



OASE Operation Autonomy
Support Engine

Grafana連携

※本書では「Operation Autonomy Support Engine」を「OASE」として記載します。

Exastro Operation Autonomy Support Engine Version 1.5

Exastro developer

目次

1. はじめに

[1.1 Grafana連携\[実習\]について](#)

2. シナリオ説明

[2.1 本書のシナリオ](#)

3. 事前設定

[3.1 Grafana 事前設定](#)

[3.2 OASE 事前設定](#)

[3.3 監視アダプタ設定](#)

4. 作業実行

[4.1 ディジションテーブルファイルの作成](#)

[4.2 ディジションテーブルファイルのアップロード](#)

[4.3 テストリクエスト](#)

[4.4 プロダクション適用](#)

[4.5 アラート発火](#)

[4.6 アクション実行結果の確認](#)

1. はじめに

1.1 Grafana連携【実習】について (1/2)

本書について

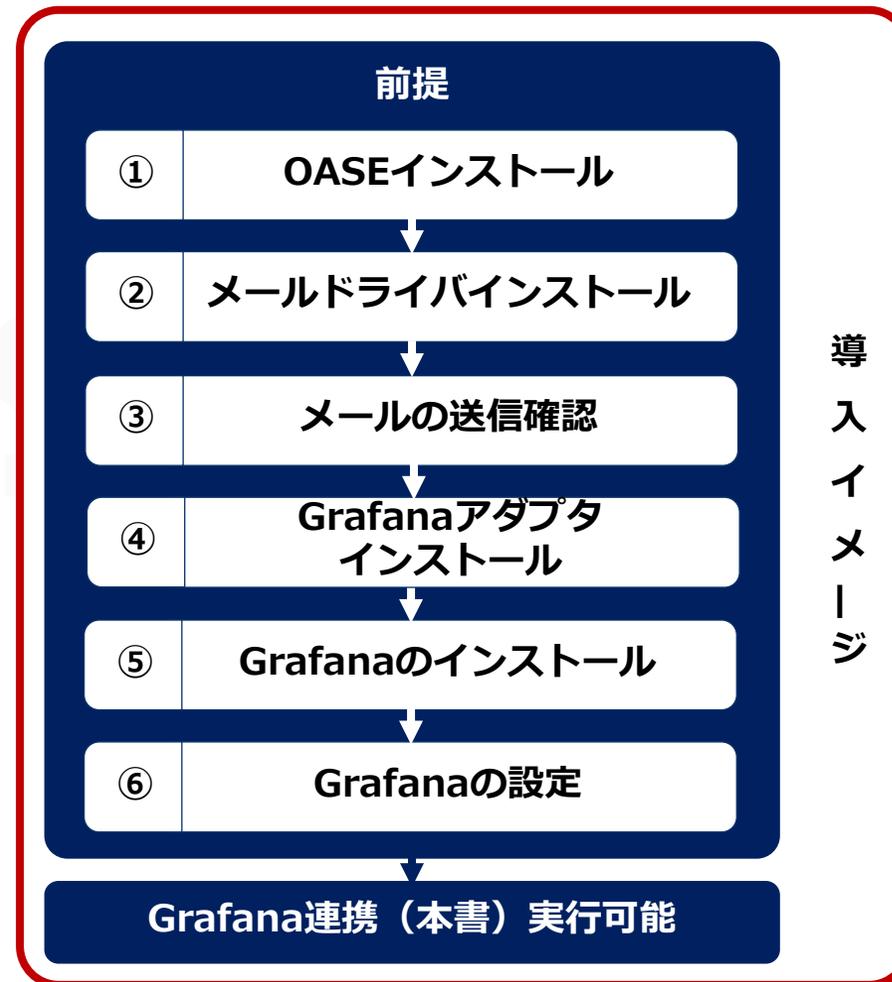
本書では Exastro OASE と監視ソフトウェア「Grafana」の連携について解説しています。

OASE はいくつかのソフトウェアと連携が可能ですが、本書では「Grafana アダプタ」および「メールドライバ」との連携をします。包括的な内容は、OASE の公式マニュアル集である [OASE docs](#) をご参照ください。

前提

OASE および以下の設定が完了していること

- 「① OASE インストール」～「③ メール送信確認」については、[OASE クイックスタート](#)の「1.1 クイックスタートについて (2/3)」を参照
- 「④ Grafanaアダプタインストール」については、[環境構築マニュアル -ドライバインストール編-](#) を参照
- 「⑤ Grafanaのインストール」～「⑥ Grafanaの設定」については別途公式ドキュメントなどを参照

