【Field Magic】Grafana マニュアル

Version: v2.0.0

Rev	更新日	更新内容・理由
v1.0	2025.12.25	新規発行
v2.0	2026.06.26	マニュアル構成の変更、及び、画像や文言の調整

目次

1. はじめに
2. 構成例
3. Grafana上でのデータ可視化の設定方法
    3-1. ダッシュボードとパネルの作成
    3-2. 温度センサーのパネル設定
    3-3. 湿度センサーのパネル設定
    3-4. ダッシュボードの保存
    3-5. ダッシュボードやパネルのカスタマイズ
4. トラブルシューティング
    4-1. パネルに「No data」と表示される
    4-2. パネルにエラーが表示される
    4-3. ダッシュボードが編集できない
5. 時系列DBへのデータ保存方法
    5-1. 時系列DBの概要
    5-2. NodeRED上での時系列データの保存方法
        5-2-1. 1分毎のフロー実行設定
        5-2-2. ネットワークデバイスからのデータ取得設定
        5-2-3. 時系列データ保存用のデータ加工設定
        5-2-4. 時系列データ保存設定
    5-3. 動作確認
    5-4. トラブルシューティング

1. はじめに

Field Magicの機能を用いて温湿度センサーの値をコントローラー内に保存し、可視化する手順例を紹介します。
本機能は、温湿度や電力量、照明の状態など、点在する情報を一覧表示したり、過去のトレンドを把握したりしたい現場にご活用いただけます。

💡 下記の事前準備を完了しておく必要があります。
時系列DBへのデータの保存

💡 本マニュアルの内容はあくまで一例です。
実際のユースケースに応じて、設定内容を調整する必要があります。

⚠️ コントローラー内部にデータを保存する場合、データのバックアップを定期的に取得し、必要に応じて外部ストレージに保存することを推奨します。以下のような場合、保存したデータが失われる可能性があります。

- コントローラーを初期化すると、保存したデータが消失します。
- コントローラーのストレージが故障した場合、保存したデータが失われる可能性があります。

⚠️ 以下のような場合は、外部のデータベースサーバーを利用することを推奨します。

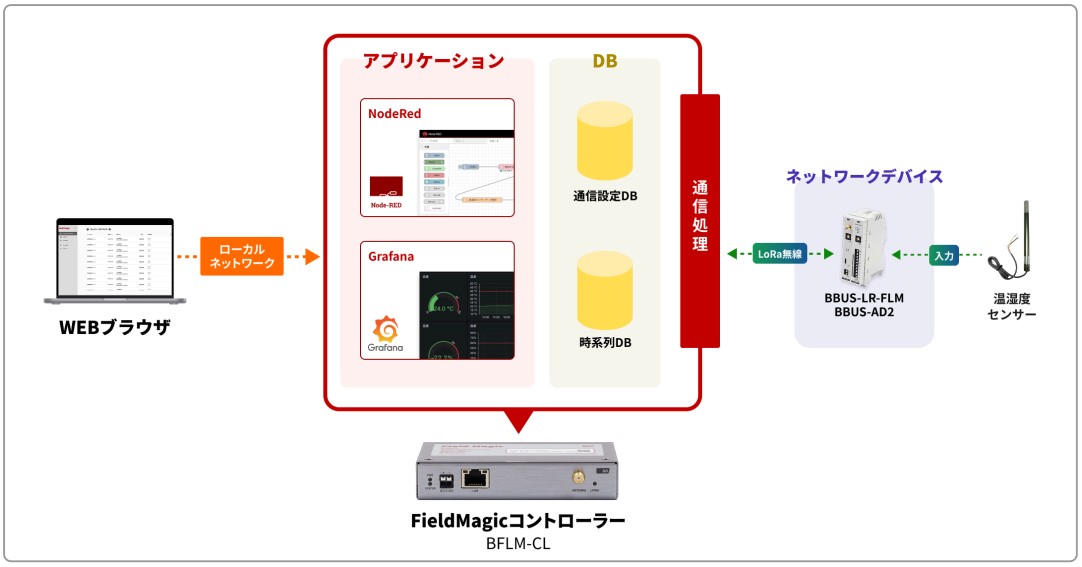
- 多数のセンサーからデータを収集する場合
- 高頻度でデータを収集する場合
- 長期間にわたってデータを保存する場合

2. 構成例

Field Magicコントローラーには、以下のオープンソースがあらかじめインストールされており、これらを活用することでデータの可視化を行います。

ソフトウェア	役割
TimescaleDB	時系列データ(センサーデータのログ)を保存するためのデータベースです。 PostgreSQLがベースになっています。
Grafana	データの可視化を行うためのツールです。 PostgreSQLを含む様々なデータソースに対応しており、ダッシュボードの作成やグラフ表示が可能です。
NodeRED	ローコードでアプリケーションを構築できるツールです。 センサーデータを取得し、TimescaleDBに保存するためのフローを作成します。

