSON 形式に合っていない場合、画面下部にエラーメッセージが表示され、保存できません。表示されるエラーメッセージは、形式エラーの内容によって異なる場合があります。

JSON error

付録

発生条件・対象選択ガイド

メトリクスアラートにおけるイベントの発生条件とイベント対象の選択は、同一の構文を使用します。ただし、イベント発生条件ではフィールド（Field）のキーを変数として使用し、イベント対象の選択ではタグ（Tag）のキーを変数として使用します。

レベル動作ガイド

イベントレベルは、危険（Critical）、警告（Warning）、正常（Info）の3段階に分類されます。

1. 優先順位に基づくイベント発生

複数のレベル条件を同時に満たした場合、最も優先度の高いイベントのみが発生します。

例

設定:

- Warning: CPU > 70%

- Critical: CPU > 90%

現在の状態: CPU 95%
→ 結果: Critical イベントのみ発生 (Warning は抑制される)

2. レベル上昇

進行中状態を持つイベントにおいて、低いレベルのイベントが進行中の状態で高いレベルの条件を満たすと、両方のレベルが進行中状態になります。

例

設定:
- Warning: CPU > 70%
- Critical: CPU > 90%

シナリオ:
1) CPU 80% → Warning 発生 (進行中)
2) CPU 95% → Warning (進行中)、Critical (進行中)

3. レベル下降

進行中状態を持つイベントで、高いレベルから低いレベルへ遷移した場合、高いレベルは解消され、低いレベルが維持されます。

例

設定:
- Warning: CPU > 70%
- Critical: CPU > 90%

シナリオ:
1) CPU 97% → Critical 発生
2) CPU 85% → Critical 解消、Warning 維持
3) CPU 65% → Warning 解消

4. 動作フロー

- Warning: CPU > 70%
- Critical: CPU > 90%

メトリクス値: 60% → 75% → 92% → 85% → 60%

状態: 正常 Warning Critical Warning 正常

 (発生) (発生) (維持) (解消)

 Warning

 (維持)

① 進行中状態を持つイベントは、条件が満たされている間は継続して有効となり、条件が解除されるまで維持されます。

条件設定ガイド

⚠️ メトリクス名が数字で始まる場合、または特殊文字を含む場合は、`${メトリクス名}` の形式で記述する必要があります。

- 括弧 () , []
- 演算子 + , - , * , / , %
- 区切り文字 : , @ , # , , (空白)
- その他の特殊文字 ! , ^ , & , | , ~ , ` , =

`{}` 使用例:

```

${cpu(xos)} > 50
${mem[0]} >= 100
${cpu-usage} > 80
${namespace:cpu} > 70
${metric name} > 60
${4xx_error} > 10

```

\${}なしで使用可能: 英字、アンダースコア (_)、ドット (.) のみを含む場合

```
cpu > 80
cpu_usage > 50
CPUUtilization.Average > 0.8
```

1. 比較演算子

数値比較

```
より大きい:  cpu > 80
以上:        cpu >= 80
より小さい:  memory < 1000
以下:        memory <= 1000
等しい:      status == 200
等しくない:  error != 0
```

文字列比較

```
status == 'OK'
region == "us-east-1"
```

2. 算術演算子

- 基本的な算術演算

```
加算:  cpu + 10 >= 90
減算:  memory - 100 >= 500
乗算:  cpu * 2 >= 100
除算:  disk / 1024 >= 100
剰余:  value % 10 == 0
```

- 括弧による優先順位の制御

```
(cpu + memory) * 2 >= 200
(disk - used) / total >= 0.2
((cpu + memory) / 2) >= 50
```

- 負の値の表現

```
cpu > -100
```

3. 論理演算子

- AND 演算 (&&)

```
cpu > 80 && memory > 1000
${cpu(xos)} > 50 && ${mem(xos)} > 60
```

- OR 演算 (||)

```
cpu > 90 || memory > 90
disk < 10 || network > 1000
```

- 複合論理演算

```
(cpu > 50 && memory > 50) || disk < 20
cpu > 80 && (memory > 1000 || disk > 500)
```

4. パターンマッチング演算子

- LIKE 演算子

```
oname like 'prod-*'
url like '*error*'
message like 'WARN%'
```

- NOT LIKE 演算子

```
oname not like 'test-*'
url not like '*debug*'
```

5. 組み込み関数

- Null チェック関数

```
isNull(value)      # null かどうかを確認
isNotNull(value)   # null でないかを確認
nvl(value, 0)      # null の場合に既定値を返す
isEmpty(str)       # 空文字列かどうかを確認
isNotEmpty(str)    # 空文字列でないかを確認
```

- 集計関数

```
sum(cpu, memory, disk)    # 合計
avg(cpu, memory)          # 平均
max(cpu, memory, disk)    # 最大値
min(cpu, memory, disk)    # 最小値
count(value1, value2, value3) # 件数
```

- 数学関数

```
round(cpu, 2)    # 四捨五入（小数点以下2桁）
```

- 文字列関数

```
length(str)      # 文字列の長さ
startsWith(str, 'prefix') # 接頭辞の確認
endsWith(str, 'suffix') # 接尾辞の確認
indexOf(str, 'search') # 文字列の位置
substring(str, 0, 10) # 部分文字列
trim(str)        # 空白の削除
replace(str, 'old', 'new') # 文字列の置換
hasStr(str, 'search') # 文字列の包含判定
```

- 条件関数

- **if:** 条件式が `true` の場合は `true値` を返し、`false` の場合は `false値` を返す条件関数

形式

```
if(条件, true値, false値)
```

例

```
if(value >= 90, 'High', 'Medium') == 'High'
```

- **decode:** 値と条件を比較し、一致した条件の結果を返します。一致しない場合は既定値を返します

形式

```
decode(値, 条件1, 結果1, 条件2, 結果2, ..., 条件N, 結果N, 既定値)
```

例

```
decode(value, 1, 'Low', 2, 'Medium', 3, 'High', 'Unknown') == 'Unknown'
```

- **in**: 値が指定した候補値のいずれかと一致した場合は `true`、それ以外は `false` を返します

形式

```
in(値, 候補値1, 候補値2, ..., 候補値N)
```

例

```
in(status, 200, 201, 204) == false
```

❗ 記述時の注意事項

- 推奨事項
 - 簡単な変数名は `${}` を使わずに記述（例: `cpu`, `memory`）
 - 特殊文字を含む場合は必ず `${}` を使用
 - 複雑な演算は括弧で明確に区切る
 - 関数名は正確に入力（大文字と小文字を区別）
- 避けるべき事項
 - 不完全な式の記述（例: `cpu >` , `memory &&` ）
 - 括弧の不一致（例: `(cpu > 80 && cpu > 70)` ）
 - 演算子の連続使用(例: `cpu >> 80` , `cpu >< 80`)

複合ログイベント設定

ホーム > プロジェクト選択 >  アラート通知 > イベント設定 > 複合ログ タブ

収集したログデータを条件に合わせてフィルタリングして警告通知を設定できます。ログイベントを使用するには、ログモニタリングを有効化する必要があります。**ログモニタリングの有効化**については、各製品のログドキュメントを参照してください。

- **複合ログイベント**: 設定したデータ照会範囲内のログを基準に、ログ検索条件を満たすログ数が閾値以上の場合にイベントが発生します。

ⓘ 権限

複合メトリクスイベントを設定するには**通知設定権限**が必要です。通知設定権限は、**ホーム画面 > プロジェクト選択 > 管理 > 統合メンバー管理**で、権限を付与するメンバーの  アイコンをクリックして設定できます。

基本オプション

- **検索**: イベント名またはイベント受信タグを選択して目的のイベントを検索
- **JSONダウンロード/アップロード**: イベントルールはJSONエディタで直接編集するか、ファイルにダウンロードして修正後に再アップロード可能
- **通知メッセージ設定**: 受信するメッセージの詳細項目を選択し、項目ごとに内容を編集
- **+ イベント追加**: 新しいログイベントを追加

表 | 複合ログイベント一覧構成

項目	説明
イベント名	通知タイトルとして使用されるイベント名
テンプレート	複合ログテンプレート

項目	説明
カテゴリ	ログ分類名（ログフォルダ名）
ルール	イベントが発生する条件
イベント状態が解消されたら追加通知	CriticalとWarningレベルのイベントが解消されたとき、正常（RECOVERED）状態の通知を受信するかどうか - 有効: 進行中 → 正常状態に移行したときRECOVERED通知を送信し、イベント履歴で進行中/解消状態を区別して確認可能 - 無効: 閾値を超えるたびに単発のイベントが発生
イベント発生一時停止	イベント通知発生後、同じイベントの発生を一定時間停止 - 選択可能時間: 使用しない、5分、10分、15分、20分、30分、1時間、2時間、3時間、6時間、12時間、1日 - 解消通知が有効な場合、正常（RECOVERED）状態の通知受信後、選択した時間中は同じイベントが発生しても通知されない
インターバル	ログデータを照会する時間範囲
イベント受信タグ	イベント通知を受け取るユーザーまたはグループ（受信タグ）
	イベントの編集と削除
	イベントの有効化状態 - 有効: 条件を満たすとイベント発生 - 無効: イベントが動作せず、条件をチェックしない

イベント設定

設定したデータ照会範囲内のログを基準に、ログ検索条件を満たすログ数が閾値以上の場合に警告通知を設定できる機能です。

複合ログイベントの追加

1. アラート通知 > イベント設定の複合ログ タブをクリックしてください。
2. 画面右側の [+ イベント追加] ボタンをクリックしてください。
3. イベントルール追加ウィンドウでログイベント条件を設定してください。詳細については、[基本設定](#)、[詳細設定](#)、[追加設定](#)を参照してください。

複合ログイベントの追加

4. イベントルール追加ウィンドウで設定が完了したら、[保存] ボタンをクリックしてください。

基本設定

- ・ **イベント名**: イベントタイトルとして使用する名前を入力してください。
- ・ **イベント有効化**: トグルボタンを有効にすると、条件を満たしたときにイベントが発生します。
- ・ **レベル**: イベントレベル (**Critical**、**Warning**、**Info**) を選択してください。

詳細設定

[テンプレート \(プリセット\)](#)、[メッセージ](#)、[カテゴリ](#)、[フィルター](#)、[イベント発行条件](#)を設定します。

テンプレート (プリセット)

定義済みのプリセットを選択すると、[フィルター](#)、[クエリタイプ](#)、[イベント発行条件](#)が自動的に設定されます。プリセットを選択後、必要に応じて変更できます。

複合ログテンプレートの種類



プリセット	クエリタイプ	デフォルトルール
使用しない	none	rows > 10
2xxステータスコード件数	count	count > 10



プリセット	クエリタイプ	デフォルトルール
3xxステータスコード件数	count	count > 10
4xxステータスコード件数	count	count > 10
5xxステータスコード件数	count	count > 10
正常ステータスコード (2xx, 3xx) 件数	count_v2	include_minus_exclude_count > 10
エラーステータスコード (4xx, 5xx) 件数	count_v2	include_minus_exclude_count > 10
エラー受信件数	count	count > 10
例外受信件数	count	count > 10
毎秒ログ発生件数	rps	rps > 100
フィールド集計 (Aggregation)	aggr	avg > 1000
比率 (Rate)	rate	rate > 10
ユニーク値数 (Cardinality)	cardinality	cardinality > 100
パーセンタイル (Percentile)	percentile	p99 > 3000

メッセージ

イベント発生時に受け取る通知メッセージを入力してください。メッセージに変数を使用して動的なメッセージを構成できます。

- 例示: メッセージ `${host}` でエラー `${count}` 件発生 → 通知メッセージ web-01 でエラー15件発生

表 | メッセージ変数

変数	説明	使用条件
<code>\${count}</code>	条件に一致するログ件数	常に使用可能
<code>\${groupField}</code>	グループ化フィールド名	グループ化通知のみ
<code>\${groupValue}</code>	グループ化フィールド値	グループ化通知のみ
<code>\${フィールド名}</code>	グループ化フィールド値（フィールド名で直接参照）	グループ化通知のみ

❗ `${count}` 変数はグループ化の有無に関わらず、すべての通知で使用できます。 `${groupField}`、`${groupValue}`、`${フィールド名}` 変数はグループ化フィールドが設定されている場合のみ置換されます。

カテゴリ

ログ分類名（ログフォルダ名）をリストから選択するか、直接入力してください。カテゴリを選択してから検索キーと検索値を入力できます。

フィルター

Luceneクエリ構文でログ検索条件を入力します。2種類の入力モードをサポートします。

- フィルターモード [▼](#): フィールドと値を選択してクエリを構成します。
- コードモード [<>](#): Luceneクエリを直接入力します。

表 | Luceneクエリ例

クエリ	説明
<code>level:ERROR</code>	levelフィールドがERRORのログ
<code>status_nxx:status_4xx OR status_nxx:status_5xx</code>	4xxまたは5xxステータスコード
<code>appender:accessLog</code>	accessLog appenderのログ
<code>response_time.n:*</code>	response_time.nフィールドが存在するログ

