

インストールマニュアル RDRMapManager

発行日 2024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. 本書について	1
2. インストール環境	1
3. 導入	2
3.1. 準備	2
3.2. MongoDB.....	3
3.3. PostgreSQL	5
3.4. ライブラリのインストール	9

1. 本書について

本書は RDRMapManager の動作環境の構築手順を記載します。本システムの操作に関しては「ユーザーズマニュアル_RDRMapManager」を参照してください。

2. インストール環境

本システムの動作環境を記載します。

依存ライブラリバージョンは本システム検証時点となります。

表 2-1 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	-	-
GPU	-	-	-
メモリ	16GB 以上	-	-
ストレージ(SSD)	512GB 以上	-	-
開発言語	Python	3.8 系	-
ミドルウェア	MongoDB	4.4 系	-
	PostgreSQL	14 系	-
依存ライブラリ	mpmath	1.2.1	-
	numpy	1.22.4	-
	opencv-python	4.5.2.54	-
	paho-mqtt	1.6.1	-
	psycogp2	2.9.3	-
	pymongo	4.1.1	-
	scipy	1.5.2	-
	sympy	1.11.1	-

3. 導入

各種ライブラリ等を導入します。導入するライブラリは本システムの環境構築時点となります。ライブラリによっては最新版もしくは任意の世代までの公開となっている場合があります。その際は最新版を導入してください。

3.1. 準備

- (1) パッケージを更新します。

```
$ sudo apt update
```

- (2) Python の仮想環境機能である `venv` モジュールをインストールします。

```
$ sudo apt-get install python3-venv
```

- (3) `RDRMapManager` を任意の場所に配置し、ディレクトリ内に仮想環境を作成します。

```
$ cd /***/***/RDRMapManager
```

```
$ python3 -m venv venv
```

```
$ ls -l
```

3.2. MongoDB¹

- (1) 公式鍵をインポートします。

```
$ wget -qO - https://www.mongodb.org/static/pgp/server-4.4.asc | sudo apt-key add -
```

- (2) MongoDB の list ファイルを作成します。

```
$ echo "deb https://repo.mongodb.org/apt/ubuntu bionic/mongodb-org/4.4 multiverse" |  
sudo tee /etc/apt/sources.list.d/mongodb-org-4.4.list
```

- (3) パッケージを更新します。

```
$ sudo apt update
```

- (4) インストールをします。本書では 4.4.15 を例にします。

```
$ sudo apt install mongodb-org=4.4.15 mongodb-org-server=4.4.15 mongodb-org-  
shell=4.4.15 mongodb-org-mongos=4.4.15 mongodb-org-tools=4.4.15
```

バージョンアップデートせずに固定したい場合は、以下を実行します。

```
$ echo "mongodb-org hold" | sudo dpkg --set-selections  
$ echo "mongodb-org-server hold" | sudo dpkg --set-selections  
$ echo "mongodb-org-shell hold" | sudo dpkg --set-selections  
$ echo "mongodb-org-mongos hold" | sudo dpkg --set-selections  
$ echo "mongodb-org-tools hold" | sudo dpkg --set-selections
```

- (5) サービスを起動します。

```
$ sudo service mongod start
```

¹ <https://docs.mongodb.com/manual/tutorial/install-mongodb-on-ubuntu/>

(6) サービスの自動起動を有効化します。

```
$ sudo systemctl enable mongod.service
```

(7) MongoDB シェルを起動します。以下のようなメッセージが表示されれば、インストール完了です。

```
$ mongo
MongoDB shell version v4.4.15
connecting to:
mongodb://127.0.0.1:27017/?compressors=disabled&gssapiServiceName=mongodb
:
MongoDB server version: 4.4.15
:
>
```

※：設定ファイルについては、本書では触れませんので、ユーザー様で適切に設定してください。

3.3. PostgreSQL²

- インストール

(1) PostgreSQL 公式の apt リポジトリを/etc/apt/sources.list.d/pgdg.list に追加します。

```
$ sudo sh -c 'echo "deb http://apt.postgresql.org/pub/repos/apt/ focal-pgdg main" >>
/etc/apt/sources.list.d/pgdg.list'
```