このドキュメントで紹介する構成例は以下の通りです。

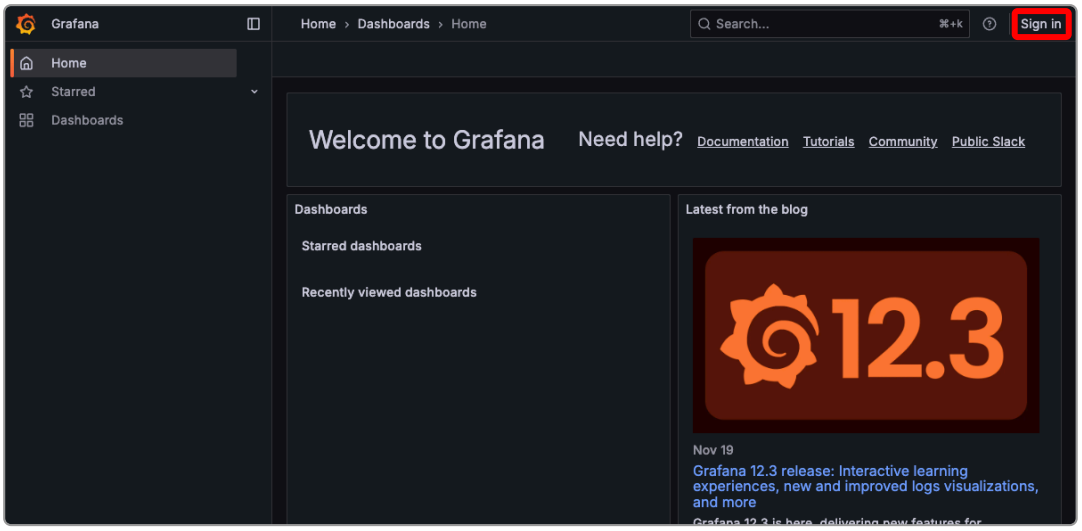


3. Grafana上でのデータ可視化の設定方法

コントローラー内のGrafanaを使用して、TimescaleDBに保存したセンサーデータを可視化する設定を行います。
以下のURLにアクセスして、Grafanaの画面を開きます。

http://<コントローラーの機器コード>.local/grafana

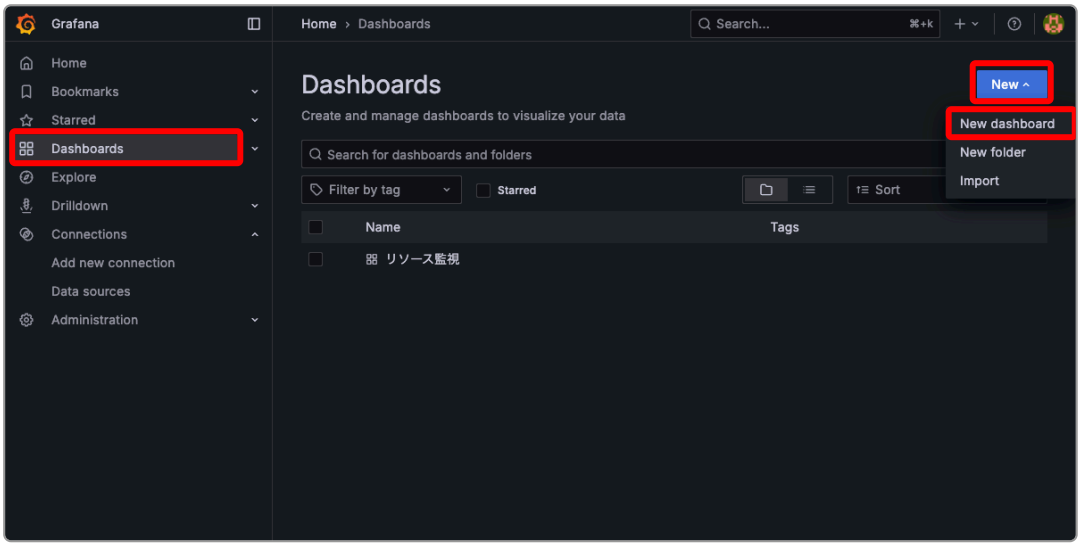
Grafanaでダッシュボードの設定をするには、画面右上の「Sign in」から管理者権限でログインする必要があります。
初期ユーザー名は「admin」、初期パスワードは「password」です。