クエリ	説明
host:web-01 AND level:ERROR	web-01ホストのエラーログ

イベント発行条件

クエリタイプ別に設定します。選択したテンプレート（プリセット）またはクエリタイプに応じて、追加設定項目が表示されます。

表 | イベント発行条件キーの意味

キー	意味
rows / count	フィルター条件に一致するログ件数
rps	毎秒ログ発生件数（件数 / インターバル秒）
avg	対象フィールド値の平均
sum	対象フィールド値の合計
min	対象フィールド値の最小値
max	対象フィールド値の最大値
rate	分子件数 / 分母件数 × 100（%）
cardinality	対象フィールドのユニーク値数
p99 / p95 / p90 / p75 / p50	対象フィールドのパーセンタイル値
include_minus_exclude_count	含むフィルター件数 - 除外フィルター件数

- **件数（count / none）**：設定したフィルター条件に一致するログの件数をカウントします。

表 | 件数設定項目

設定項目	説明
フィルター	ログ検索条件（Luceneクエリ）
イベント発行条件	件数 または 条件に一致するログ {演算子} {閾値}（例: count > 10）

- **毎秒ログ発生件数（rps）**：設定したフィルター条件に一致するログの毎秒発生件数（Requests Per Second）を計算します。インターバル時間中の総件数をインターバル秒数で割って算出します。

表 | 毎秒ログ発生件数設定項目

設定項目	説明
フィルター	ログ検索条件（Luceneクエリ）
イベント発行条件	RPS {演算子} {閾値}（例: rps > 0.5）

- **フィールド集計（aggr）**：指定した数値フィールドの集計値（平均、合計、最小値、最大値）を計算します。イベント発行条件で集計方式をドロップダウンから選択できます。

表 | フィールド集計設定項目

設定項目	説明
フィルター	ログ検索条件（Luceneクエリ）
対象フィールド	集計対象の数値フィールド（例: response_time.n）
イベント発行条件	平均 / 合計 / 最小値 / 最大値 {演算子} {閾値}（例: avg > 1000）

- **比率（rate）**：分子フィルターと分母フィルターの件数比率（%）を計算します。全ログに対する特定条件の比率を監視する際に有用です。

表 | 比率設定項目

設定項目	説明
フィルター（分子）	比率計算の分子となるログ条件（例: <code>level:ERROR</code> ）
分母フィルター	比率計算の分母となるログ条件（例: <code>level:*</code> ）
イベント発行条件	比率（%） {演算子} {閾値}（例: <code>rate > 10</code> ）

- 例示: フィルター `level:ERROR`、分母フィルター `level:*`、ルール `rate > 10` → 全ログ中のエラー比率が10%を超えると通知が発生
- **ユニーク値数（cardinality）** : 指定したフィールドのユニーク値数を計算します。異常に多くの種類の値が発生する場合を検出する際に有用です。

表 | ユニーク値数設定項目

設定項目	説明
フィルター	ログ検索条件（Luceneクエリ）
対象フィールド	ユニーク値を計算するフィールド（例: <code>host</code> ）
イベント発行条件	ユニーク値数 {演算子} {閾値}（例: <code>cardinality > 100</code> ）

- **パーセンタイル（percentile）** : 指定した数値フィールドのパーセンタイル値を計算します。レスポンスタイムのテールレイテンシを監視する際に有用です。パーセンタイルの選択を変更すると、イベント発行条件のキーが自動的に更新されます。

表 | パーセンタイル設定項目

設定項目	説明
フィルター	ログ検索条件（Luceneクエリ）
対象フィールド	パーセンタイルを計算する数値フィールド（例: <code>response_time.n</code> ）
パーセンタイル	P99、P95、P90、P75、P50から選択

設定項目	説明
イベント発行条件	P99 / P95 / ... {演算子} {閾値} (例: p99 > 3000)

表 | パーセンタイルの意味

パーセンタイル	意味
P50	中央値。全データの50%がこの値以下
P75	全データの75%がこの値以下
P90	全データの90%がこの値以下
P95	全データの95%がこの値以下
P99	全データの99%がこの値以下

- 含む/除外件数 (count_v2) : 含むフィルターの件数から除外フィルターの件数を引いた純件数を計算します。

表 | 含む/除外件数設定項目

設定項目	説明
フィルター (含む)	含むログ条件 (例: status_nxx:status_4xx OR status_nxx:status_5xx)
フィルター除外	除外するログ条件 (例: status_nxx:status_4xx)
分母フィルター	比率計算用の分母 (任意)
イベント発行条件	含む-除外件数 {演算子} {閾値} (例: include_minus_exclude_count > 10)

- 例示: フィルター status_nxx:status_4xx OR status_nxx:status_5xx、フィルター除外 status_nxx:status_4xx
→ 4xx+5xxから4xxを除いた純5xx件数が10を超えると通知が発生

グループ化フィールド（Group By）

グループ化フィールドを設定すると、そのフィールドの値ごとに独立して通知条件が判定されます。ドロップダウンからグループ化するフィールドを選択します。ドロップダウンをクリックすると、現在のカテゴリで使用可能なフィールド一覧がオートコンプリートで表示されます。

- 例示: フィルター `level:ERROR`、グループ化フィールド `host`、ルール `rows > 3`

表 | グループ化設定の比較

項目	グループ化なし	グループ化あり
判定単位	全ログを1つに集計	グループ値ごとに独立して集計
通知発生	1件	条件を満たすグループ数分
通知タイトル	イベントタイトル	[グループ値] イベントタイトル
Stateful	全体基準での状態遷移	グループごとに独立した状態遷移

- `[web-01] エラー通知` — web-01でエラー8件 → 通知発生
- `[web-02] エラー通知` — web-02でエラー5件 → 通知発生
- `[api-01] エラー通知` — api-01でエラー2件 → 条件未達、通知なし

❗ イベント状態が解消されたら追加通知を有効にした場合、グループごとに独立してABNORMAL → RECOVERED状態が遷移します。例えば web-01 のエラーが解消されると `[web-01] RECOVERED 通知` が発生し、他のホストの状態には影響しません。

追加設定

- **インターバル:** 選択した時間（`unset`、`1分`、`5分`、`10分`、`30分`、`1時間`、`12時間`）間隔で通知条件を確認します。
- **無音:** イベント発生後、選択した時間（`unset`、`1分`、`5分`、`10分`、`30分`、`1時間`、`12時間`）中は同じイベントが発生しません。

- **イベント動作時間:** 特定の時間帯（業務/非業務/メンテナンス時間など）にイベントが動作するようにタグを設定してください。タグを設定しない場合、イベントが有効な状態で24時間常時動作します。
 1. **+追加**をクリックしてください。
 2. **イベント動作時間で**+新しいタグを作成****をクリックしてください。
 3. **イベント動作時間タグの作成でタグ名、曜日、時間、色**を選択し、**[適用]** ボタンをクリックしてください。
 4. 作成したタグはタグリストで確認でき、チェックボックスを選択すると適用されます。
 5. 変更または削除するタグの  アイコンをクリックして、**イベント動作時間タグの編集**でタグを編集または削除できます。

 タグを削除すると、そのタグが適用されているすべてのユーザーのタグが削除されます。ただし、イベントルールで使用中のタグは削除できません。

- **イベント受信タグ:** イベントに受信タグを設定すると、そのタグを持つプロジェクトメンバーに通知が送信されます。受信タグを設定しない場合、プロジェクト内のすべてのユーザーに通知が送信されます。
 1. **+タグ追加**または**+**をクリックしてください。
 2. **イベント受信タグウィンドウ下部の**+新しいタグを作成****をクリックしてください。
 3. **タグ作成ウィンドウ**でタグ名を入力し、色を選択してから、**[タグ作成]** ボタンをクリックしてタグを作成してください。
 4. 作成されたタグはタグリストの  アイコンをクリックして編集または削除してください。
 5. **イベント受信タグウィンドウのタグリスト**から目的のタグを選択すると適用されます。

 イベント設定時にイベント受信タグを選択して、そのタグを持つプロジェクトメンバーとサードパーティプラグインに通知を送信できます。**アラート通知 > 通知設定**でプロジェクトメンバーとサードパーティプラグインにそれぞれタグを指定できます。

シミュレーション

通知を保存する前に、過去のデータでシミュレーションして通知発生予測件数を確認できます。

1. 通知設定を完了後、下部の時間範囲を選択してください。
2. [シミュレーション] ボタンをクリックしてください。
3. チャートに時間帯別データと通知発生回数が表示されます。

❗ シミュレーション時間範囲を調整して、さまざまな時間帯の通知発生パターンを確認できます。直近1~2分のデータはインデックスビルド遅延により照会されない場合があります。

4. イベントルール追加ウィンドウで設定が完了したら、[保存] ボタンをクリックしてください。

ログエクスプローラーから通知を作成

ログエクスプローラーから直接通知を作成できます。

1. ログ>ログ検索メニューに移動してください。
2. カテゴリと検索クエリを設定してください。
3. 画面右上の[イベント追加] ボタンをクリックしてください。
4. Drawerで通知設定フォームが開き、現在のカテゴリと検索クエリが自動的に入力されます。
5. 追加設定を完了し、[保存] ボタンをクリックしてください。
6. 保存完了後、[移動] ボタンをクリックすると通知一覧に移動します。

イベントの編集/削除

1. アラート通知>イベント設定の複合ログタブに移動してください。
2. ログイベントリストで編集または削除するイベントの✎アイコンをクリックしてください。
3. イベント設定画面でオプションを変更し、[保存] ボタンをクリックしてください。
 - a. 選択したイベントを削除するには、イベント設定画面の右上の[削除] ボタンをクリックしてください。

イベントの共有

メトリクスイベント設定を JSON ファイルとして保存し、他のユーザーと設定を共有したり、他のユーザーの設定を取り込んだりすることができます。

- JSON ファイル名: `event-rules-YYYYMMDD.json`

エクスポート

1. 画面右上の [JSON ] ボタンをクリックします。
2. JSON 編集画面が表示されたら、[ エクスポート] ボタンをクリックします。
 - イベントを検索した状態でエクスポート機能を使用すると、検索結果のみが JSON ファイルとしてダウンロードされます。
3. JSON ファイルがダウンロードされたら、共有したい他のユーザーに渡してください。

インポート

1. 画面右上の [] ボタンをクリックします。
2. エクスポート 機能でダウンロードした JSON ファイルを選択します。
3. JSON 編集画面が表示されたら、[一覧に追加] または [上書き] を選択します。

 同じ種類の製品間での利用を推奨します。異なる製品のプロジェクトからイベント設定を取り込むことは可能ですが、正常に動作しない場合があります。

JSON 形式で編集する

1. 画面右上の [JSON ] ボタンをクリックします。
2. 編集画面が表示されたら、JSON 形式に従って内容を修正してください。
3. 修正が完了したら、画面下部の [保存] ボタンをクリックします。

❗ 修正した内容が JSON 形式に合っていない場合、画面下部にエラーメッセージが表示され、保存できません。表示されるエラーメッセージは、形式エラーの内容によって異なる場合があります。

JSON error

データベース

ホーム > プロジェクト選択 >  **アラート通知 > イベント設定** classic

データベースの指標ごとにイベント条件を設定し、アラート通知を送信します。指標ごとにしきい値を入力することで、**危険** および **警告** レベルに分けて通知できます。

基本オプション

データベースインスタンスの主要指標を基準にイベント条件を設定できます。

- ・ **インスタンス**: イベント条件を設定するデータベースエージェント
- ・ **データベース**: モニタリング対象のデータベース
- ・ **エージェント非アクティブ警告**: 設定した時間後に応答がない場合、エージェントから警告通知を送信

表 | データベースイベント設定画面の構成

項目	説明
メトリクス	イベント条件を設定するデータベース指標
危険	Critical レベルのしきい値
警告	Warning レベルのしきい値
検知回数	指標が 5 秒ごとに収集される際、しきい値を連続して超過した回数 - 例: 3 回に設定した場合、5 秒 × 3 回連続 条件で通知
状態解消時の追加通知	イベント発生後、正常状態に復旧したことを知らせる追加通知 - DBX エージェント 1.6.30 以上対応
	イベント削除
	イベントの有効 / 無効 - 有効: 条件を満たすとイベント発生

項目	説明
	- 無効: イベントは動作せず条件も評価されません

イベント追加

選択した **メトリクス** 項目に基づき、入力した **危険** および **警告** 条件が **検知回数** 分だけ発生すると、アラート通知を送信するよう設定できます。