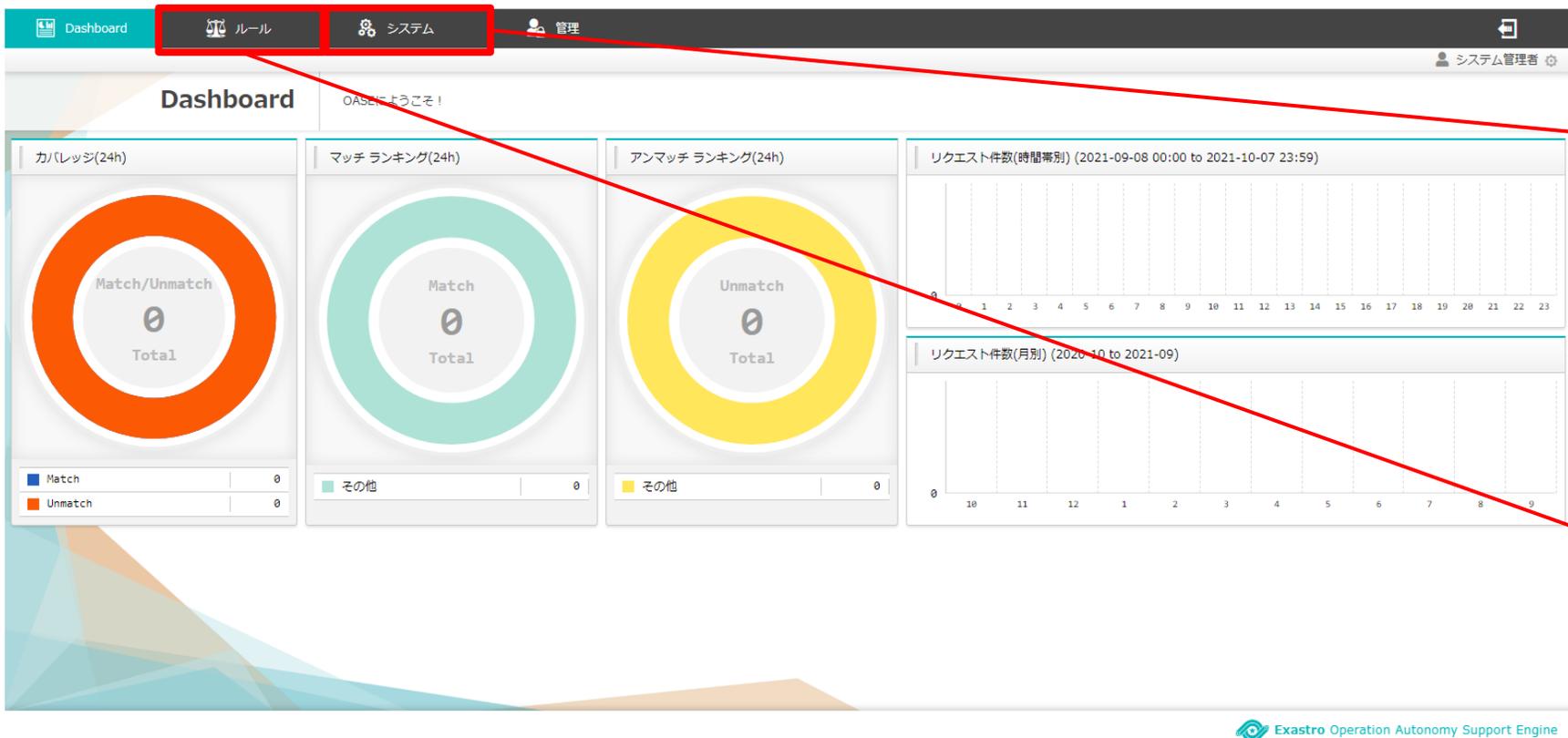


1.1 Grafana連携【実習】について (2/2)

Grafana連携メニュー概要

Grafanaとの連携では、下記のメニューを操作します。

それぞれのメニュー機能についての説明は、Exastro OASEの公式マニュアル集である [OASE docs](#) をご参照ください。



カテゴリ：システム

画面名称

アクション設定

監視アダプタ

カテゴリ：ルール

画面名称

ディジションテーブル

トークン払い出し

ルール

リクエスト履歴

アクション履歴

2. シナリオ説明

2.1 本書のシナリオ (1/2)