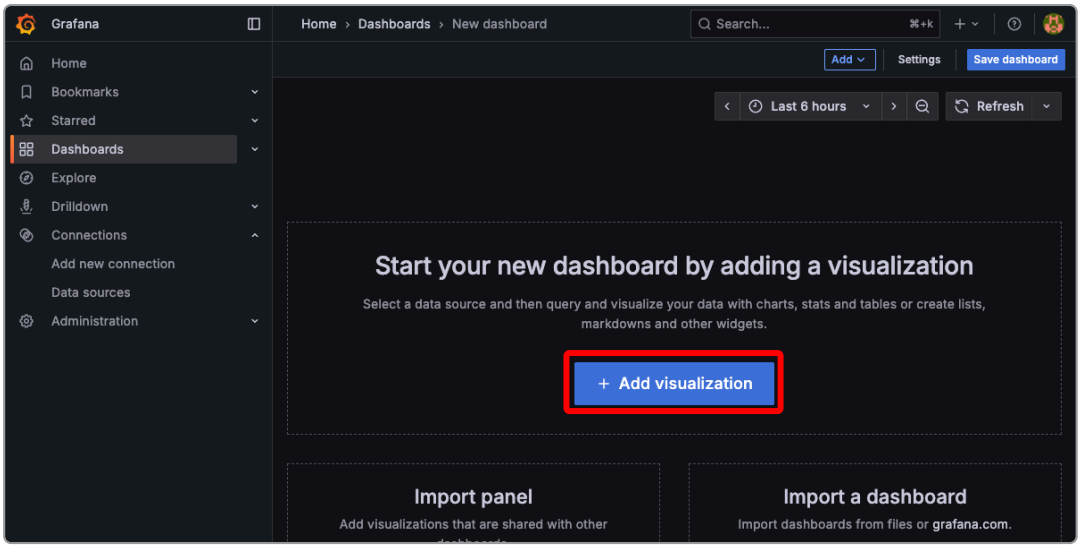
3-1. ダッシュボードとパネルの作成

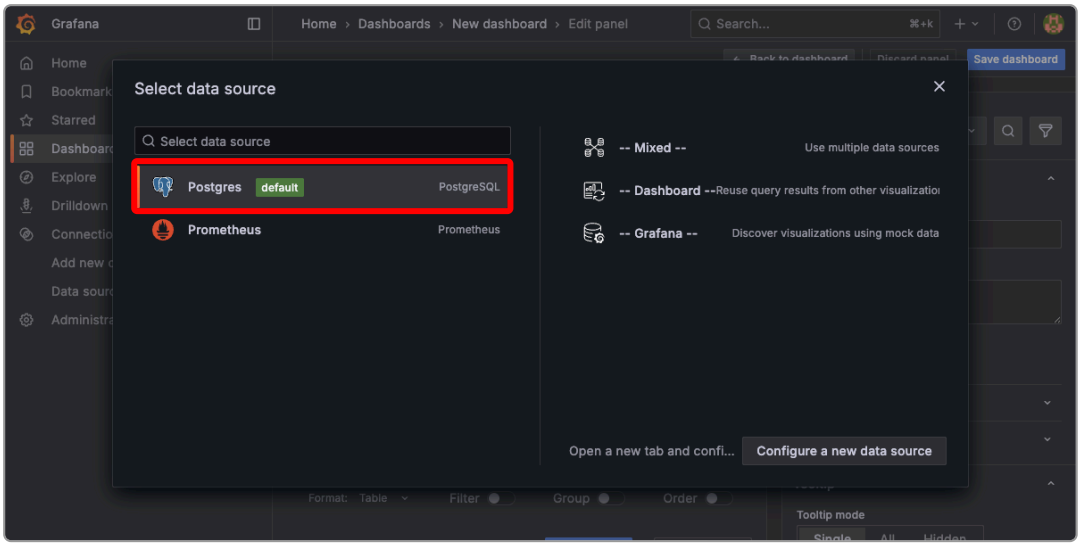
管理者権限でログインしたら、以下の手順でダッシュボードとパネルを作成します。

画面左のメニューから「Dashboards」を選択し、ダッシュボード画面右上の「New」から「New dashboard」を選択します。



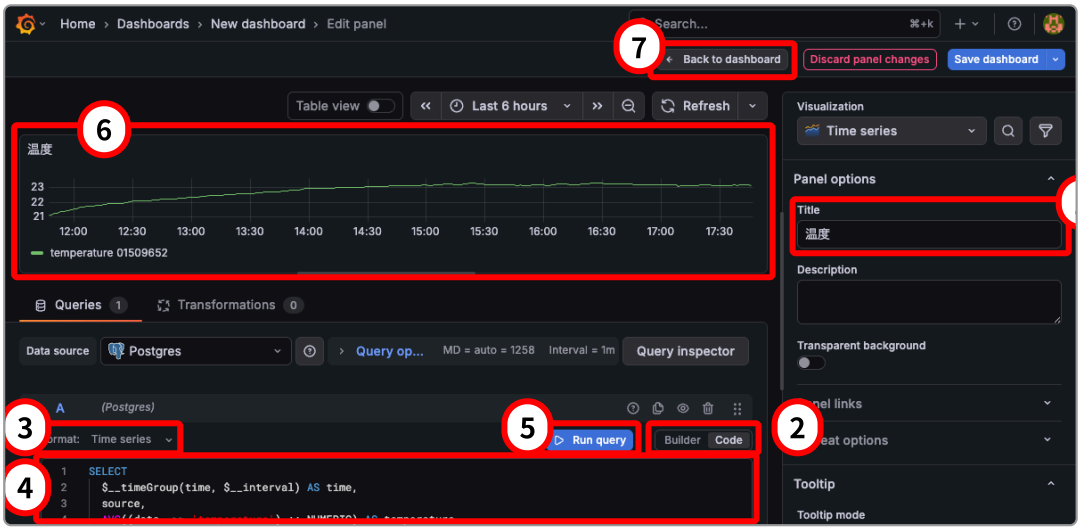
ダッシュボードが作成されたら、「Add visualization」ボタンを押し、その後「Postgres」を押すと、パネルが作成されパネルの設定画面が表示されます。





3-2. 温度センサーのパネル設定

パネルの設定画面で温度センサーのパネル設定を行います。



1. PanelOptionsの「Title」に「温度」と入力します。

2. 「Builder | Code」で「Code」を選択します。
3. Formatで「Time series」を選択します。
4. 入力欄に、以下のSQLクエリを入力します。
5. 「Run query」を実行します。
6. グラフ表示エリアにグラフが表示されることを確認します。
7. 「Back to dashboard」でダッシュボードに戻ります。

```
SELECT
 $__timeGroup(time, $__interval) AS time,
 source,
 AVG((data ->> 'temperature') :: NUMERIC) AS temperature
FROM
 time_series
WHERE
 $__timeFilter(time)
 AND (data ->> 'temperature') :: NUMERIC IS NOT NULL
GROUP BY
 time,
 source
ORDER BY
 time ASC
```