1. アラート通知 > イベント設定 **classic** に移動します。
2. インスタンス でアラートを設定するエージェントを選択します。
3. [+ イベント追加] ボタンをクリックします。
4. **メトリクス** から目的の指標を選択します。
5. **危険** と **警告** の比較演算子 (< , ≤ , > , ≥) を選択し、数値を入力します。
 - 設定値を超過するとアラートが発生します。
6. **検知回数** に希望する数値を入力します。
 - 5 秒間隔で設定回数連続発生すると通知されます。
 - 例: 検知回数が 3 の場合
5 秒 (1 回目) - 5 秒 (2 回目) - 5 秒 (3 回目) → 合計 15 秒 → 通知
7. **状態解消時の追加通知** を有効にすると、イベントが解消 (RECOVERED) された際に追加通知が送信されます。

❗ 状態解消時の追加通知 機能は DBX エージェント バージョン 1.6.30 以上 で対応しています。
8. 右端のトグルボタンを有効 () にするとアラート通知が送信されます。
 - a. イベントを削除する場合は、該当イベントの  をクリックしてください。

アラート複製

現在選択しているインスタンスに作成したアラートイベントを、他のインスタンスに複製できます。

1. アラート通知 > イベント設定 **classic** に移動します。
2. [アラート複製] ボタンをクリックします。
3. アラート複製 画面で複製先インスタンスを選択します。すべてのインスタンスに複製する場合は **全体選択** を選択してください。
4. [複製] ボタンをクリックしてアラートを複製します。

❗ 複製先インスタンスに既存のアラートイベントがある状態で **アラート複製** 機能を使用すると、既存のアラートイベントは削除されます。

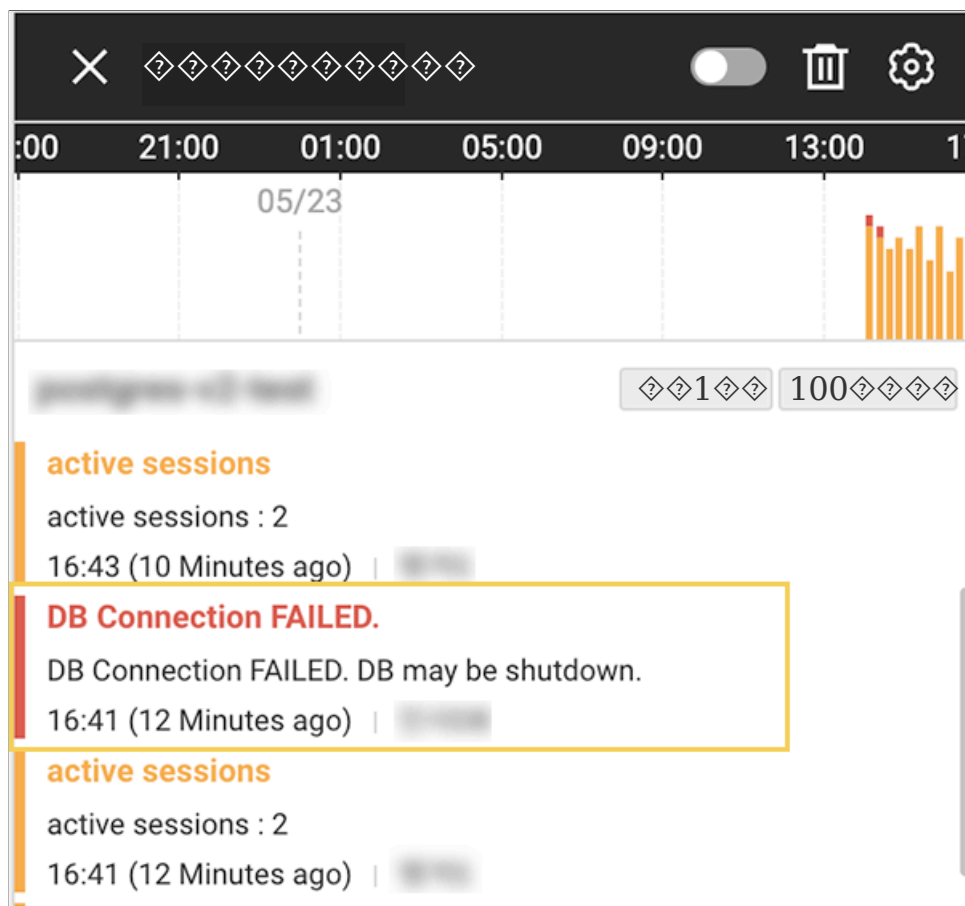
エージェント非アクティブ警告

設定した時間後に応答がない場合、エージェントから警告通知を送信します。

1. アラート通知 > イベント設定 **classic** に移動します。
2. エージェント非アクティブ警告 で希望する時間（ **30秒** , **1分** , **2分** , **3分** , **5分** , **10分** , **30分** ）を選択します。
3. 右側のトグルボタンをクリックして有効化します。
 -  **有効**: 設定した時間中にエージェントからデータを受信できない場合、**INACTIVE MONITORING AGENT** イベントが発生します。

基本アラート: Connection FAILED

イベントルールを別途設定しなくても、エージェントが DB 接続に 15 回以上失敗した場合、アラートが送信されます。DB シャットダウンやネットワーク障害、アカウントまたは権限の問題が原因となる場合があります。



通知設定

ホーム > プロジェクト選択 >  アラート通知 > 通知設定

ユーザー別アラート受信設定

 **ユーザー別アラート受信設定** における他メンバーの情報は、**メンバー管理** 権限を持つメンバーのみが確認できます。それ以外のメンバーには、必須情報がマスキングされて表示されます。

- イベント通知の一括受信設定およびアクセス設定のためのモバイル端末管理は、**アカウント情報** で行うことができます。
- メンバー権限の詳細については、[こちらのドキュメント](#) を参照してください。

通知手段の設定

メール、**SMS**、**WhatsApp**、**モバイル** による通知を提供します。

ユーザー別アラート受信設定 セクションで、受信したい通知手段のチェックボックスを選択すると警告アラートを受信できます。通知手段のチェックボックスを解除すると、その手段ではアラートは送信されません。

 プロジェクトの最高管理者を除き、すべてのユーザーは自分自身の受信設定のみ変更できます。

メール通知

メール通知は、会員登録時に入力したメールアドレス宛に送信されます。

SMS通知

SMSで通知を受信できます。

 SMS通知機能は、有料プロジェクトでのみ利用できます。

1. 画面右上の **プロフィールアイコン** をクリックし、**アカウント管理** に移動します。
2. **アカウント情報** の **ユーザー電話番号** セクションで、[一般携帯電話] ボタンをクリックします。
3. 認証番号を受信する **電話番号** を入力します。
 - SMS通知を受信できる電話番号は、**韓国の携帯電話番号のみ**対応しています。
4. [保存] ボタンをクリックします。

✔ 登録した電話番号の変更および削除

- 電話番号を変更する場合は、[変更] ボタンをクリックしてください。SMS認証が必要です。
- 電話番号を削除する場合は、[削除] ボタンをクリックしてください。削除後は、SMSによる通知を受信できなくなります。

WhatsApp通知

WhatsAppを通じて通知を受信できます。

ⓘ WhatsApp通知機能は、有料プロジェクトでのみ利用できます。

1. 画面右上の **プロフィールアイコン** をクリックし、**アカウント管理** に移動します。
2. **アカウント情報** の **ユーザー電話番号** セクションで、[WhatsApp] ボタンをクリックします。
3. 認証番号を受信する **電話番号** を入力します。
4. [認証番号送信] ボタンをクリックします。
5. WhatsAppアプリケーションに送信された6桁の認証番号を入力します。
 - 10分以内に認証番号を受信できなかった場合は、[認証番号再送信] ボタンをクリックしてください。
6. [認証] ボタンをクリックします。

モバイル通知

モバイル通知を受信するには、モバイル端末にWhaTapアプリケーションをインストールし、ログインする必要があります。

- WhaTapアプリケーションをダウンロード – [Android](#)
- WhaTapアプリケーションをダウンロード – [iOS](#)

繰り返し通知（エスカレーション）設定

アラート発生後も状況が解消されない場合、初回アラート発生時刻を基準に通知を繰り返す間隔を設定できます。この設定は **Critical** レベルのアラートにのみ適用されます。

例えば、初回アラート発生から即時（0分）、1時間後、1日後に通知を繰り返す場合は、**繰り返し通知（エスカレーション）** カラムに **0,1H,1D** と入力します。

1. **アラート通知 > 通知設定** に移動し、**ユーザー別アラート受信設定** セクションを開きます。
2. **繰り返し通知（エスカレーション）** 項目で設定します。
 - **M**: 分、**H**: 時間、**D**: 日。単位を省略した場合は分単位として扱われます。
 - 数値、または 数値+単位（**M, H, D**）の形式で入力してください。入力形式が正しくない場合はメッセージが表示されます。
3. [保存] ボタンをクリックしてください。保存しない場合、設定は反映されません。

イベント受信タグ

イベント設定時に **イベント受信タグ** を選択すると、そのタグを持つプロジェクトメンバーや 3rd-party プラグインに通知を送信できます。イベント受信タグを設定しない場合、**すべてのプロジェクトメンバー** に警告アラートが送信されます。

イベント受信タグの追加・編集・削除

 イベントに **適用中のイベント受信タグ** は削除できません。

1. **アラート通知 > 通知設定** に移動します。
2. **ユーザー別アラート受信設定** セクションのユーザー一覧で、**+ タグ追加** または **+** をクリックします。
3. **イベント受信タグ** 画面で **+ 新しいタグ作成** をクリックします。
4. **タグ作成** 画面でタグ名を入力し、色を選択して [タグ作成] ボタンをクリックします。

5. 作成されたタグは、タグ一覧の  をクリックして編集または削除できます。
6. イベント受信タグ画面の **タグ一覧** から使用したいタグを選択すると適用されます。

イベント受信タグを解除する

1. アラート通知 > 通知設定 に移動し、**ユーザー別アラート受信設定** セクションを開きます。
2. イベント受信タグを解除したいユーザーの **イベント受信タグ** 項目にある + をクリックします。
3. イベント受信タグ画面で、解除したいタグの X をクリックすると、そのタグが解除されます。

受信タグ未設定のアラートを受信する

イベント設定時にイベント受信タグを選択すると、そのタグを持つプロジェクトメンバーや 3rd-party プラグインに通知を送信できます。

- イベント受信タグが設定されていないアラートを受信するには、**受信タグ未設定のアラートを受信** オプションを有効にしてください。
- イベント受信タグが設定されているアラートのみ受信したい場合は、このオプションを無効にしてください。

❗ すべての警告アラートを受信しない場合は、このオプションを無効にし、**イベント受信タグ** が一つも選択されていない状態にしてください。

イベント受信モードの選択

ユーザー別アラート受信設定 セクションで、メンバーまたはサードパーティチャネルごとにイベント通知の受信範囲を次の3つのモードから選択できます。受信モードは、各メンバーまたはチャネル行の**イベント受信タグ**列で選択します。

Mode	Description
すべての通知を受信	イベント受信タグに関係なく、すべての警告通知を受信
選択したタグのみ通知を受信	選択したイベント受信タグを持つ警告通知のみ受信。タグを最低1つ以上選択
すべての通知を受信しない	いかなるイベント通知も受信しない。選択したタグは削除されずそのまま保存

- 選択したタグのみ通知を受信モードでは、タグ選択領域で通知を受信するタグを1つ以上選択してください。
- イベント受信タグが指定されていない警告通知まで受信するには、タグ選択領域で**未分類**仮想タグを選択してください。**未分類**タグは通常のタグと同様に選択または解除でき、タグ自体を編集・削除することはできません。

❗ すべての通知を受信しないモードでは、該当メンバーまたはチャネルがイベント通知をまったく受信しません。このとき選択したタグは削除されず保存されるため、再び**選択したタグのみ通知を受信モード**に切り替えると、以前に選択したタグ設定をそのまま使用できます。

通知受信言語の選択

ユーザーが受信するアラート通知の言語を選択できます。

1. アラート通知 > 通知設定 に移動します。
2. ユーザー別アラート受信設定 セクションのユーザー一覧で、言語を設定したいユーザーの一番左にある ► ボタンをクリックします。
3. 通知受信言語 を選択します。

曜日・時間別通知設定

ユーザーごとに、曜日および時間帯別の通知受信可否を設定できます。

1. アラート通知 > 通知設定 に移動します。
2. ユーザー別アラート受信設定 セクションのユーザー一覧で、曜日・時間別通知を設定したいユーザーの一番左にある ► ボタンをクリックします。
3. 警告アラートを受信したい曜日を選択するか、時間帯を入力します。
 - 通知手段ごとに設定できます。

サードパーティプラグイン

外部アプリケーションを通じて警告アラートを受信できます。

- 対応アプリケーション: Slack、Telegram、Teams、Teams Workflows、Jandi、AlertNow、Webhook、Webhook