シナリオ

下図は本書で実行するシナリオのイメージです。

監視対象のインスタンスが Down することでアラートが発生し、OASE が連動してアクションを実行します。

※実際には、監視対象のインスタンスにある Node Exporter のコンテナを停止させます。

①インスタンスDown
(Node Exporter を停止)



監視対象サーバ



②インスタンスDownの
検知とアラート発報



Grafanaサーバ



③アラートリストを収集し、
ルールとマッチさせる



OASEサーバ

メールドライバの
実行



2.1 本書のシナリオ (2/2)

OASEの事前設定～作業実行

【事前設定】

OASEの各種設定

アクション設定 ※メールドライバ

トークンの払い出し

ディビジョンテーブル作成

監視アダプタ ※Grafanaアダプタ

【作業実行】

ルールの作成・登録

ルールマッチング
およびアクション実行

ディビジョンテーブルファイル作成 ※エクセル操作

ディビジョンテーブルファイルアップロード

テストリクエスト

プロダクション適用

実行

アクション実行結果の確認

3. 事前設定

3.1 Grafana 事前設定 (1/4)

Grafana 事前設定

OASEとGrafanaを連携させるために、Grafanaでは以下の設定をする必要があります。

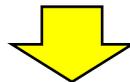
本書では、「Node Exporter」を利用した設定を行います。

環境によって設定方法が変わりますので、ご注意ください。

Prometheusとの連携



ダッシュボード > クエリの作成



ダッシュボード > アラートの設定

3.1 Grafana 事前設定 (2/4)

Dashboard > Panel > Query

クエリの設定

① Data sourceに連携したPrometheusを設定する

② Metrics browserに対象サーバを設定する

他設定につきましては、Grafanaの公式ドキュメントをご参照ください。

The screenshot shows the Grafana Query editor interface. At the top, there are tabs for 'Query 1', 'Transform 0', and 'Alert 1'. Below the tabs, the 'Data source' dropdown is set to 'External' and is highlighted with a red box and a red circle containing the number '1'. To the right of the 'Data source' dropdown, the 'Query options' section shows 'MD = auto = 1166' and 'Interval = 1s'. Below the 'Data source' dropdown, the 'Metrics browser' field contains the query 'up {host="hostname"}' and is highlighted with a red box and a red circle containing the number '2'. Below the 'Metrics browser' field, there are several configuration options: 'Legend' is set to '{{container}}', 'Format' is set to 'Time series', 'Instant' is a toggle switch, 'Prometheus' is a dropdown menu, and 'Exemplars' is a toggle switch. At the bottom left, there are buttons for '+ Query' and '+ Expression'.