💡 上記SQL内の「temperature」は、時系列データ保存用のデータ加工設定で指定したデータ項目です。
時系列DBの `data` カラムはJSONB型（PostgreSQLのバイナリJSON型）で、`data ->> 'temperature'` のように `->>` 演算子を使うことで、JSONオブジェクト内の任意のプロパティを文字列として取り出すことができます。
数値として扱う場合は `:: NUMERIC` でキャストしています。

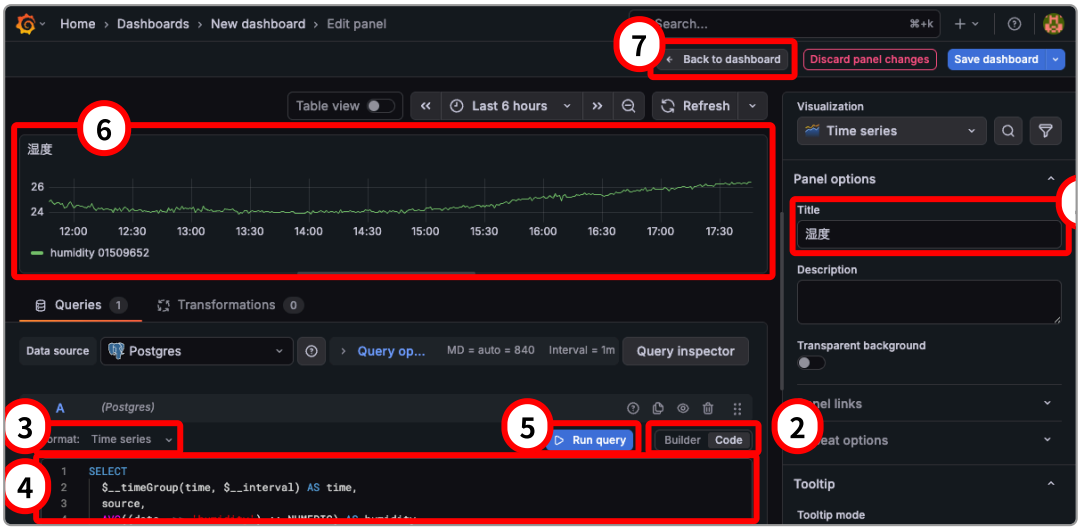
💡 上記SQLクエリでは、Grafanaが提供する以下の変数・関数を使用しています。

- `$__timeGroup(time, $__interval)` — 時刻カラムをGrafanaの表示間隔に応じた単位でグルーピングします。
  ダッシュボードの時間範囲やパネル幅に応じて自動的に最適な間隔が選択されます。
- `$__interval` — Grafanaが自動算出する時間間隔です。
  表示範囲やパネル幅に応じて `1m` 、`5m` 、`1h` などに変化します。
- `$__timeFilter(time)` — ダッシュボード右上で選択した時間範囲に基づく `WHERE` 条件を自動生成します。
  例えば「Last 6 hours」を選択した場合、直近6時間のデータのみが抽出されます。

3-3. 湿度センサーのパネル設定

ダッシュボードの画面右上の「Add」→「Visualization」から新しいパネルを追加し、湿度センサーのパネルの設定画面を開きます。

温度センサーと同様の手順で、湿度センサーのパネル設定を行います。



1. PanelOptionsの「Title」に「湿度」と入力します。
2. 「Builder | Code」で「Code」を選択します。
3. Formatで「Time series」を選択します。
4. 入力欄に、以下のSQLクエリを入力します。
5. 「Run query」を実行します。
6. グラフ表示エリアにグラフが表示されることを確認します。
7. 「Back to dashboard」でダッシュボードに戻ります。

```
SELECT
 $__timeGroup(time, $__interval) AS time,
 source,
 AVG((data ->> 'humidity') :: NUMERIC) AS humidity
FROM
 time_series
WHERE
 $__timeFilter(time)
 AND (data ->> 'humidity') :: NUMERIC IS NOT NULL
GROUP BY
 time,
```

```
  source
ORDER BY
  time ASC
```

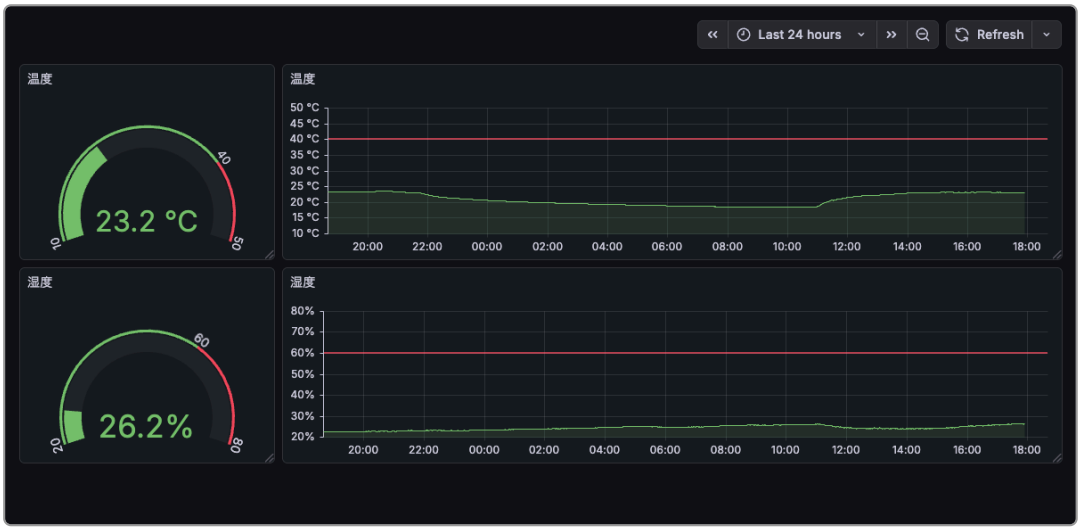
> 💡 上記SQL内の「humidity」は、時系列データ保存用のデータ加工設定 で指定したデータ項目です。