JSON、Opsgenie、PagerDuty、LINE、ilert

❗ WhaTap のサポート範囲に含まれない社内メッセンジャーは、標準の **Webhook** または **Webhook JSON** を使用して連携できます。

表 | サードパーティプラグイン一覧

項目	説明
プラグイン名	プラグインの名称
認証キー	認証キー
認証値	認証値
受信レベル	警告、全体、危険のいずれか
繰り返し通知（エスカレーション）	問題が解消されない場合、初回通知時刻から再通知する間隔 - Critical レベルのアラートにのみ適用
Smart メッセージ（AI）	Telegram プラグイン追加時に AI 通知機能を提供 - 対応言語：韓国語、英語、日本語
イベント受信タグ	プロジェクト内の特定メンバーまたはチームに対する通知受信設定
削除	サードパーティプラグインの削除

サードパーティプラグインの追加

1. アラート通知 > 通知設定 に移動します。
2. サードパーティプラグイン セクションで **追加** ボタンをクリックします。
3. サードパーティプラグイン追加 画面で、使用するサービスのタブを選択します。
4. 選択したサービスの案内（[サードパーティプラグイン通知の追加方法](#)）に従って設定します。
5. 受信テスト ボタンをクリックして設定を確認します。

- すべて完了したら **追加** または **登録** ボタンをクリックします。

サードパーティプラグイン通知の追加方法

Slack

- サードパーティプラグイン追加 画面の **Slack** タブで、[**Slackに登録**] ボタンをクリックすると、Slack のチームおよびチャンネルに **WhaTapNotiBot** を登録できます。

Telegram

- Telegramで @WhaTapApmNotiBot を検索してください。

- ボットを選択し、[開始] ボタンをクリックします。
- チャット画面に `/license` コマンドと、通知を受信したいプロジェクトのアクセスキーを入力してください。



① 使用可能なコマンド一覧

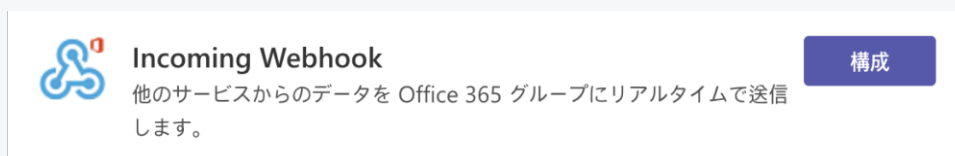
- `/help` : コマンド一覧を表示
- `/license` : 通知を受信するプロジェクトのアクセスキーを入力
- `/remove` : 通知を受信しないプロジェクトコードを入力
- `/list` : 現在通知を受信しているプロジェクトコードを表示

Teams

- 通知を受信したいチャンネルの **コネクタ** に移動します。



2. Webhook を追加し、[構成] ボタンをクリックします。



3. Webhook 名を入力し、[作成] ボタンをクリックします。



Incoming Webhook

フィードバックの送信

Incoming Webhook コネクタを利用すると、トラッキング対象の活動についての通知を外部サービスから受け取ることができます。このコネクタを使うには、別サービスで設定を作成する必要があります。そのサービスが、Office 365 コネクタ形式 対応の Web フックをサポートしている必要があります。

* が付いているフィールドは必須です

着信 Web フック (Incoming Webhook) をセットアップするには、名前を入力して [作成] を選んでください。*

この着信 Web フック (Incoming Webhook) からのデータに関連付ける画像をカスタマイズします。

イメージをアップロードします。



既定のイメージ

作成

キャンセル

4. 生成された Webhook URL をコピーします。

以下の URL をコピーしてクリップボードに保存してから、[保存] を選んでください。この URL は、グループへのデータ送信元となるサービスの画面で必要になります。

`https://whatap.webhook.office.com/webf`



URL は最新です。

完了

削除

5. WhaTap モニタリングサービスの サードパーティプラグインの追加 画面に戻り、Teams タブ下部の Channel Name と Webhook URL に値を入力します。

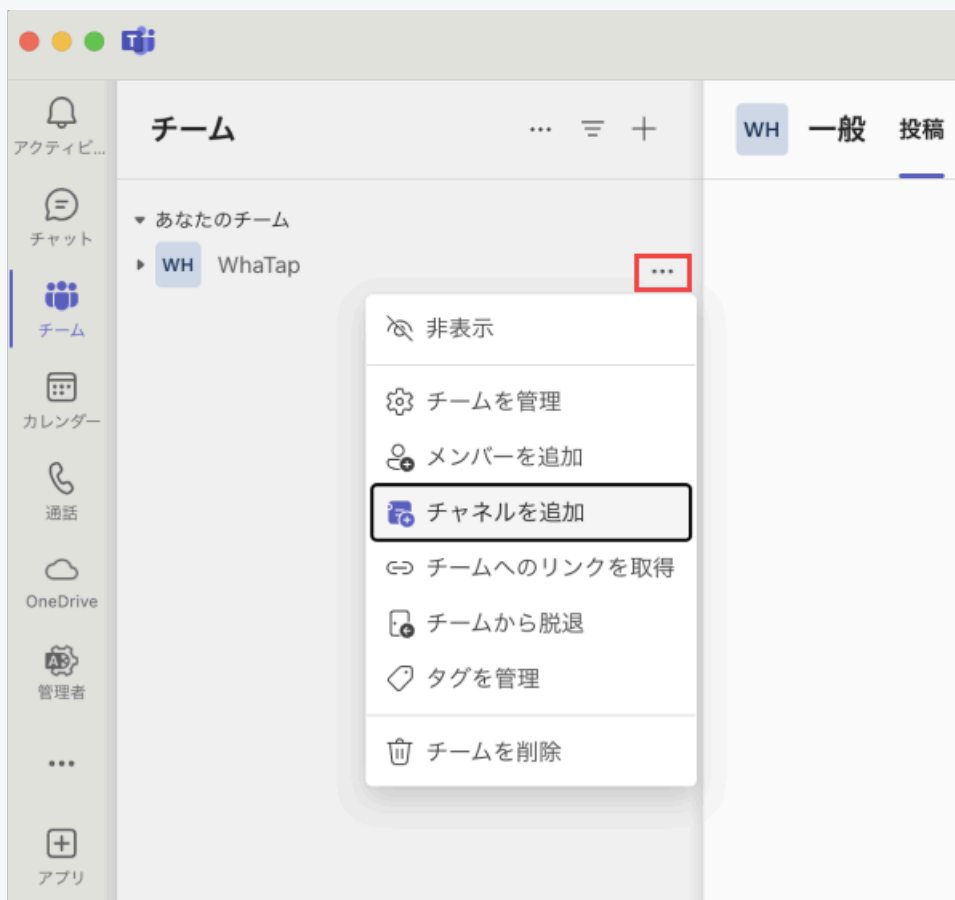
6. [登録] ボタンをクリックします。



Teams Workflows

- ① ワークフローを使用するには、**Microsoft Teams** にアクセス可能なアカウントと **ビジネスプランのサブスクリプション**が必要です。

1. チーム メニューの **その他の操作 (...)** > **チャンネルを追加** をクリックします。



2. チャンネルの作成画面で **チャンネル名** と **説明** を入力します。チャンネルの種類は **標準** を選択し、[作成] ボタンをクリックします。



チャンネルの作成

チャンネル名 *

WhaTapのイベントチャンネル

説明

他のユーザーが、適切なチャンネルを見つけられるように説明を入力します

チャンネルの種類を選択する * ⓘ

選択

☒ 標準
チームの全員がアクセスできます。

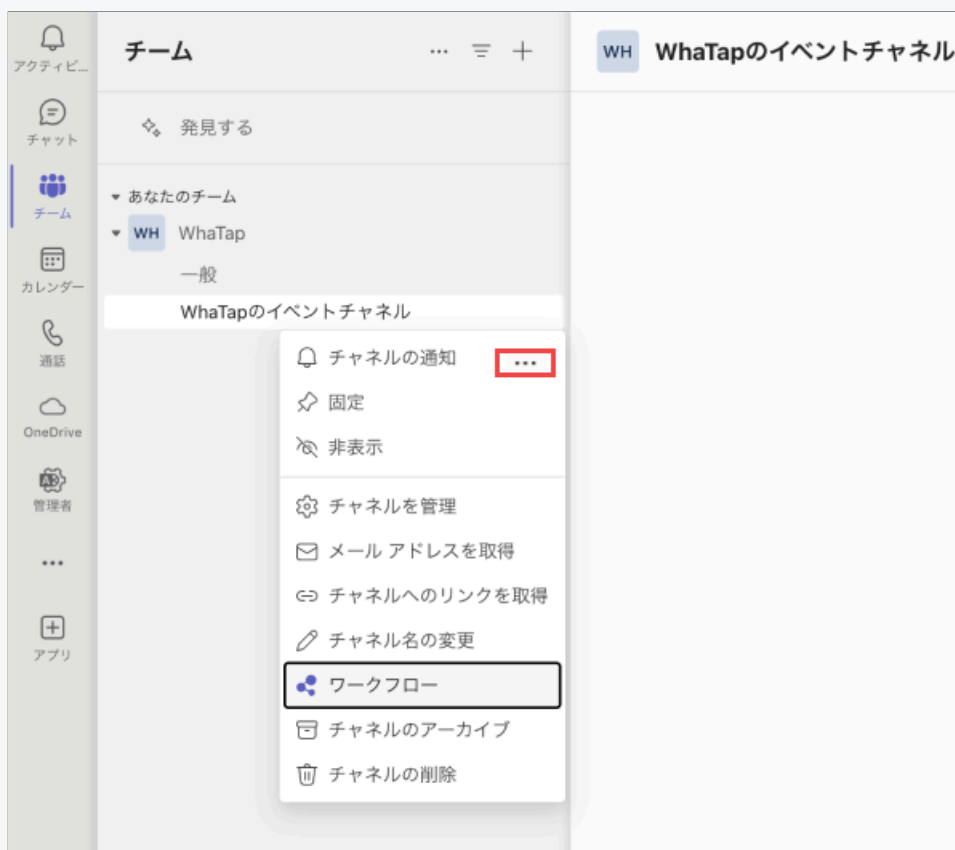
☐ 共有
組織内または組織外の人またはチームがアクセスできます。

☐ プライベート
チームの特定のユーザーがアクセスできます。

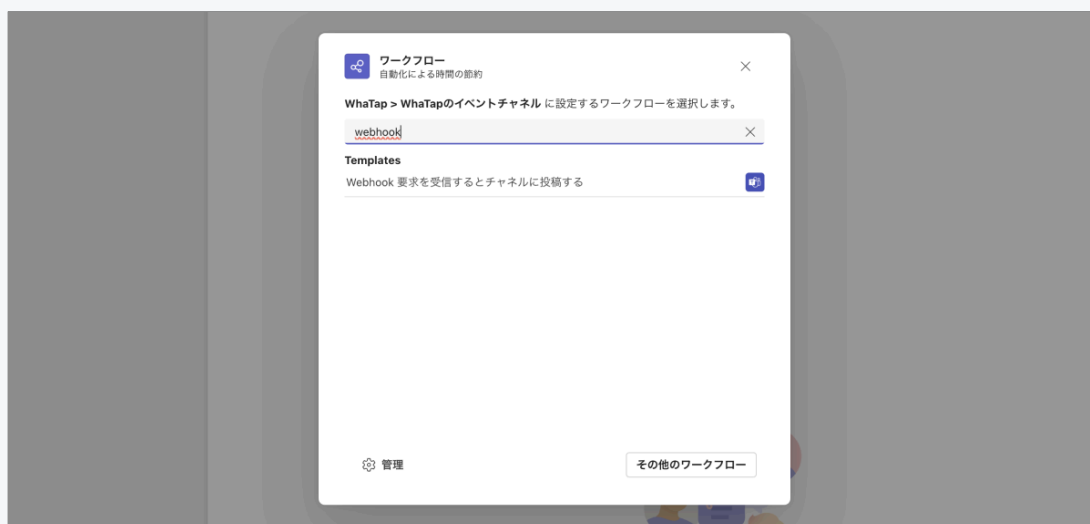
キャンセル 作成

❗ チャンネルの種類を **共有** にするとワークフローは使用できません。**プライベート** を選択した場合、ワークフローは追加できますが通知は受信できません。

- 作成したチャンネルの **その他の操作 (...)** > **ワークフロー** をクリックします。



4. ワークフロー ダイアログの一覧から Webhook 要求を受信したらチャンネルに投稿 を選択し、[ワークフローを追加] ボタンをクリックします。



5. 名前と接続の状態を確認し、[次へ] ボタンをクリックします。
6. 詳細を確認後、[ワークフローを追加] ボタンをクリックします。
7. ワークフローが追加されました。生成された URL をコピーします。