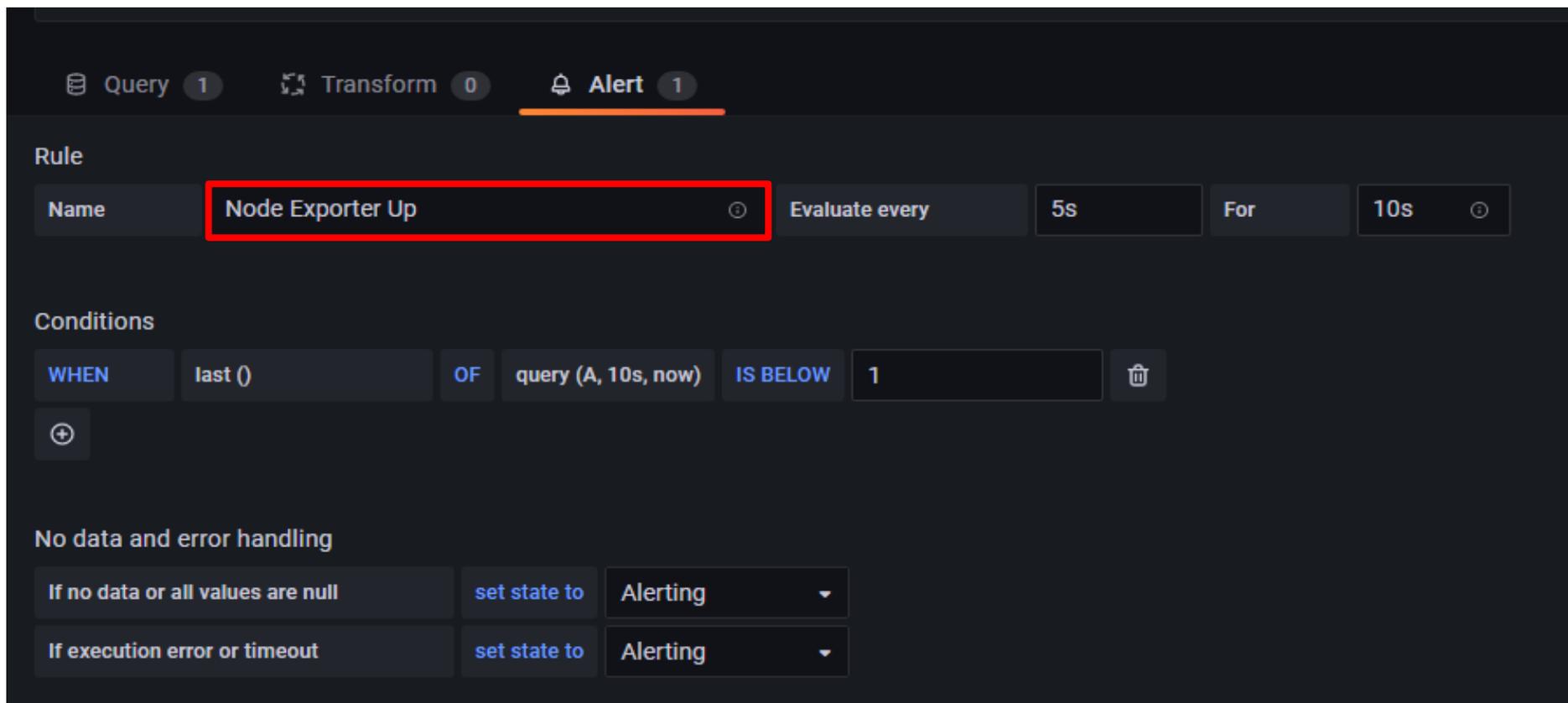
3.1 Grafana 事前設定 (3/4)

Dashboard > Panel > Alert

アラートの設定

OASEでは、定義した名前の情報を検知することができます。

監視間隔等と条件設定については、Grafanaの公式ドキュメントをご参照ください。



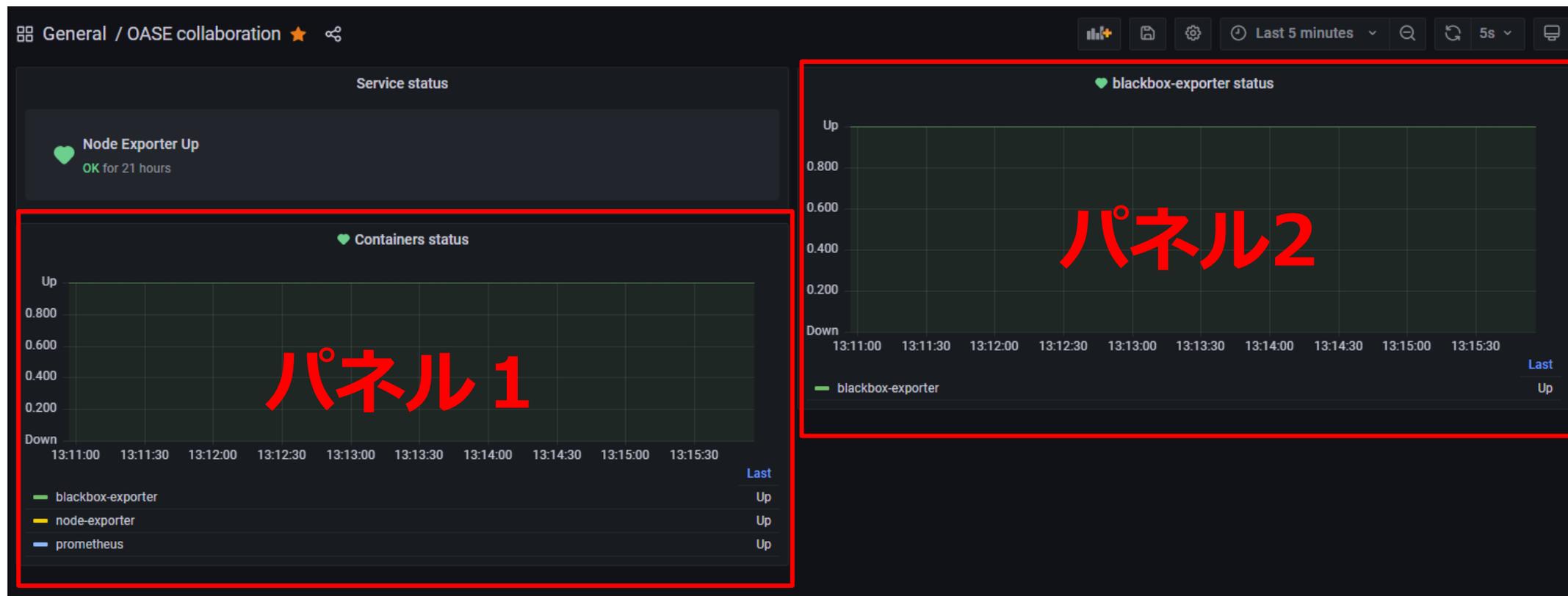
The screenshot shows the Grafana Alert configuration interface. At the top, there are tabs for 'Query' (1), 'Transform' (0), and 'Alert' (1). The 'Alert' tab is selected. Below the tabs, the 'Rule' section is visible. The 'Name' field is set to 'Node Exporter Up' and is highlighted with a red box. The 'Evaluate every' field is set to '5s' and the 'For' field is set to '10s'. Below the 'Rule' section, the 'Conditions' section is visible. It shows a condition: 'WHEN last () OF query (A, 10s, now) IS BELOW 1'. There is a plus sign button to add more conditions. Below the 'Conditions' section, the 'No data and error handling' section is visible. It has two rows: 'If no data or all values are null' and 'If execution error or timeout'. Both rows have a 'set state to' button and a dropdown menu set to 'Alerting'.