3-4. ダッシュボードの保存

ダッシュボードの設定が完了したら、画面右上の「Save dashboard」ボタンを押し、確認画面で「Save」ボタンを押してダッシュボードを保存します。

3-5. ダッシュボードやパネルのカスタマイズ

作成したダッシュボードやパネルは、Grafanaの豊富なカスタマイズ機能を使用して、見た目や表示内容を自由に変更できます。
詳細なカスタマイズ方法については、Grafanaの公式ドキュメントを参照してください。



4. トラブルシューティング

4-1. パネルに「No data」と表示される

- 時系列DBにデータが保存されていない、またはSQLクエリの時間範囲にデータが存在しない可能性があります。
  時系列DBへのデータ保存フローが正常に動作しているか確認してください。
  また、Grafana画面右上の時間範囲セレクタで、データが存在する期間が選択されているか確認してください。

4-2. パネルにエラーが表示される

- SQLクエリの構文に誤りがある可能性があります。
  エラー内容に従って、SQLクエリを修正してください。

4-3. ダッシュボードが編集できない

- 管理者権限でログインしていない可能性があります。
  画面右上の「Sign in」から adminアカウントでログインしてください。

5. 時系列DBへのデータ保存方法

本項では、コントローラー内蔵の「時系列DB」にデータを保存する手順を記述します。

> 💡 ユーザーガイドに従い、事前にネットワークデバイスの登録を完了しておく必要があります。

> 💡 本項の内容はあくまで一例です。
> 実際のユースケースに応じて、設定内容を調整する必要があります。

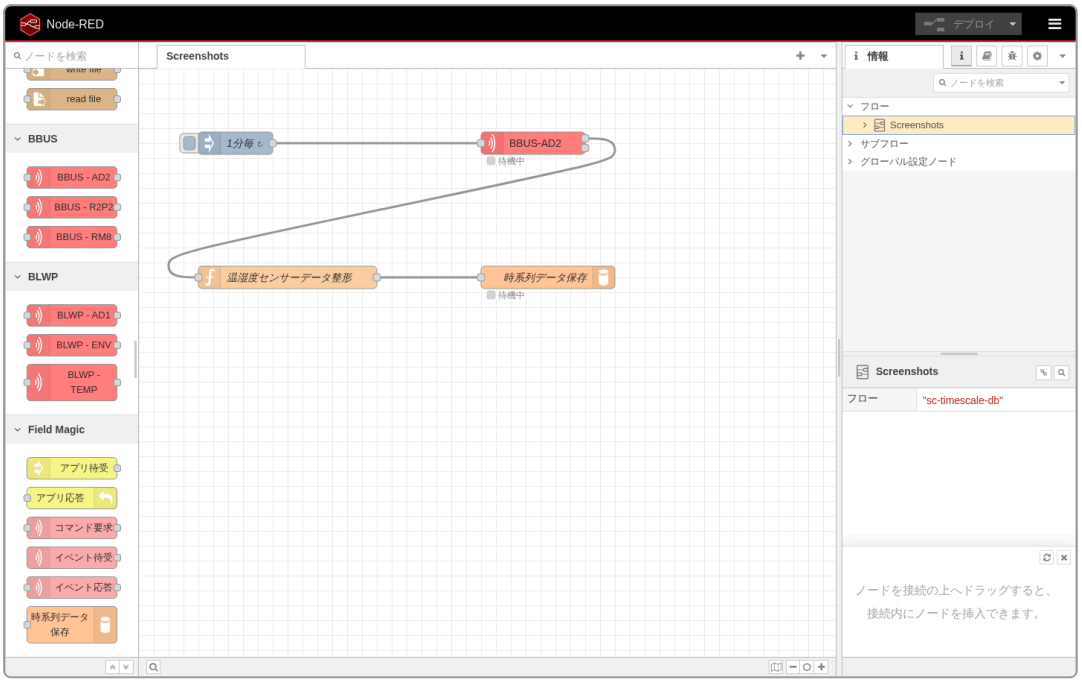
5-1. 時系列DBの概要

時系列DBはField Magic本体に搭載されたデータベースです。
NodeREDの専用ノードを介して、マップアプリやGrafanaなど各アプリケーションから使用されるデータ値を保存します。

