

ユーザーズマニュアル

MqttPublicManager

発行日 2025年3月31日
公立大学法人会津大学
株式会社東日本計算センター

目次

1. はじめに.....	1
1.1. MqttPublicManager とは.....	1
1.2. 本書の記載範囲.....	2
1.3. 動作環境.....	2
1.4. 前提事項/注意事項	3
1.5. 使用機器.....	3
1.6. 関連資料.....	3
2. 本システムでできること	4
3. 動作手順.....	4
3.1. ディレクトリ構成	4
3.2. DB 接続設定	5
3.3. MQTT 接続ファイルの設定.....	6
3.4. アプリケーションの起動.....	7
3.5. アプリケーションの停止	7
4. エラーメッセージ	7

1. はじめに

1.1. MqttPublicManager とは

複数のロボットやクラウド機能（サーバー機能）がインターネットを介して複数拠点上に分散配置されているような、「クラウドロボティクスインターネット分散システム（The Distributed Cloud Robotics System on the Internet）」（以下、「DCR システム」）のアーキテクチャ開発および実証試験を実施し、DCR システムに係る技術の確立を図る際に一般の方に RDR を利用して貰うように RDR(Robot Data Repository)のコンポーネント化を図りました。

MqttPublicMnanger は RDR に登録されたトピック情報ごとにデータの蓄積や配信を行うシステムです。

本システムは会津大学 産学連携ロボット研究開発支援事業の一環として開発したものです。

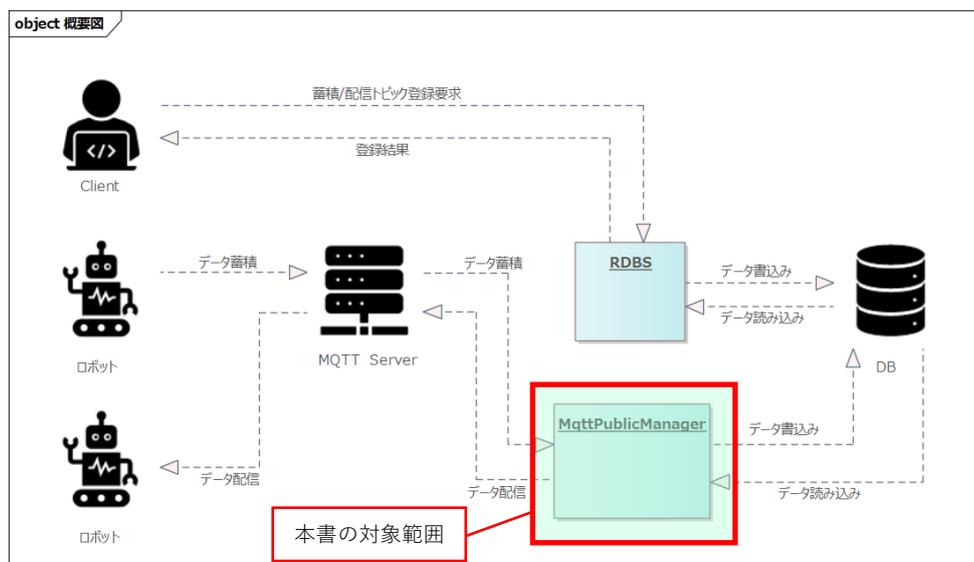


図 1-1 RDR のコンポーネント化の概要図

1.2. 本書の記載範囲

本書の記載範囲は、MqttPublicMnanger を使用するユーザー向けの操作マニュアルになります。

本システムの環境構築に関しては「インストールマニュアル_ MqttPublicMnanger」を参照してください。

1.3. 動作環境

動作環境一覧を表 1-1 に示します。

表 1-1 動作環境一覧

環境		バージョン	補足
OS	Ubuntu	20.04 LTS	-
CPU	Intel(R) Core(TM) i7-10510U CPU @ 1.80GHz 2.30 GHz	-	-
メモリ	8GB 以上推奨	-	-
ストレージ	100GB 以上推奨	-	-
開発言語	Python	3.8 系	-
ミドルウェア	MongoDB	4.4 系	Server Side Public License(SSPL)
依存ライブラリ	paho-mqtt	1.6.1	Eclipse Public License 2.0
	numpy	1.21.6	modified BSD license
	pandas	1.3.5	BSD 3-Clause License
	pymongo	4.3.3	Apache License Version 2.0