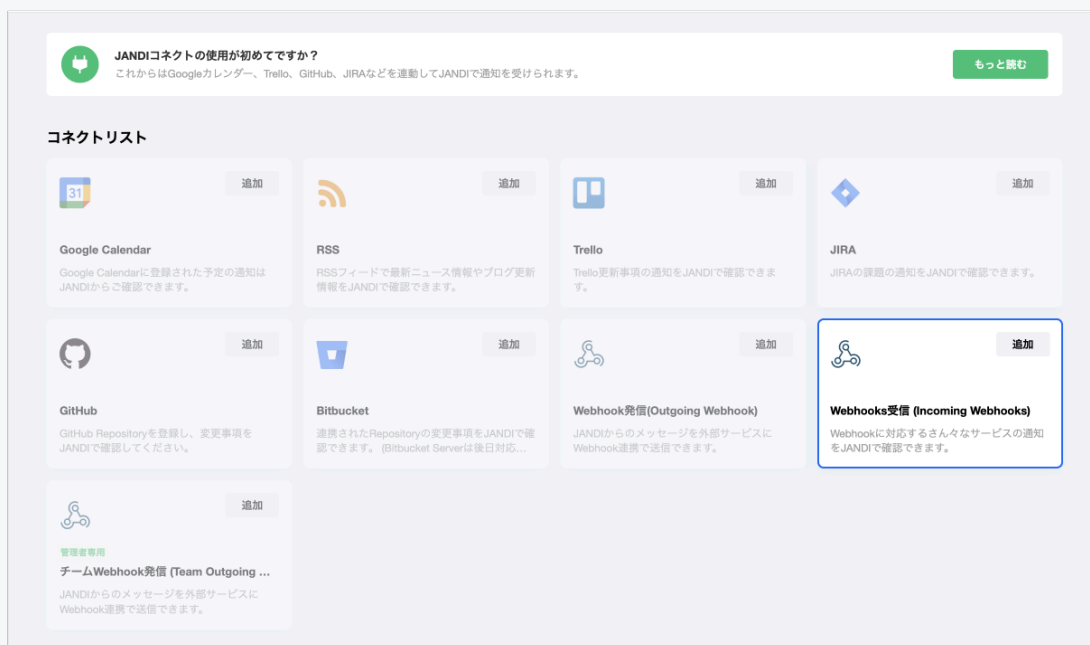
8. WhaTap モニタリングサービスの **サードパーティプラグインの追加** 画面に戻り、Teams **Workflows** タブ下部の **Channel Name** と **Webhook URL** に値を入力します。
9. [登録] ボタンをクリックします。

Jandi





1. 通知を受信するトピックまたはチャットで、**Jandi Connect 設定** に移動します。
2. **コネクト一覧** から **Webhook 受信 (Incoming Webhook)** を選択し、[連携する] ボタンをクリックします。



3. [**Webhook 受信を追加**] ボタンをクリックします。
4. **Webhook URL 登録** 項目に表示される Webhook URL をコピーします。



5. WhaTap モニタリングサービスの **サードパーティプラグインの追加** 画面に戻り、**Jandi** タブ下部の **Channel Name** と **Webhook URL** に値を入力します。
6. [**登録**] ボタンをクリックします。



1. AlertNow サービスにアクセスし、設定 > Integration メニューに移動します。



2. Integration 画面で [Integration 作成] ボタンをクリックします。
3. Integration タイプの中から **WhaTap** を選択します。
4. Integration 作成 画面で、名前 および 命名規則 を設定します。

5. [確認] ボタンをクリックすると、Integration が作成されます。
6. 作成された Integration の一覧を確認し、該当する Integration を選択します。



インテグレーション

インテグレーションの作成 検索語を入力 最後のアップデート: 2024-02-21 11:25:10

統合名	統合タイプ	統合キー	サービス名
ワットテスト	ワタップ		ワットテスト

7. 詳細画面で、URL 右側のコピーアイコンをクリックして URL をコピーします。

← **インテグレーション**

ワットテスト

基本情報

統合タイプ	ワタップ
統合キー	fda527a5140
URL	https://alertnowitgr.opsnow.com/integration/whatap/v1/
コンストラクタ	リサ

クリップボードにコピーする

8. Whatap モニタリングサービスの **サードパーティプラグインの追加** 画面に戻り、AlertNow タブ下部の **Channel Name** と **Webhook URL** に値を入力します。
9. [登録] ボタンをクリックします。

Webhook



1. 通知を受信する Webhook を登録します。登録した URL に対して、通知の **title** と **text** が POST メソッドで送信されます。

```
{
  "title": "Alert Title",
  "text": "Sample warning message"
}
```

2. **Webhook Name** と **Webhook URL** の項目を入力します。
3. [登録] ボタンをクリックします。

Webhook JSON



1. 通知を受信する Webhook を登録します。登録した URL に対して、通知データが **JSON 形式で POST** メソッドにより送信されます。JSON に含めることができる置換変数については、**使用可能な変数**を参照してください。

- `${something}` は WhaTap が提供する置換値を表します。対応する全一覧と意味については、**使用可能な変数**を参照してください。
- イベント状態（`status`）は、**イベント発生**（`on`）と**イベント解消**（`off`）に区別されます。
- `level` は `None`、`Info`、`Warning`、`Critical` に区別されます。

```
{
  "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}"
}
```

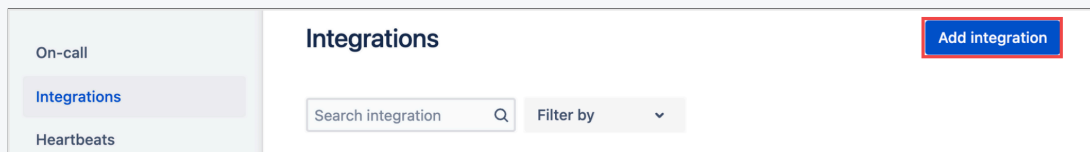
2. **Webhook Name** と **Webhook URL** の項目を入力してください。
3. ヘッダー値を追加するには、**ユーザーヘッダー値の追加** の + 追加 をクリックし、key と value を入力してください。
4. [登録] ボタンをクリックしてください。

Opsgenie

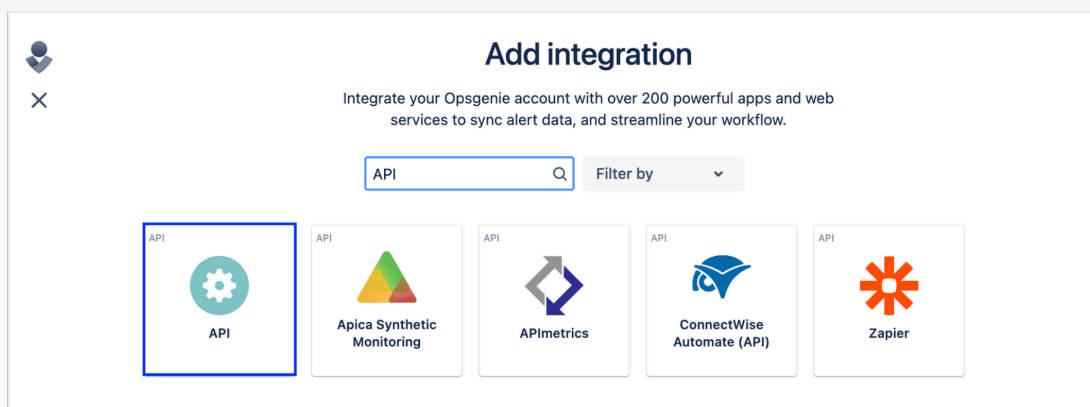




1. **Teams** メニューで、アラートを受信するチームを選択してください。
2. **Integration** メニューに移動し、[Add Integration] ボタンをクリックしてください。



3. インテグレーション一覧で **API** を検索し、選択してください。



4. API 名を設定し、チームを割り当て、各権限を設定した後、[Save Integration] ボタンをクリックしてください。



Forwarding rules
Your on-call schedule

ON-CALL
Global policies
All schedules
All escalations

CONNECTED APPS
Atlassian products

INTEGRATIONS
Integrations
Heartbeats
Slack
GitLab

Settings

Name:

Assigned to Team:

API Key:

Read Access: ☒

Create and Update Access: ☒

Delete Access: ☒

Restrict Configuration Access: ☐

Enabled: ☒

Suppress Notifications: ☐

[Save Integration](#) [Cancel](#)

5. Whatap モニタリングサービスの **サードパーティプラグインの追加** 画面に戻り、Opsgenie タブ下部の **Name** と **API Key** を入力してください。
6. [登録] ボタンをクリックしてください。



PagerDuty



1. [PagerDuty](#) アカウントを登録し、ログインしてください。
2. PagerDuty で **Services > Service Directory** メニューに移動し、[+ New Service] ボタンをクリックしてください。
3. サービス情報 (Name) を入力し、[Next] ボタンをクリックしてください。
4. **Escalation Policy** を設定し、[Next] ボタンをクリックしてください。



PagerDuty

5. **Alert Grouping** と **Transient Alerts** を設定し、[Next] ボタンをクリックしてください。

PagerDuty

6. **Integrations** で **Event API V2** を選択し、[Create Service] ボタンをクリックしてください。

PagerDuty

7. 作成されたサービス画面で **Integration Name** と **Integration Key** の値をコピーしてください。

PagerDuty

8. WhaTap モニタリングサービスの **サードパーティプラグインの追加** 画面に戻り、**Integration Name** と **Integration Key** を入力してください。
9. [登録] ボタンをクリックしてください。

❗ メッセージを受信できない、またはプラグインが登録されない場合は、**Integration Name** または **Integration Key** の値が正しいか確認してください。

LINE



1. LINE アプリをインストールしてください。
2. 以下の QR コードをスキャンして、**WhaTapLabs** 公式チャンネルを友だち追加してください。

LINE 3rd パーティプラグイン追加

3. 次のコマンドを使用して、プロジェクトを LINE と連携し、通知を設定してください。

コマンド

```
/help # 使用可能なコマンド一覧を表示
/list # 連携されているプロジェクト一覧を表示
/license [アクセスキー] # 新しいプロジェクト連携を追加
```



```
/remove [プロジェクトコード] # プロジェクト連携を解除
/language [ko|ja|en] # 通知言語を設定（韓国語／日本語／英語）
```

- プロジェクトのアクセスキーは **管理 > プロジェクト管理** で確認できます。
- LINE 内で `/language` コマンドを使用した場合、その設定は **該当する LINE チャットルーム内のみ** に適用されます。他の 3rd パーティプラグインやプロジェクトの通知言語は変更されません。

使用例

```
/license XXXXXXXXXX # プロジェクトを連携
/language ko # 言語を韓国語に設定
/list # 連携済みプロジェクトを確認
```



- プロジェクトアクセスキー:** WhaTap プロジェクト内の **管理 > プロジェクト管理** で確認できます。
- 言語設定:** `/language` コマンドは LINE チャットルーム内でのみ有効で、他の通知チャンネルの言語は変更されません。

ilert



- ilert コンソールの **Alert sources** 画面右上にある **[+ Create alert source]** をクリックします。
- Integration type** ステップ下部の **Filter integrations** 検索欄に **WhaTap** を入力して検索します。
- 検索結果から **WhaTap** タイルを選択し、**[Next]** ボタンをクリックします。
- アラートソース名を入力し、チームを指定した後、**[Next]** ボタンをクリックします。
 - エスカレーションポリシーは、新規作成または既存のものを選択できます。
- アラートのグループ化方式を選択し、**[Continue setup]** ボタンをクリックします。
 - 最初に選択した **Do not group alerts** は後から変更可能です。
- 次の画面で **アラートテンプレートや優先度などの追加設定** を行うことができます。ここでは [



Finish setup] ボタンをクリックして設定を完了します。

7. 設定が完了すると、完了画面で **API キー** または **Webhook URL** が生成されます。
8. **Webhook JSON** タブで生成された **Webhook URL** をコピーし、下部の **Webhook URL** 入力欄に貼り付けます。
9. [登録] ボタンをクリックします。

大量アラート発生の防止

短時間に大量のアラートが発生した場合、指定した時間だけアラート通知を一時的に停止します。たとえば、5 分間に 20 件のイベントが発生すると、その後 5 分間は警告アラートが送信されません。設定した **停止時間** が経過すると、大量アラート防止機能は自動的に解除されます。

1. 次の画面に移動してください。 **アラート通知 > 通知設定 > 大量アラート通知発生防止** セクション
2. トグルボタンをクリックして **有効化** または **無効化** します。
 -  **有効**: 大量アラートを検知した場合、一時的に通知を停止
 -  **無効**: アラート通知を再開
3. **検知時間**, **検知回数**, **停止時間** を設定してください。
 - **検知時間** 内に **検知回数** 以上のイベントが発生すると、**停止時間** の間、警告アラートが停止されます。