5-2. NodeRED上での時系列データの保存方法

ネットワークデバイスから取得したセンサーデータを時系列DBに保存するためのNodeREDフローを作成します。
以下のURLにアクセスして、NodeREDの画面を開きます。

http://<コントローラーの機器コード>.local/red



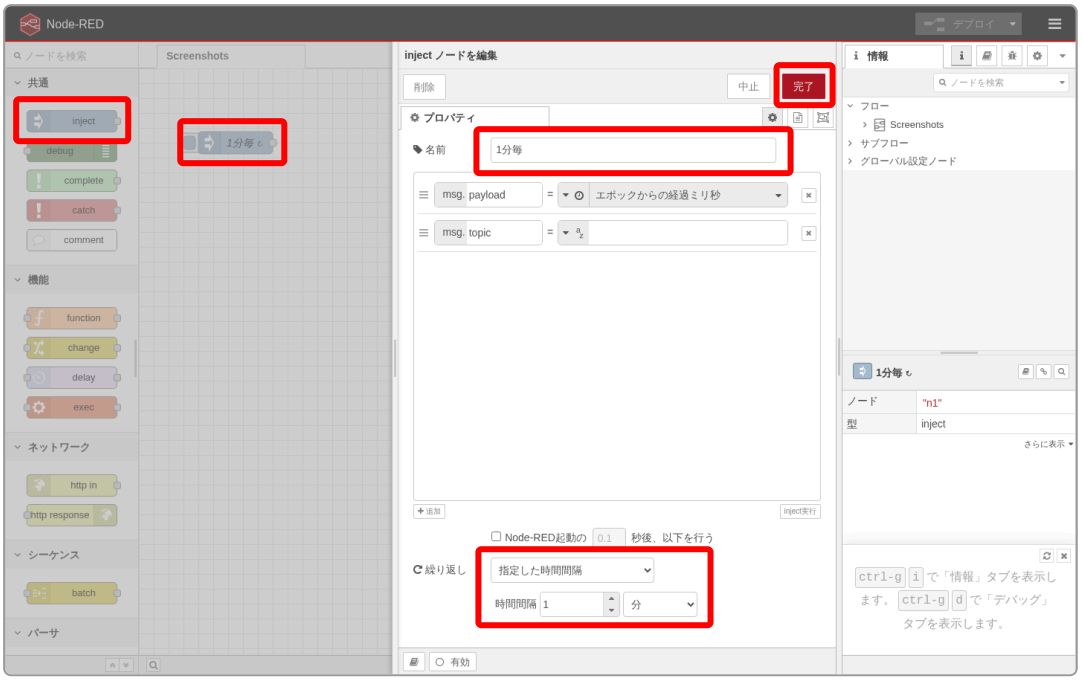
使用しているブロックは以下の4つです。
これらのノードを順番に接続し、各ノードの設定を行います。

ノードカテゴリ名	パッケージ名	ノード名	名前(個別に設定)	説明
共通	node-red	inject	1分毎	フローの実行をトリガーするノードです。 定期的に実行する設定が可能で、ここでは1分毎に自動実行される設定としています。
BBUS	flm-bbus	BBUS - AD2	BBUS-AD2	Field Magic独自のノードです。 ネットワークデバイスを指定してBBUS-AD2のアナログ値取得のコマンドを送信し、センサーデータを取得しています。
機能	node-red	function	温湿度センサーデータ整形	任意のプログラムを実行できるノードです。 ここではBBUS-AD2で取得したデータを、時系列データに保存できるように加工しています。
Field Magic	flm-core	時系列データ保存	時系列データ保存	Field Magic独自のノードです。 任意の時系列データを保存することができます。

> 💡 Field Magicのパッケージ「flm-core」「flm-bbus」は、コントローラーに標準でインストールされています。

5-2-1. 1分毎のフロー実行設定

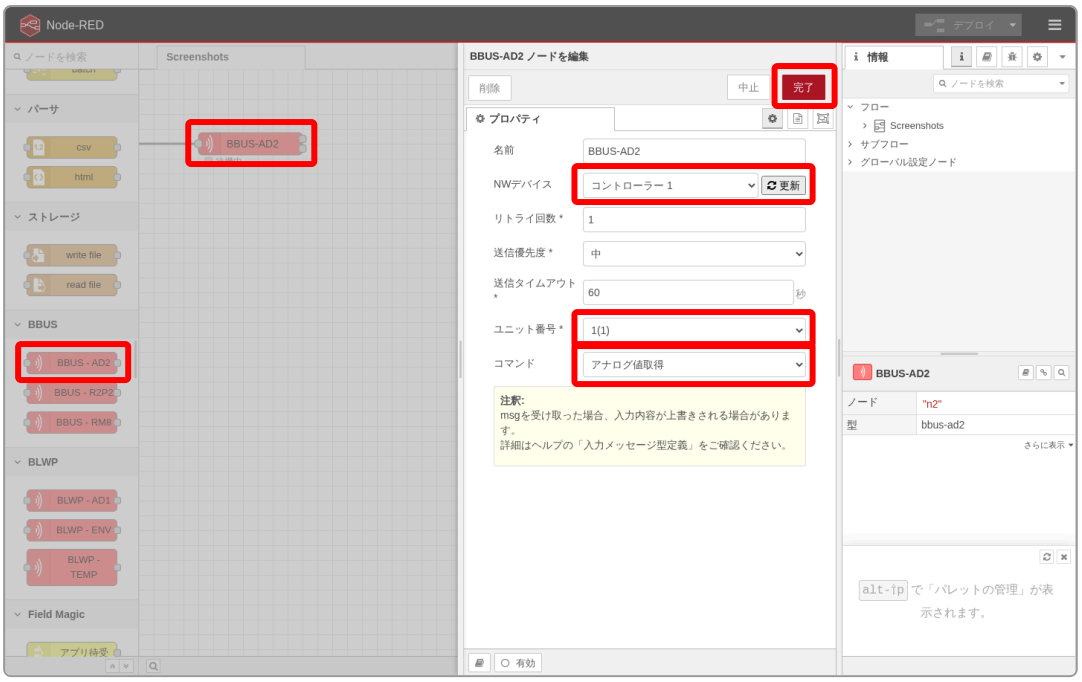
NodeREDの標準ノードである**injectノード**を使用して、1分毎にフローを実行する設定を行います。