1.4. 前提事項/注意事項

導入にあたっての前提ならびに注意事項を表 1-2 に示します。

表 1-2 前提ならびに注意事項

前提事項	(1)インストールマニュアル(別紙)に沿って、システム構築が完了していること (2)対象となる DB サービスが起動していること (3)データを参照する DB および collection が作成済みでいること
注意事項	(1)トピックを登録するには RDBS を起動し、ブラウザからトピック登録を行う必要があります

1.5. 使用機器

本システムで使用する機材を次の表に記載します。

表 1-3 使用機器一覧

No.	使用機器	個数	補足
1	パーソナルコンピューター	1	-

1.6. 関連資料

本システムと関連する資料を次の表に記載します。

表 1-4 関連資料一覧

No.	資料名
1	インストールマニュアル_MqttPublicManager
2	インストールマニュアル_RDBS
3	ユーザーマニュアル_RDBS

2. 本システムでできること

本システムは RDR に登録されたトピック情報を基に、MQTT 通信で publish および subscribe を行い、トピックごとにデータの蓄積や配信を行います。トピックには、受信したデータを RDR に蓄積するものと、RDR に蓄積されたデータを取得し配信するものがあります。

3. 動作手順

3.1. ディレクトリ構成

MqttPublicManager のディレクトリ構成を図 3-1 に示します。

図 3-1 ディレクトリ構成

MqttPublicManager	メインディレクトリ
├── config	設定ファイルを格納するディレクトリ
│ ├── db_settings.json	DB 接続設定ファイル
│ └── mqtt_settings.json	MQTT 接続設定ファイル
├── crt	証明書、秘密鍵を格納するディレクトリ
├── log	ログを格納するディレクトリ
├── mongodb.py	Mongodb 接続クラス
├── mqtt_public_manager_main.py	メイン処理実行クラス
├── rdr_mqtt.py	MQTT 接続クラス
└── requirements.txt	ライブラリパッケージ

3.2.DB 接続設定

DB 設定ファイル仕様として、ファイル形式は JSON、改行コード: LF、文字コードは UTF-8 とします。パラメータを表 3-2、記述例を図 3-1 に示します。

表 3-1 DB 設定ファイル仕様

スペック	フォーマット
ファイル形式	JSON
項目	説明
host	ホスト名または IP アドレスを指定
port	ポート番号を半角数字で指定
db_name	使用するデータベース名を指定
public_db_name	各トピックのデータを蓄積するデータベース名を指定
topic_collection	db_name 内のトピック情報を参照するコレクション名を指定

```
{
  "host": "XXXXXXXXXX",
  "port": XXXXX,
  "db_name": "XXXXXXXXXX",
  "public_db_name": "XXXXXXXXXX",
  "topic_collection": "XXXXXXXXXX"
}
```

図 3-2 db_settings.json の記載内容

3.3.MQTT 接続ファイルの設定

MQTT 接続設定ファイル仕様として、ファイル形式は JSON、改行コード: LF、文字コードは UTF-8 とします。パラメータを表 3-3、記述例を図 3-2 に示します。

表 3-2 mqtt_settings.json の設定項目

スペック	フォーマット
ファイル形式	JSON
項目	説明
host	ホスト名または IP アドレスを指定
port	ポート番号を半角数字で指定
ca_certs	認証局の証明書ファイルパスを指定
certfile	クライアント証明書ファイルパスを指定
keyfile	クライアント証明書の秘密鍵ファイルパスを指定

```
{
  "host": "192.168.1.1",
  "port": 1883,
  "ca_certs": "ca.crt",
  "certfile": "client.crt",
  "keyfile": "client.key"
}
```

図 3-3 mqtt_settings.json の設定箇所

3.4. アプリケーションの起動

ターミナルを起動し、以下のコマンドを実行することでアプリケーションを起動します。

```
$ cd ~/***/MqttPublicManager
$ source venv/bin/activate
(venv)$ mqtt_public_manager_main.py
```

3.5. アプリケーションの停止

アプリケーション実行中にターミナルから ctrl+c を入力するとアプリケーションが停止します。

```
(venv) 実行中...
ctrl + c
```

4. エラーメッセージ

物体認識システム実行中にエラーが発生した場合は以下のメッセージが Log ファイルに出力されます。

表 4-1 エラーメッセージ一覧

No.	項目	説明
1	Unable to connect DB. Please confirm the HOST and PORT.	DB に接続が失敗した場合に出力されるエラーです
2	Unable to MQTT connection	MQTT に接続が失敗した場合に出力されるエラーです
3	Unable to read setting file.	設定ファイルの読み込みが失敗した場合に出力されるエラーです

著作権

本文書の著作権は公立大学法人 会津大学に帰属します。