大量アラートによってブロックされた場合、ブロック案内ボックスが表示されます。大量アラート防止機能をすぐに解除するには、ボックス内の **大量アラート発生中止** ボタンをクリックしてください。

イベント通知フォーマット

共通アラートテンプレート

共通テンプレートは、ほとんどのケースで適用できます。さまざまな環境において、同一のフォーマットでアラートを受信できます。

- プロダクトタイプ: アプリケーション、データベース、Kubernetes
- イベントタイプ: アプリケーションアラート、データベースアラート、メトリクスアラート
- イベントチャネル: SMS、Mobile、3rd-party プラグイン、プラグイン

イベントタイトル

Event title format

```
[Level][Platform][ProjectName][ApplicationName][EventTitle]
```

Event title example

```
[Info][JAVA][アプリケーションプロジェクト][TC-0-1-8081][CRITICAL_HIGH_MEMORY]
```

❗ ApplicationName が設定されていない場合、イベントタイトルから省略されます。

- Platform は以下のいずれかで表示されます。
 - 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

イベントメッセージ

イベントメッセージに含めることができる情報です。 `optional` が `false` の場合、常にメッセージに含まれます。
`optional` が `true` の場合、該当データが確認できるときのみ表示されます。

イベントメッセージ例

```
Project Name : アプリケーションプロジェクト  
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

表 | イベントメッセージ構成要素

En	Ko	対応するアラートタイプ	説明
Project Name	プロジェクト名	全て	-
Project Code	プロジェクトコード	全て	-
Agent Name	エージェント名	全て (optional)	oname
Message	イベントメッセージ	全て	-
Alert Type	イベント種別	全て	下記の AlertType 表を参照
Event Time	イベント発生時間	全て	2022-04-13 10:40:49 +0900 の +0900 は GMT オフセット
Event Off Time	イベント解除時間	全て (optional)	2022-04-13 10:40:49 +0900 の +0900 は GMT オフセット
Metric Name	メトリクス名	全て (optional)	イベント条件判定に使用されるメトリクス名
Metric Value	メトリクス値	全て (optional)	メトリクス値が閾値を超えた場合に含まれる
Metric Threshold	メトリクス閾値	全て (optional)	メトリクス条件の閾値

En	Ko	対応するアラートタイプ	説明
Stateful	復旧イベント通知	全て (optional)	復旧通知が有効な場合 <code>true</code> 、それ以外は <code>false</code>
Event Rule	イベント発生条件	メトリクスアラート	-
Target Filter	イベント対象選択	メトリクスアラート	特定対象から収集されたメトリクスのみ評価
Repeat Count	イベント繰り返し回数	メトリクスアラート	繰り返し時間内に指定回数条件を満たすと発生
Repeat Duration	イベント繰り返し時間	メトリクスアラート	繰り返し回数を評価する時間
Receiver	受信者	メトリクスアラート	-
Query	MXQL クエリ	複合メトリクスアラート	-
Rule	イベント発生条件	複合メトリクスアラート	-
Query Period	クエリ期間	複合メトリクスアラート	-
Query Interval	クエリ間隔	複合メトリクスアラート	-
Silent Time	サイレント時間	複合メトリクスアラート	-
Query	URL	Exception アラート	例外を発生させたリクエストの URL
TXID	トランザクション ID	Exception アラート	-

En	Ko	対応するアラートタイプ	説明
Class	エラークラス名	Exception アラート	-
Log Message	ログメッセージ	サーバーファイル ログアラート	-
Log File	ログファイルパス	サーバーファイル ログアラート	-
IP	IP	サーバーアラート全 体	-
CPU	CPU	サーバーアラート全 体	イベント発生時のスナップショット
CPU_load1	CPU_load1	サーバーアラート全 体	イベント発生時のスナップショット
CPU_loadPerCore	CPU_loadPerCore	サーバーアラート全 体	イベント発生時のスナップショット
Memory	メモリ	サーバーアラート全 体	イベント発生時のスナップショット
Swap	スワップ	サーバーアラート全 体	イベント発生時のスナップショット
Disk Name Used Percent Free Size IO Percent	ディスクパフォー マンス	サーバーアラート全 体	イベント発生時のスナップショット
Name Bps Pps	トラフィックパフ ォーマンス	サーバーアラート全 体	イベント発生時のスナップショット
Message Time Name	処理履歴メッセー ジ	サーバーアラート全 体	-

❗ このセクションでは、該当するイベントで提供可能な情報を最大限に表示します。

- **AlertType** は以下のいずれかの値で表示されます。

AlertType	説明
APPLICATION_CPU	アプリケーション CPU アラート
APPLICATION_MEMORY	アプリケーション メモリアラート
APPLICATION_DISK	アプリケーション ディスクアラート
APPLICATION_ACTIVE_TRANSACTION	アプリケーション アクティブTXアラート
APPLICATION_ERROR_TRANSACTION	アプリケーション エラートランザクションアラート
APPLICATION_SLOW_TRANSACTION	アプリケーション トランザクション応答時間アラート
METRICS	メトリクスアラート
COMPOSITE_METRICS	複合メトリクスアラート
ANOMALY	異常値アラート
LOG_REALTIME	リアルタイムログアラート
COMPOSITE_LOG	複合ログアラート
SERVER_REBOOT	サーバー 再起動アラート
SERVER_NO_DATA	サーバー データ未受信アラート
SERVER_PORT	サーバー ポートアラート
SERVER_NETWORK_IOPS	サーバー ネットワーク IOPS アラート
SERVER_NETWORK_BPS	サーバー ネットワーク BPS アラート



AlertType	説明
SERVER_DISK_IO	サーバー ディスク I/O アラート
SERVER_DISK_QUOTA	サーバー ディスク使用量アラート
SERVER_DISK_INODE	サーバー inode アラート
SERVER_CPU	サーバー CPU アラート
SERVER_MEMORY	サーバー メモリアラート
SERVER_CPU_STEAL	サーバー CPU steal アラート
SERVER_MEMORY_SWAP	サーバー スワップメモリアラート
SERVER_LOG_FILE	サーバー ログファイルアラート
SERVER_WINDOW_EVENT	サーバー Windows イベントアラート
SERVER_OFF	サーバー アラート OFF 通知
SERVER_ACKNOWLEDGE	サーバー 処理確認通知
SERVER_PROCESS_COUNT	サーバー プロセス数アラート
SERVER_PROCESS_CPU	サーバー プロセス CPU アラート
SERVER_PROCESS_MEMORY	サーバー プロセス メモリアラート
SERVER_PROCESS_OFF	サーバー プロセスアラート OFF 通知
AGENT_ACTIVE	エージェント 有効化アラート
AGENT_INACTIVE	エージェント 無効化アラート



AlertType	説明
AGENT_REACTIVATED	エージェント 再有効化アラート
URL	URL アラート
TOO_MANY_EVENT	イベント多発アラート
CLOUD_WATCH	CloudWatch アラート
EXCEPTION	例外アラート

イベント履歴

ホーム > プロジェクト選択 > 🔍 アラート通知 > イベント履歴

イベント履歴では、プロジェクトで発生したさまざまなイベントを確認し、詳細情報を分析できます。Elasticsearch DSL クエリをサポートしており、指定した条件に一致するイベントを迅速かつ正確に検索できます。

イベント履歴の確認

検索結果でイベントタイトルをクリックすると、そのイベントが発生した時間帯の詳細データを分析できる画面に移動します。これにより、イベント発生前後の状況を総合的に把握できます。

1. アラート通知 > イベント履歴 に移動してください。
2. イベント履歴 画面の 時間選択 で、イベント履歴を確認する期間を設定してください。
3. フィルター入力欄にクエリを入力し、[虫眼鏡] ボタンをクリックしてください。
 - クエリを入力しない場合、すべてのイベントが表示されます。

🔧 カラム設定

表示するカラムや非表示にするカラムを設定できます。

1. イベント履歴 画面の左上にある [🔧 カラム設定] ボタンをクリックしてください。
2. カラム設定 画面で、表示または非表示にするカラムを選択してください。
 - 表示項目 でカラムの順序を変更できます。
 - カラム名検索 欄で目的のカラムを検索できます。

カラム項目	説明
イベント名	イベントのタイトル
メッセージ	イベントのメッセージまたはスナップショット情報

カラム項目	説明
イベント状態	イベントの現在の状態
エージェント名	イベントが発生した対象エージェントの名前 (Oname) - 特定のエージェントに紐づかないイベントの場合は空欄で表示されます
エージェントグループ	イベントが発生した対象エージェントの種類 (OkindName) - 特定のエージェントに紐づかないイベントの場合は空欄で表示されます
エージェントノード	イベントが発生した対象エージェントのサーバー (OnodeName) - 特定のエージェントに紐づかないイベントの場合は空欄で表示されます
イベント発生時刻	イベントが発生した時刻
イベント解消時刻	イベントが解消された時刻 - 解消状態を持たないイベントの場合は - と表示されます

3. 設定が完了したら、[確認] ボタンをクリックしてカラム設定を反映してください。

↓ CSV ダウンロード

検索結果を CSV ファイル形式でダウンロードできます。

1. アラート通知 > イベント履歴 で [CSV] ボタンをクリックしてください。
2. 現在のフィルター条件に一致するイベント一覧が CSV ファイルとして保存されます。

進行中のイベントのみ表示

現在進行中の状態のイベントのみを表示できます。

1. アラート通知 > イベント履歴 で [進行中のイベントのみ表示] ボタンをクリックしてください。
2. フィルタークエリに次の条件が自動的に追加され、検索が実行されます。


```
Stateful: true and Status : "ON"
```

イベント検索

- **正確な検索:** 必要なイベントだけを取得できるよう、条件を正確に指定
- **複合条件:** 複数のフィールドを同時に検索し、複雑な条件も一度に処理
- **柔軟なパターン:** ワイルドカードや部分一致など、さまざまな検索方式をサポート
- **高速パフォーマンス:** インデックス化されたデータを活用した高速検索

検索可能なフィールド

フィールド名および値は大文字と小文字を区別しません。

- **フィールド名タグ:** フィールドがサポートされるイベントタイプ
- **フィールドタイプ:** フィールドのデータ型
 -  #: Number
 -  T: String
 -  C: Boolean
 -  D: Date

イベント情報フィールド

フィールド名	タイプ	説明	例
Title	String	イベントタイトル	"Database Connection Error"
OffTitle	String	イベント解消タイトル	"RECOVERED: Database Connection Error"
Message	String	イベントメッセージ	"Connection timeout occurred"
OffMessage	String	イベント解消メッセージ	"RECOVERED: Connection timeout occurred"

フィールド名	タイプ	説明	例
Level	String	現在のイベントレベル	Critical, Warning, Info
OriginLevel	String	元のイベントレベル	Critical, Warning, Info
Status	String	イベント状態	ON, OFF, CANCEL, ACKNOWLEDGE, MAINTENANCE, DISABLED
ActCount	Number	発生回数	2
MetricName	String	指標名	"memory"
MetricValue	String	指標値	"85.5"
OffValue	String	解消時の値	"72.8"
MetricThreshold	String	閾値	"80"
alertType	String	イベント種別	"BASIC", "METRICS", "TRANSACTION" など
alertId	String	イベントルールの一意識別子	"zf3uoj0fv4v7"

イベントタイプ別対応フィールド

ⓘ 特定のイベントタイプでのみサポートされるフィールドです。該当しないイベントタイプの場合、値が存在しないことがあります。

フィールド名	タイプ	イベント種別	説明	例
eventRule	String	基本、メト	イベント発生	"memory ≥ 80"

フィールド名	タイプ	イベント種別	説明	例
		リクス	ルール	
field	String	リアルタイムログ	ログ検索キー	"content"
keyword	String	リアルタイムログ	ログ検索値	"Error"
logCategory	String	リアルタイムログ	ログカテゴリ	"AppLog"
logContent	String	リアルタイムログ	ログ内容	"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."

対象フィールド

フィールド名	タイプ	説明	例
Oid	Number	エージェントの一意識別子	-98765432
Oname	String	エージェント名	"web-server-01"
Okind	Number	エージェント種別の一意識別子	867318026

フィールド名	タイプ	説明	例
OkindName	String	エージェント種別名	"web-server"
Onode	Number	エージェントノードの一意識別子	334634079
OnodeName	String	エージェントノード名	"production-node-1"

イベント固有値フィールド

フィールド名	タイプ	説明	例
Id	Number	イベントの一意識別子	5768121
UUID	String	イベント固有識別子	"550e8400-e29b-41d4-a716"

状態／フラグフィールド

フィールド名	タイプ	説明	例
Stateful	Boolean	状態ベースイベントかどうか	true, false
Disabled	Boolean	無効化されたイベントかどうか	true, false
Escalation	Boolean	エスカレーション有無	true, false
SystemEvent	Boolean	システムイベントかどうか	true, false

検索タイプ別クエリ構文

❗ 検索の基本構文については、[WhaTap ログ検索クエリ構文](#) ドキュメントを参照してください。

1. キーワード検索

「Database Connection」という文字列を含むイベント