1. 左のノード一覧から**injectノード**をドラッグアンドドロップでフロー上に配置します。
2. ノードをダブルクリックして設定画面を開きます。
3. 「名前」欄に「1分毎」を入力します。
4. 「繰り返し」欄で「指定した時間間隔」を選択し、「1分」を設定します。
5. 「完了」ボタンを押して設定を保存します。

5-2-2. ネットワークデバイスからのデータ取得設定

flm-bbusパッケージの**BBUS-AD2ノード**を使用して、ネットワークデバイスからアナログ入力値を取得する設定を行います。

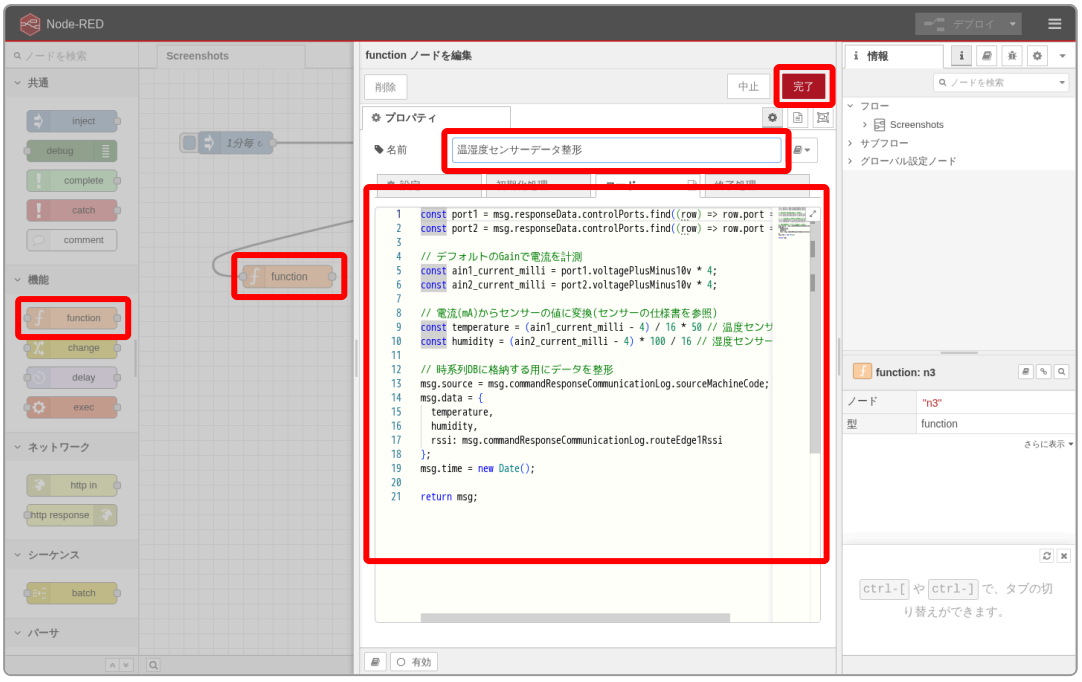


1. 左のノード一覧から**BBUS-AD2ノード**をドラッグアンドドロップでフロー上に配置します。
2. ノードをダブルクリックして設定画面を開きます。
3. 「NWデバイス」欄で、データを取得したいネットワークデバイスを選択します。
4. 「ユニット番号」欄で、データを取得したいBBUS-AD2のユニット番号（ロータリスイッチの数字）を入力します。
5. 「コマンド」欄で、「アナログ値取得」を選択します。
6. 「完了」ボタンを押して設定を保存します。

> 💡 事前にField Magicのネットワークデバイス設定画面で、対象のネットワークデバイスの登録を行っておく必要があります。

5-2-3. 時系列データ保存用のデータ加工設定

NodeREDの標準ノードである**functionノード**を使用して、センサー値の取得結果を時系列データに保存可能なフォーマットに加工する設定を行います。



1. 左のノード一覧から**functionノード**をドラッグアンドドロップでフロー上に配置します。
2. ノードをダブルクリックして設定画面を開きます。
3. 「名前」欄に「温湿度センサーデータ整形」を入力します。
4. 「コード」タブに、以下のコードを入力します。
5. 「完了」ボタンを押して設定を保存します。

```
const port1 = msg.responseData.controlPorts.find((row) => row.port == 1);
const port2 = msg.responseData.controlPorts.find((row) => row.port == 2);

// デフォルトのGainで電流を計測
const ain1_current_milli = port1.voltagePlusMinus10v * 4;
const ain2_current_milli = port2.voltagePlusMinus10v * 4;

// 電流(mA)からセンサーの値に変換
// 4-20mAの入力を各センサーのレンジに変換しています（使用するセンサーの仕様に合わせて変更してください）
const temperature = (ain1_current_milli - 4) / 16 * 50 // 温度センサー: 4-20mA → 0-50℃
const humidity = (ain2_current_milli - 4) * 100 / 16 // 湿度センサー: 4-20mA → 0-100%

// 時系列DBに格納する用にデータを整形
msg.source = msg.commandResponseCommunicationLog.sourceMachineCode;
msg.data = {
 temperature,
 humidity,
 rssi: msg.commandResponseCommunicationLog.routeEdge1Rssi
};

return msg;
```