3.1 Grafana 事前設定 (4/4)

補足

Grafanaの仕様上、下記の「パネル1」のように1つのパネル内で複数のメトリックスを監視する場合、最初に発生した事象のみが Alerting ステータスになります。つまり、同一パネルで別の事象が新たに発生しても Alerting ステータスは更新されません。(OASEでも取得することはできません)

従って、複数のイベントでルールをマッチングしたい場合には、下記の「パネル2」のように監視対象単位でパネルを分ける必要があります。



3.2 OASE 事前設定

事前設定

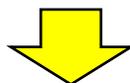
ルールマッチングした場合メール通知するために以下の設定が必要となります。

Zabbix 実習編同様、事前設定が必要となりますので [Exastro OASE Zabbix連携（実習）5.事前設定](#) を参照し以下の設定を行ってください。

アクション設定 メールドライバの追加



トークンの払い出し
※テストリクエスト送信時に利用



ディビジョンテーブルファイルの作成

POINT

事前に[環境構築マニュアル-ドライバインストール編](#)を参照のうえメールドライバをインストールしてください。

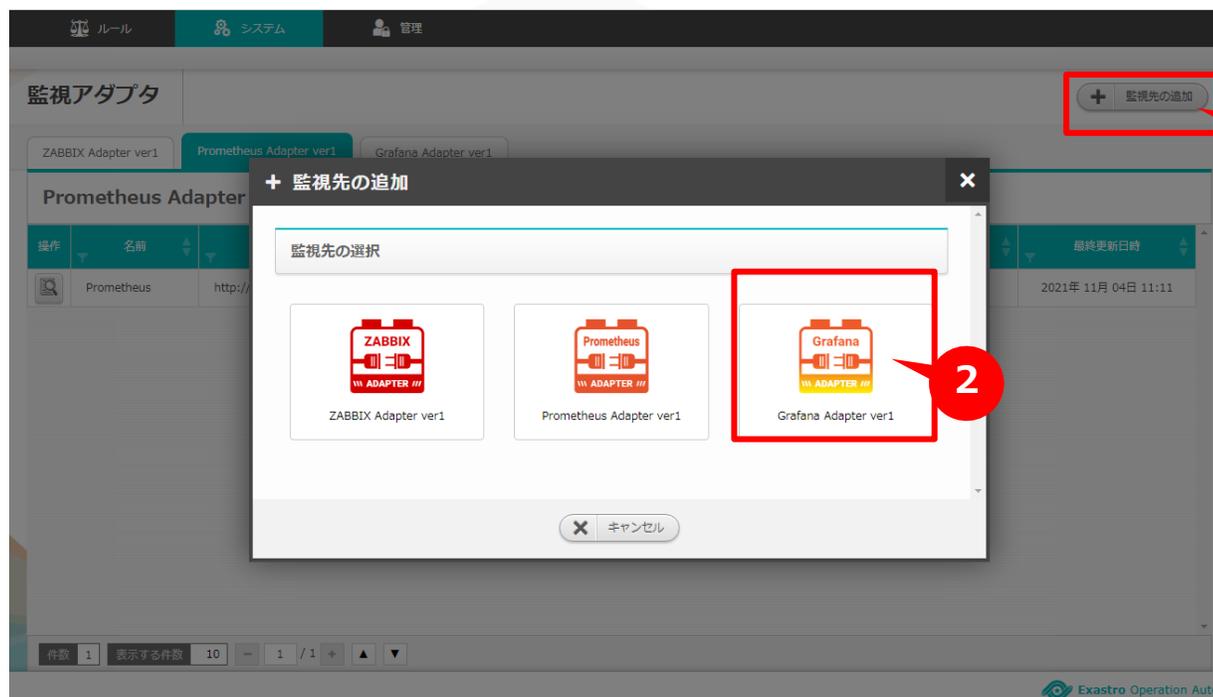
3.3 監視アダプタ設定 (1/3)