```
title: "Database Connection"
```

2. 複数値検索

指定した複数の値のうち、いずれかを含むイベントを検索します。

```
title: Database Connection
```

現在のイベントレベルが Info または Warning のイベント

```
level: info warning
```

3. パターン検索

ワイルドカードを使用してパターン検索ができます。

名前パターンで特定のエージェントで発生したイベントを検索

```
oname: web-*-prod
```

タイトルが Connection で終わるイベントを検索

```
title: *Connection
```

4. 複合条件検索

複数の条件を組み合わせて、より正確に検索できます。

レベルが Critical で、タイトルに Database を含むイベント

```
level: critical and title: Database
```

特定のエージェントにおける Warning または Critical のイベント

```
oname: web-server-01 and level: Warning Critical
```

5. OR 条件検索

複数の条件のうち、**いずれか一つでも満たす**イベントを検索します。

タイトルまたはメッセージに特定のキーワードを含むイベント

```
title: Connection or message: Connection
```

6. 除外条件

特定の条件を除外して検索できます。

Info レベルを除いたすべてのイベント

```
not level: info
```

リアルタイムアラート

リアルタイムアラート機能は、イベント設定で設定したイベントが発生した際に、ポップアップ形式で表示されます。イベント発生と同時に通知を受け取ることで、問題を迅速に把握し、対応することができます。また、すでに発生した通知や見逃した通知の履歴を確認することで、リアルタイム対応が難しかった状況を把握し、解決に役立てることができます。

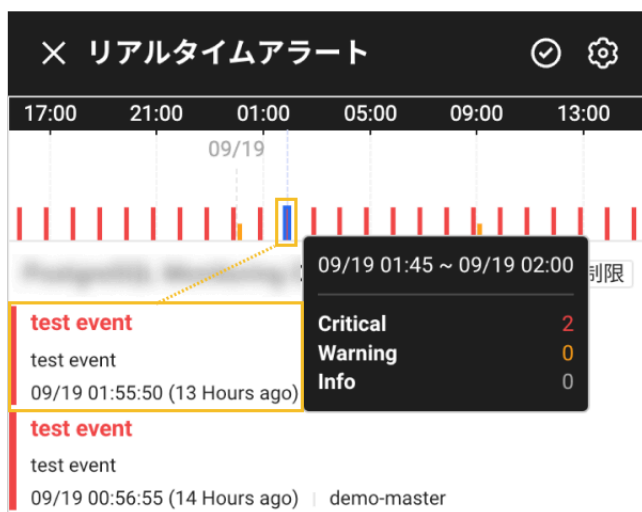
リアルタイムアラートを確認する

リアルタイムアラートは、画面右上にポップアップとして表示されます。通知が発生した時刻、タイトル、メッセージを確認できます。

- 個別の通知を閉じるには、メッセージ右上の ✕ ボタンをクリックしてください。
- 画面に表示されているすべてのポップアップ通知を閉じるには、**すべての通知を閉じる** をクリックしてください。

通知履歴を確認する

1. プロジェクトを選択して表示される画面の最上部にある 🔔 をクリックしてください。
2. リアルタイムアラート ウィンドウが表示されます。
 - 現在のプロジェクトについて、直近24時間以内の通知を最大100件まで確認できます。
 - 個別のイベント項目を選択すると、関連するメニューに移動します。
 - リアルタイムアラート ウィンドウ内の時間別チャートで任意の時間帯を選択すると、該当するイベント項目に移動します。



- すべての通知履歴を確認するには、**全体の通知履歴** ボタンをクリックしてください。**アラート通知 > イベント履歴** メニューに移動します。

❗ リアルタイムアラート の一覧が空の場合、**全体の通知履歴** ボタンが表示されます。受信タグ未設定の通知を受け取る オプションを無効にしてください。

イベントを一括で解消する

リアルタイムアラート の一覧から、進行中のすべてのイベントを一括で解消できます。

イベント解消が完了するまでに、約30秒～1分程度かかる場合があります。

- イベント一括処理** 機能を使用すると、現在進行中のイベントに対する解消通知が大量に発生する可能性があります。
- イベント一括処理** 機能を使用した後でも、イベントルールが有効であり、データが引き続きしきい値を超えている場合、イベントは再度発生することがあります。
- イベント解消通知は、**アラート通知 > イベント設定** メニューでイベントを編集し、**イベントが解消されたときに追加通知** または **解消通知** オプションを無効にすることで停止できます。

使用方法:

1. プロジェクトを選択して表示される画面上部の  をクリックします。
2. リアルタイムアラート ウィンドウが表示されたら、右上の  をクリックします。
3. リアルタイムアラートリストの進行中イベントを一括で解消します。 というメッセージが表示されたら、解消 ボタンをクリックします。

リアルタイムアラートを設定する

1. プロジェクトを選択して表示される画面上部の  をクリックします。
2. リアルタイムアラート ウィンドウが表示されたら、右上の  をクリックします。
3. リアルタイムアラート設定画面で各項目を設定します。
 - ポップアップ通知: リアルタイムアラート ウィンドウが閉じている状態でも、ポップアップ形式で通知を表示します。
 - 解消済みイベントを表示: リアルタイムアラート 一覧に解消済みイベントも含めて表示します。
 - 通知音の試聴: ブラウザで再生される通知音を事前に確認できます。Critical または Warning の横にある  をクリックしてください。音が再生されない場合は、ブラウザの設定をご確認ください。

システムイベント

WhaTap モニタリングは、プラットフォーム全体で自動的に検知・生成される共通イベントを提供します。本ドキュメントでは、システムによって発生するイベントの種類、発生条件、および対応方法について説明します。

EVENT_BLOCKED_BY_SYSTEM

プロジェクト内で短時間に過剰な数のイベントが発生した場合、WhaTap システムがこれを検知し、イベントの発生を一時的にブロックします。イベントがブロックされると、システムは次のメッセージをユーザーに通知します。このイベントが発生した場合は、イベント設定内容を確認し、必要に応じて調整してください。

 **Critical**



イベントメッセージ

Excessive events detected, event blocked for stability. name: server agent

プロジェクト番号	8
プロジェクト名	new server
対象	server agent
イベント名	EVENT_BLOCKED_BY_SYSTEM
イベント日時	(GMT+0900) 2024-11-18 18:03:18

WhaTapにログイン

new serverプロジェクトのserver agentから過度なイベントが検知されたため、イベント発生が遮断されます。server agentから一定期間の間にイベントが発生しなくなると、遮断が自動で解除されます。

ブロック基準

イベントは次の基準に基づいてブロックされます。

1. エージェント単位のブロック

1つのエージェントで **1分以内に10件** のイベントが発生した場合、そのエージェントから発生する以降のイベントはブロックされます。

2. 複数エージェントによる全体ブロック

条件 **1** によりイベントがブロックされたエージェントが **1分以内に3件以上** 発生した場合、プロジェクト全体のイベントがブロックされます。

3. プロジェクト単位のブロック

1つのプロジェクトで **1分以内に100件** のイベントが発生した場合、プロジェクト全体のイベントがブロックされます。

ブロック解除方法

システムによってイベントがブロックされた場合、対象のエージェントまたはプロジェクトで **1分間** イベントが発生しない状態を維持する必要があります。

- **アラート通知 > イベント設定** メニューから設定されているすべてのイベントを無効化してください。**イベント設定** を見直し、過剰なイベントが発生しないように閾値を調整する必要があります。
- イベントを設定する前に、**シミュレーション** 機能を活用してイベント発生回数を事前に予測できます。

連携プロジェクト管理

ホーム画面 > プロジェクト選択 > 管理 > 連携プロジェクト管理

データベースモニタリング(DPM) プロジェクトとアプリケーションのモニタリング(APM) プロジェクトを生成した場合、2つのプロジェクトを連携してDPMで収集したモニタリングデータをAPMプロジェクトで確認できます。

データベースで有効になったセッション(アクティブセッション)と連動したアプリケーションのアクティブTXにより、そのトランザクションの詳細情報をリアルタイムで確認できます。このような統合モニタリングにより、アプリケーションとデータベース間の相互作用を深層分析し、リアルタイムで性能ボトルネック現象を診断して解決できます。

連携プロジェクト管理

連携プロジェクトの登録

連携するプロジェクトを選択してから、「追加」ボタンをクリックします。(サポートタイプ: Server, MySQL, PostgreSQL, Oracle, Oracle Pro, Redis, MSSQL, Altibase, MongoDB, Cubrid, Java, Kubernetes, Namespace)

追加

連携プロジェクトのリスト

プロジェクトコード	プロジェクト名	グループ	タイプ	
2994	petclinic-Postgresql-apm		Java	

連携プロジェクト管理機能を使用すると、製品または機器中心のモニタリングを超えて、複数のプロジェクト間のモニタリングデータを統合して確認できます。これは、アプリケーション、データベースなど、様々なシステムコンポーネントがどのように相互作用するかを分析するのに役立ちます。特に、性能低下が発生した時に装備の問題なのか個別アプリケーションの問題なのか迅速に把握でき、問題解決時間を大幅に短縮できます。連携プロジェクト管理これにより、ユーザーシステム単位で性能データを総合的に分析し、より可視的なインサイトを確保できます。

詳細については、次の動画を通じて確認できます。

❗ 開始前の参考事項

- 現在はJavaプラットフォームのみサポートします。今後のアップデートでサポートプラットフォームを拡張します。
- 連携DBセッション機能の場合、Javaエージェント2.2.33バージョン以降が必要です。
- アプリケーションプロジェクトでデータベースプロジェクトを連携する方法についての詳細は、[次の文書](#)を参照してください。

連携プロジェクトを追加する

1. 管理 > 連携プロジェクト管理メニューに移動します。
2. 連携プロジェクトの登録セクションでプロジェクト選択入力欄を選択します。

連携プロジェクトの登録

連携するプロジェクトを選択してから、「追加」ボタンをクリックします。(サポートタイプ: Server, MySQL, PostgreSQL, Oracle, Oracle Pro, Redis, MSSQL, Alti)

プロジェクト選択

☐ 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

追加

3. 選択できるプロジェクトの一覧が表示されたら、連携するプロジェクトを選択します。文字列を入力して、一致するプロジェクトを検索もできます。
4. プロジェクトを1つ以上選択した後に有効になった追加ボタンを選択します。

連携プロジェクトのリストに選択したプロジェクトが追加されます。

❗ 連携プロジェクトを追加するためには、プロジェクト修正権限が必要です。修正権限がある場合連携プロジェクト管理メニューで追加ボタンが有効になります。

連携プロジェクトを確認する

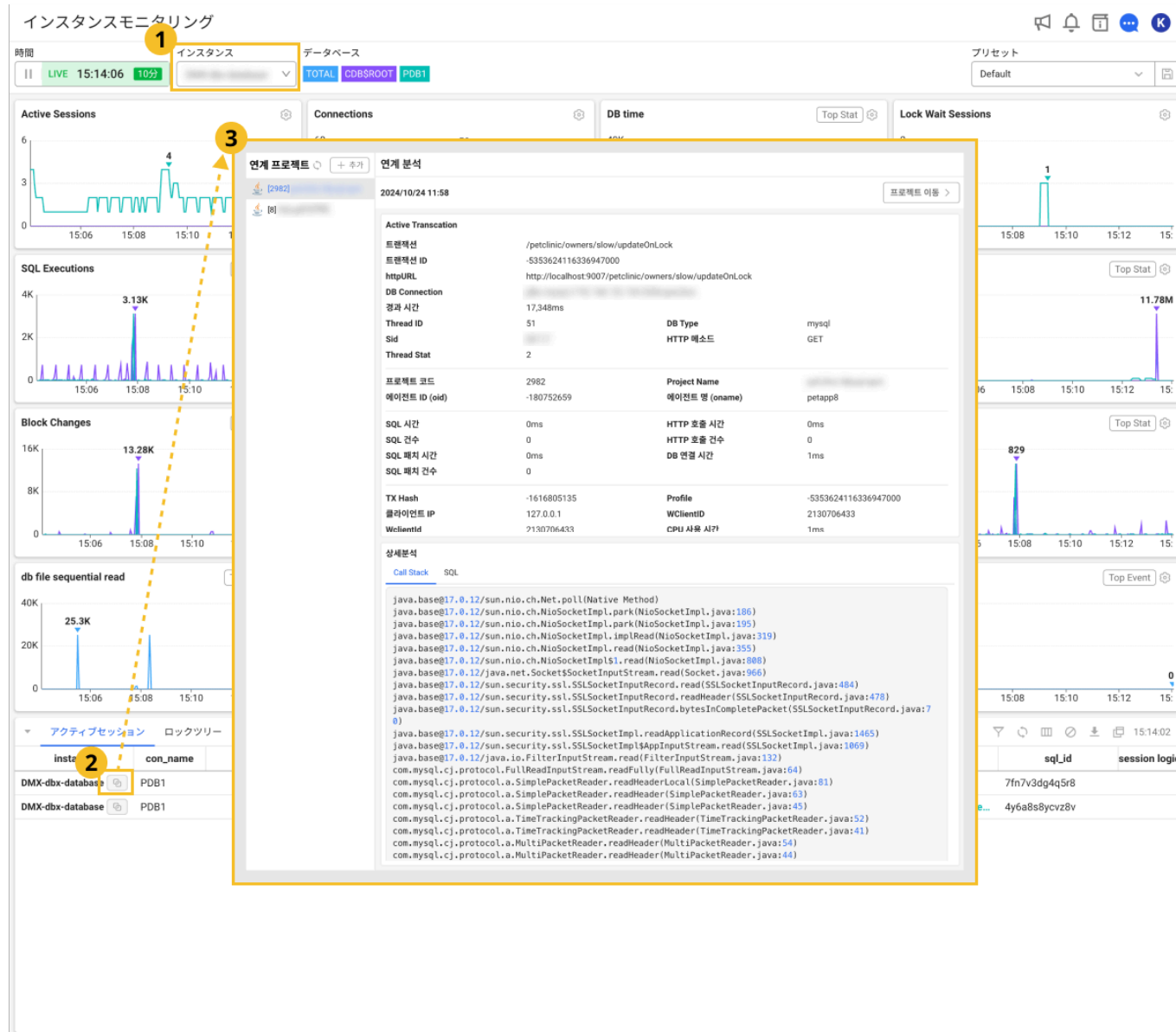
データベースで有効になったセッション(アクティブセッション)と連動したアプリケーションのアクティブTXに関する詳細情報をリアルタイムで提供します。ユーザーは、トランザクションURLおよびデータベース接続情報、DBタイプ、HTTPメソッド、実行時間などの基本情報とともに、トランザクションID、スレッドID、クライアントIP、CPU使用時間などの具体的なメトリクスを確認できます。また、SQLクエリの実行時間、呼び出されたSQLの数、DB接続時間などもモニタリングでき、システムの性能を総合的に分析できます。

特に、スタックトレースによるコールスタック(Call stack) 情報とSQLクエリを一緒に提供し、問題発生時の詳細な原

因を追跡して解決するのに役立ちます。これにより、アプリケーションとデータベース間の相互作用を深く分析し、性能のボトルネック現象を迅速に診断できます。

アクティブTXを追跡する

ダッシュボード > インスタンスモニタリングメニューに移動します。連携分析機能は、リアルタイム照会状態でのみ確認できます。



1. インスタンスオプションで照会したいインスタンス(エージェント)を選択します。
2. アクティブセッションテーブル一覧で連携分析しようとするアクティブセッション項目の🔗ボタンを選択します。
3. 選択したアクティブセッションに関連するトランザクション情報を提供する**連携分析**画面が表示されます。

連携プロジェクト一覧でご希望のプロジェクトを選択し、連携分析情報を確認します。

- ❗ • 画面右上の**プロジェクト移動**ボタンを選択すると、当該プロジェクトの**アプリケーションメニュー**に移動できます。
- 選択したアクティブセッションに連携されたトランザクションがない場合、**データなし**として画面に表示されます。

連携分析画面案内

連携分析画面では、アクティブTXに関する詳細情報およびCall Stack、SQL文を確認できます。

Active Transaction

連携したアクティブTXの詳細を確認できます。

Active Transcation			
TX	/petclinic/owners/{id}		
TXID	8464258678557279000		
httpURL	http://localhost:9007/petclinic/owners/1		
DB Connection	jdbc:mysql://127.0.0.1:3306/petclinic		
経過時間	7,170ms		
Thread ID	49	DB Type	mysql
Sid	49530	HTTPメソッド	GET
Thread Stat	RUNNABLE		
プロジェクトコード	2982	Project Name	petclinic-2982-agent
エージェントID(oid)	-180752659	エージェント名(oname)	petapp8
SQL時間	0ms	HTTPコール時間	0ms
SQL件数	0	HTTPコール件数	0
SQLパッチ時間	0ms	DBコネクション時間	1ms
SQLフェッチ数	0		
TX Hash	-1597521412	Profile	8464258678557279000
クライアントIPアドレス	127.0.0.1	WClientID	2130706433
Wclientid	2130706433	CPU使用時間	2ms

- **DB Connection:** データベース接続URLです。
- **Thread ID:** そのトランザクションを処理するために使用されたスレッドの固有識別子です。
- **Sid:** 選択したアクティブセッションのIDです。
- **Thread Stat:** スレッドの現在の状態です。
- **DB Type:** 連携したデータベースプラットフォームの種類です。

- ❗
- **Javaエージェント2.2.40**以前のバージョンでは**ThreadStat**値が数値形式で表示されます。文字形式で表示するには、連携したプロジェクトで**Javaエージェント2.2.40**バージョン以降をインストールします。
 - その他の項目についての詳細は、[次の文書](#)を参照してください。

Call Stack

メソッドや関数呼び出しの順序を追跡する情報を提供します。

詳細情報

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

連携されたアクティブTXによって実行されたSQLクエリです。

詳細情報

Call Stack SQL

SQLフォーマット