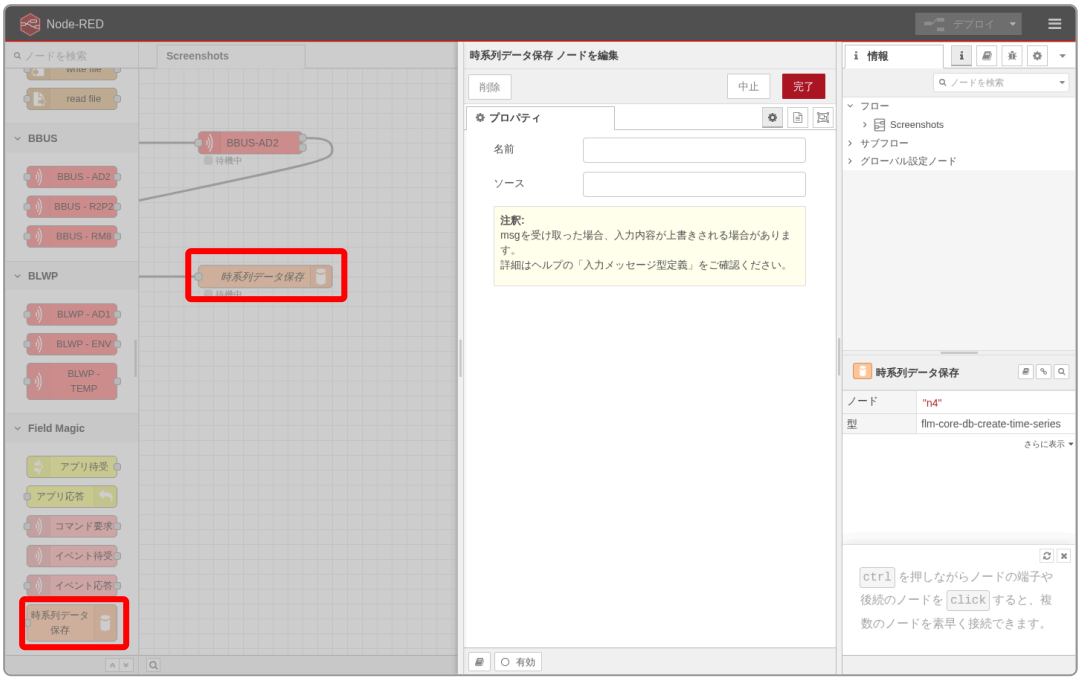
> 💡 上記コード内の **temperature** / **humidity** / **rssi** が、時系列データとして保存されるデータ項目です。

> 💡 前段のノードの出力（msg）に含まれるデータ構造は、接続するノードやデバイスの種類によって異なります。
> 実際の内容はNodeREDの**debugノード**を接続して確認してください。
>
> BBUS-AD2ノードの場合、以下のデータが含まれます。
>
> - `msg.responseData` : デバイスからの応答データです。
> - `msg.commandResponseCommunicationLog` : 通信ログ情報です。送信元の機器コードやRSSIなどが含まれます。

> 💡 BBUS-AD2のゲイン設定や接続方法、センサーの仕様に応じて、コード内の変換式を調整する必要があります。

5-2-4. 時系列データ保存設定

flm-coreパッケージの**時系列データ保存**を使用して、加工したセンサーデータを時系列DBに保存する設定を行います。



1. 左のノード一覧から**時系列データ保存ノード**をドラッグアンドドロップでフロー上に配置します。

※このノードに対し、特別な設定は不要です。

5-3. 動作確認

すべての設定が完了したら、NodeREDの右上にある**デプロイ**ボタンを押してフローを保存・有効化します。

injectノード左のボタンを押すか、1分待つと、フローが開始されます。
フローが開始されると、BBUS-AD2ノードや時系列データ保存ノードのステータスが変化します。
正常に動作していれば、ステータスに「完了しました」や「保存しました」が表示されます。「エラー」と表示されないことを確認してください。

5-4. トラブルシューティング

BBUS-AD2ノードのステータスに「エラー」と表示される

- ネットワークデバイスとの通信に失敗している可能性があります。
  ネットワークデバイスの電源・接続状況・無線チャンネルの設定を確認してください。
  管理画面の疎通確認機能も活用してください。

時系列データ保存ノードのステータスに「エラー」と表示される

- functionノードで加工したデータのフォーマットが不正な可能性があります。
  NodeREDの**debugノード**をfunctionノードの出力に接続し、`msg.source` と `msg.data` の内容を確認してください。

フローを変更したが反映されない

- フローを変更した後は、必ずNodeREDの右上にある**デプロイ**ボタンを押してください。