システム > 監視アダプタ

監視先の追加

Grafanaアダプタを追加します。

- ① 監視先の追加を押下する
- ② Grafana Adapter ver1を押下する



※ドライバをインストールしていない場合、上記の画面は表示されません。

3.3 監視アダプタ設定 (2/3)

システム > 監視用アダプタ

監視アダプタ設定

① Grafana Adapter ver1の必要な情報を入力する

名前 * ⓘ grafana

URI * ⓘ http://...

ユーザ名 * ⓘ admin

パスワード * ⓘ

ディビジョンテーブル名選択 grafana_adapter

監視情報

	Grafana項目
イベント発生日時 *	[] .newStateDate
インスタンス情報 *	[] .id

条件名

	Grafana項目
アラート名 * ⓘ	[] .name

× キャンセル 戻る 保存

項目	設定値
名前	(任意の文字列)
URI	http://xx.xxx.xx.xxx:xxx/api/alerts (アラートAPIのエンドポイント)
ユーザ名	Grafanaのユーザ名
パスワード	Grafanaのパスワード
ディビジョンテーブル名選択	grafana_adapter (イベントリクエストの実施対象となるディビジョンテーブル名をプルダウンで選択する)

監視用アダプタ設定

#アラート情報

```
[  
  {  
    "id": 1, ← インスタンス情報  
    "dashboardId": 1,  
    "dashboardUid": "ycVfzCx7z",  
    "dashboardSlug": "oase-collaboration",  
    "panelId": 2,  
    "name": "Node Exporter Up", ← アラート名  
    "state": "alerting",  
    "newStateDate": "2022-02-03T06:16:41Z", ← イベント発生日時  
    "evalDate": "0001-01-01T00:00:00Z",  
    "evalData": {  
      "evalMatches": [  
        {  
          "metric": "node-exporter",  
          "tags": {  
            "__name__": "up",  
            "container": "node-exporter",  
            "host": "oase-learn-monitoring",  
            "instance": "node-exporter:9100",  
            "job": "node-exporter"  
          },  
          "value": 0  
        }  
      ],  
    }  
  }  
]
```

<突合情報>

突合情報には、JSONの文法に従い記述する必要があります。

イベント発生日時: アラート更新日時にあたる情報を指定。

インスタンス情報: アラートをユニークに識別可能な値を指定。

条件: ルールマッチングさせたい項目を指定。

(設定値は環境によって異なる可能性があるのでご注意ください)

4. 作業実行

4.1 ディジジョンテーブルファイル作成

ルール > ディジジョンテーブル

ディジジョンテーブルファイルのダウンロードおよび作成

- ① 事前設定で作成したディジジョンテーブルの「ダウンロードボタン」を押下しディジジョンテーブルファイルをダウンロードする
- ② 以下の内容を記述作成する(詳細については[Exastro OASE docs](#)をご参照ください。)