```

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,p
1_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

```

連携プロジェクトを削除する

1. 管理 > 連携プロジェクト管理メニューに移動します。
2. 連携プロジェクトのリストセクションで連携を解除しようとするプロジェクトを削除するには、一覧の一番

右にある  ボタンを選択します。

3. 確認メッセージが表示されたら削除ボタンを選択します。

連携プロジェクトのリストセクションで削除したプロジェクトは除外されます。

実験室

ユーザーに新機能または試験的な機能を提供します。サイドメニューの  **実験室** または **サイトマップ** からアクセスできます。商品によって提供する機能が異なる場合があります。

✔ サイトマップから実験室メニューにアクセスする

サイドメニューに  **実験室** が表示されていない場合は、 **サイトマップ** からアクセスできます。

1. 画面左下の  **サイトマップ** アイコンをクリックしてください。
2. **サイトマップ** ウィンドウの右下にある **実験室** セクションで、目的のメニューを選択してください。

⚠ 注意

実験室の機能は現在開発中のベータ版であり、予期せぬ挙動やエラーが発生する可能性があります。十分な検証の上でご利用ください。重要なデータがある環境や運用環境では使用を推奨しません。フィードバックや問題点に対するお問い合わせは、サポートチーム(support@whatap.io)までご連絡ください。

MXQL Data

MXQLは、WhaTapの性能データ(メトリクス)を柔軟に検索するためのクエリ言語です。プロジェクトに含まれる複数のエージェントから収集したメトリクスを統合的に照会・活用するために使います。**MXQL**の詳細については、[次のドキュメント](#)をご参照ください。

OpenMetricsエクスプローラー

OpenMetricsエクスプローラーは、Prometheus ベースのカスタムメトリクスを PromQL で直接探索し、さまざまな形式で可視化して、メトリクスの状態を確認したりインサイトを得たりできる機能です。

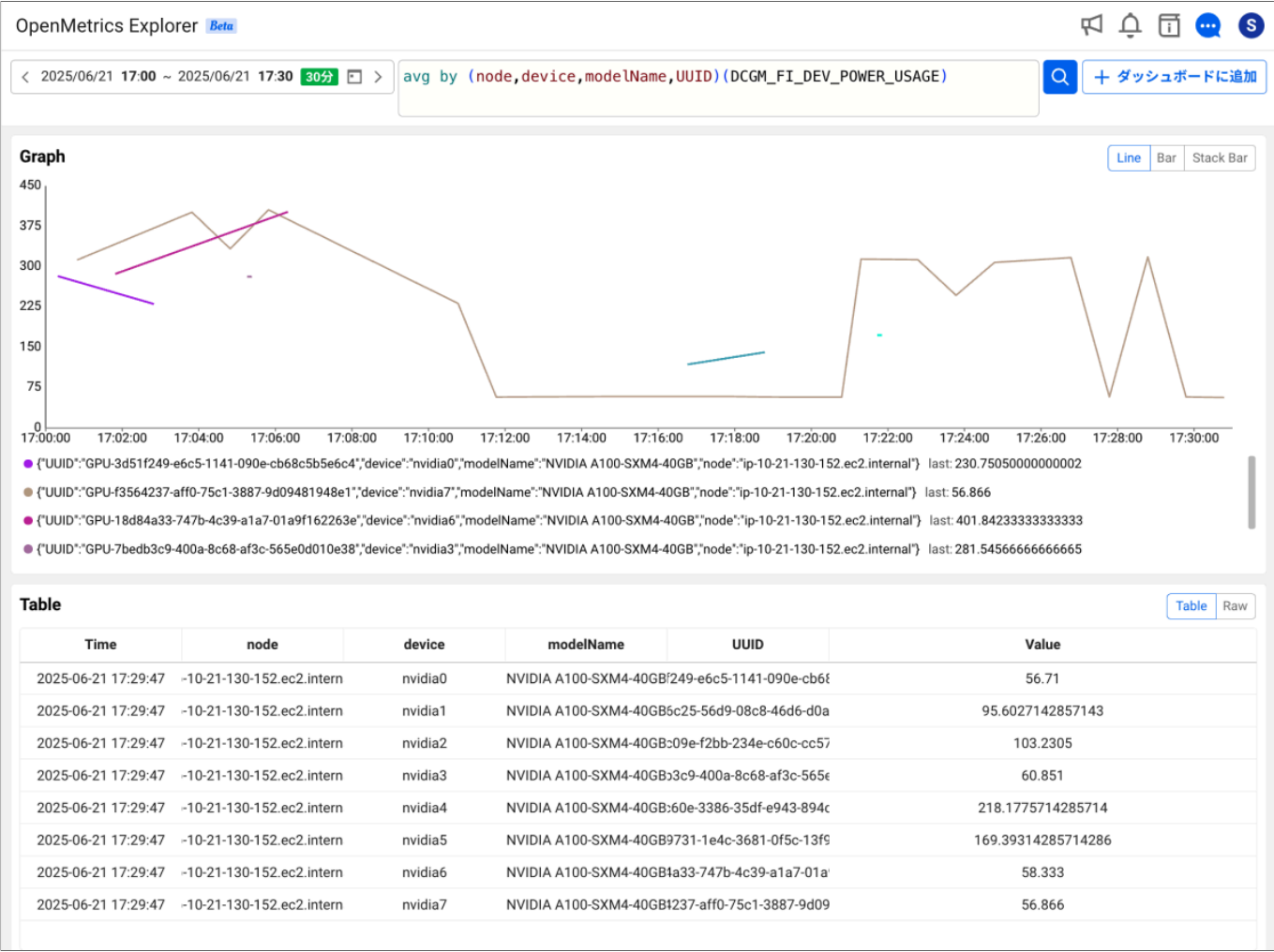
この機能を利用することで、必要なメトリクスを自由に照会し、有意義なデータを抽出したり、ダッシュボードに連携して継続的にモニタリングできます。

❗ OpenMetricsエクスプローラーを利用するには、Open Agent のインストールが必要です。

- [WhaTap Operator のインストール](#) ドキュメントを参照してください。

PromQL で探索する

OpenMetricsエクスプローラーでは、PromQL を実行し、さまざまな可視化オプションで結果を確認できます。



グラフ

- **Line:** デフォルトの時系列折れ線グラフ
- **Bar:** 棒グラフ形式で可視化
- **StackBar:** 積み上げ値を表示するスタック型棒グラフ

テーブル / Raw

- **Table:** 各時系列の値を時間順に並べて表形式で表示
- **Raw:** Prometheus API の応答 (JSON 形式) をそのまま確認可能

PromQL 文法サポートとオートコンプリート

`node_cpu_seconds_total` や `container_memory_usage_bytes` などを入力すると、オートコンプリート機能でメトリクスや演算子の候補が提示されます。

`{pod="example"}` 形式のラベルフィルタリングにも対応しています。

ダッシュボードに追加する

照会した PromQL クエリをダッシュボードウィジェットとして保存し、対象メトリクスを継続的にモニタリングできます。

追加方法

1. OpenMetricsエクスプローラー 画面右上の **+ダッシュボードに追加** ボタンをクリックします。
2. **ダッシュボードに追加** ウィンドウで既存のダッシュボードを選択するか、新しいダッシュボードを作成します。
3. **適用** ボタンをクリックすると、指定したダッシュボードにウィジェットが追加されます。
 - 追加されたダッシュボードでウィジェット名と説明を入力できます。
 - ウィジェットは Flex ボードのグリッドレイアウトで自由に配置でき、PromQL を修正できます。