ルール説明	アラート名(必須)	ルール名(必須)	発生事象(必須) ※平常の場合は「X」を記載	発処概要(必須) ※平常の場合は「X」を記載	アクション種別(必須)	アクションパラメータ情報(必須) ※ダブルクォーテーションは使用不可	承認パラメータ情報(必須) ※ダブルクォーテーションは使用不可	リトライ回数	リトライ間隔	抑止回数	抑止間隔	条件回数	条件期間	大グループ(必須) ※平常の場合は「X」を記載	小グループ(必須) ※平常の場合は「X」を記載	優先順位	保存日	検出日
Grafana連携	Node Exporter Up	grafana_adapter	インスタンスダウン	アラート通知	mail(ver1)	MAIL_NAME=mail,MAIL_TO=,MAIL_CC=,MAIL_BCC=,MAIL_TEMPLATE=mail	X	1	1	1	1	X	X	X	X	X		



ルール説明	Grafana連携	リトライ間隔	1
アラート	Node Exporter Up	リトライ回数	1
ルール名	Grafana_adapter	抑止間隔	1
発生事象	インスタンスダウン	抑止回数	1
対処概要	アラート通知	条件回数	X
アクション種別	mail(ver1)	条件期間	X
アクションパラメータ情報	MAIL_NAME= (メールアドレス), MAIL_TO=,MAIL_CC=,MAIL_BCC=, MAIL_TEMPLATE=(メールテンプレート名)	大グループ	X
承認パラメータ情報	X	優先順位	X
		小グループ	X
		優先順位	X

ディジジョンテーブルファイル作成

ディジジョンテーブルファイル
アップロード

テストリクエスト

プロダクション適用

アラート発火

アクション実行結果の確認

4.2 ディジジョンテーブルファイルのアップロード

ルール > ルール

ディジジョンテーブルのアップロード

作成したファイルを選択しアップロードします。

- ① 「ファイル選択」を押下し作成したディジジョンテーブルファイルを選択する
- ② 「アップロード」を押下する
- ③ ステージング適用完了になっていることを確認する

Dashboard navigation: dashboard | ルール | システム | 管理

Staging Applicable Rules: テストリクエスト | ファイルを選択: id00000000015.xlsx | アップロード | 過去を含め表示 OFF

操作	ディジジョンテーブル名	ルールファイル	運用ステータス	作業ステータス	最終更新者	最終更新日時
	grafana_adapter	id00000000015.xlsx	検証未実施	ステージング適用完了	システム管理者	2022年 02月 08日 10:11

ディジジョンテーブルファイル作成

ディジジョンテーブルファイル
アップロード

テストリクエスト

プロダクション適用

アラート発火

アクション実行結果の確認

4.3 テストリクエスト

ルール > ルール

登録したルールのテスト

- ① 「テストリクエスト」を押下する
- ② ディジジョンテーブルタブを押下し作成したディジジョンテーブルをプルダウン選択する
- ③ 設定タブを押下し作成したルールと合致する値を入力する
- ④ 実行ボタンを押下しルールに合致したことを確認する
- ⑤ 閉じるボタンを押下しステータスを検証完了にする

ディジジョンテーブルファイル作成

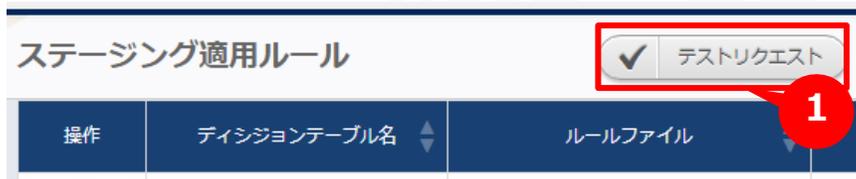
ディジジョンテーブルファイル
アップロード

テストリクエスト

プロダクション適用

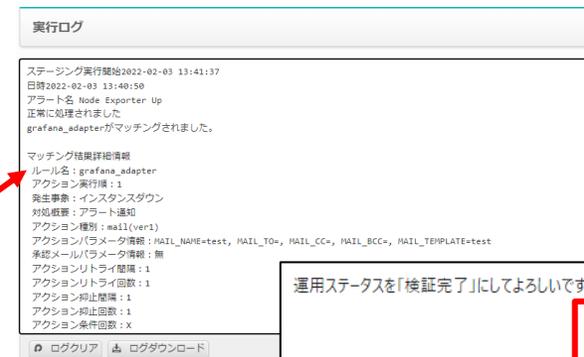
アラート発火

アクション実行結果の確認



3 以下のように記述

アラート Node Exporter Up



運用ステータスを「検証完了」にしてよろしいですか？

4.4 プロダクション適用

ルール > ルール

検証完了したルールを本番環境で使用できるようにする

- ① チェックマークを押下しプロダクション適用を開始する
- ② 作業ステータスがプロダクション適用完了になっていることを確認する

ステージング適用ルール

✓ テストリクエスト 📁 ファイルを選択！ ファイルが選択されていません。 📄 アップロード 🕒 過去を含め表示 OFF

操作	ディビジョンテーブル名	ルールファイル	適用ステータス	作業ステータス	最終更新者	最終更新日時
<input checked="" type="checkbox"/>	grafana_adapter	id00000000015.xlsx	検証完了	ステージング適用完了	システム管理者	2022年 02月 03日 13:46

1

プロダクション適用します。よろしいですか？

OK キャンセル

プロダクション適用ルール

🕒 過去を含め表示 OFF

操作	ディビジョンテーブル名	ルールファイル	適用ステータス	作業ステータス	最終更新者	最終更新日時
<input type="checkbox"/>	grafana_adapter	id00000000015.xlsx	プロダクション適用	プロダクション適用完了	システム管理者	2022年 02月 03日 14:36

2

ディビジョンテーブルファイル作成

ディビジョンテーブルファイルアップロード

テストリクエスト

プロダクション適用

アラート発火

アクション実行結果の確認

node-exporterの起動と確認

① 「node-exporter」を起動する

```
# docker-compose up -d node-exporter
Starting node-exporter...done
```

② 「node-exporter」がUpになっていることを確認する

```
# docker-compose ps
Name                Command                State                Ports
-----
node-exporter       /bin/node_exporter    Up                  0.0.0.0:9100->9100/tcp,::9100->9100/tcp
```

ディビジョンテーブルファイル作成

ディビジョンテーブルファイル
アップロード

テストリクエスト

プロダクション適用

アラート発火

アクション実行結果の確認

node-exporterを停止(アラート発火)

③ 「node-exporter」を停止する

```
# docker-compose stop node-exporter
Stopping node-exporter...done
```

④ Exit2になっていることを確認する

```
# docker-compose ps
```

Name	Command	State	Ports
node-exporter	/bin/node_exporter	Exit2	

ディビジョンテーブルファイル作成

ディビジョンテーブルファイル
アップロード

テストリクエスト

プロダクション適用

アラート発火

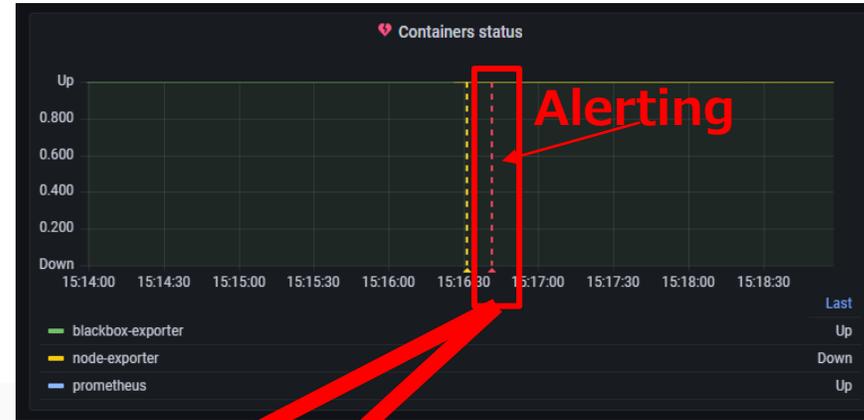
アクション実行結果の確認

4.6 アクション実行結果の確認

各種画面を確認する

OASEの「リクエスト履歴」と「アクション履歴」画面に該当のディビジョンテーブル名が上がっていることを確認します。

OASEではstatusがAlertingになったタイミングでルールを評価し、履歴に反映します



リクエスト履歴			
ルールマッチ状況	リクエスト種別	ディビジョンテーブル名	リクエスト受信日時
✓	プロダクション環境	grafana_adapter	2022年2月3日15:16

アクション履歴						
状態	操作	ディビジョンテーブル名	ルール名	発生事象	対処概要	ア
✓		grafana_adapter	grafana_adapter	インスタンスダウン	アラート通知	mai

- ディビジョンテーブルファイル作成
- ディビジョンテーブルファイルアップロード
- テストリクエスト
- プロダクション適用
- アラート発火
- アクション実行結果の確認**



Exastro