(2) 公式鍵をインポートします。

```
$ wget --quiet -O - https://www.postgresql.org/media/keys/ACCC4CF8.asc | sudo apt-key
add -
```

(3) パッケージを更新します。

```
$ sudo apt update
```

(4) インストールをします。

```
$ sudo apt -y install postgresql-14
```

(5) サービスの自動起動を有効化します。

```
$ sudo systemctl enable postgresql
```

(6) サービスを起動します。

```
$ sudo systemctl start postgresql
```

(7) PostgreSQL シェルを起動します。以下のようなメッセージが表示されればインストール完了です。¥q で切断してください。

```
$ sudo -u postgres psql
psql (14.4 (Ubuntu 14.4-1.pgdg20.04+1)、サーバー 10.21 (Ubuntu 10.21-
1.pgdg20.04+1))
"help"でヘルプを表示します。
postgres=# ¥q
```

※：設定ファイルについては、本書では触れませんので、ユーザー様で適切に設定してください。

² <https://www.postgresql.org/download/linux/ubuntu/>

- DB/テーブル作成と設定

- (1) postgres ユーザー(OS ユーザー)のパスワードを設定します。すでに設定済みでしたら、次にお進みください。

```
$ sudo passwd postgres
新しい UNIX パスワードを入力してください:
新しい UNIX パスワードを再入力してください:
```

- (2) postgres ユーザー(PostgreSQL ユーザー)のパスワードを設定します。

```
$ su - postgres
postgres@pcname:~$ psql
psql (14.4 (Ubuntu 14.4-1.pgdg20.04+1)、サーバー 10.21 (Ubuntu 10.21-1.pgdg20.04+1))
"help"でヘルプを表示します。
postgres=# ALTER ROLE postgres WITH PASSWORD '(パスワード);
```

- (3) DB の作成ならびにロール(ユーザー)の作成をします。本書では説明の便宜上、DB は rdr_db と命名します。

```
$ su - postgres
$ postgres@pcname:~$ createdb rdr_db
$ postgres@pcname:~$ psql rdr_db
rdr_db=# CREATE ROLE (ロール名) WITH LOGIN PASSWORD '(パスワード);
```

作成されたことを確認します。確認後、ログアウトします。

```
rdr_db=# \l
名前 | 所有者 | エンコーディング | 照合順序 | Ctype(変換演算子) | アクセス権限
-----+-----+-----+-----+-----+-----
postgres | postgres | UTF8 | ja_JP.UTF-8 | ja_JP.UTF-8 |
rdr_db | postgres | UTF8 | ja_JP.UTF-8 | ja_JP.UTF-8 |
rdr_db=# \q
postgres@pcname:~$ exit
```


(4) DB のオーナーを(3)で作成したユーザーに変更します。

```
$ sudo -u postgres psql
postgres=# ALTER DATABASE "rdr_db" OWNER TO (ロール名);
ALTER DATABASE
postgres=#
```

(5) (3)の方法でデータベース一覧を表示し、所有者が(4)で指定したロール名であることが確認できます。

(6) DB にデータを格納するテーブルを作成するために、rdr_db に切り替えます。

```
postgres=# \c rdr_db
データベース "rdr_db" にユーザー "postgres" として接続しました。
rdr_db=#
```

(7) テーブル env_map_tbl を作成します。すべてワンラインで記述しますが、本書では説明の便宜上データ毎に改行表示します。

```
CREATE TABLE IF NOT EXISTS public.env_map_tbl
(
  robot_id character varying(255) COLLATE pg_catalog."default" NOT NULL,
  type character varying(255) COLLATE pg_catalog."default" NOT NULL,
  space character varying(255) COLLATE pg_catalog."default" NOT NULL,
  "time" timestamp(6) without time zone NOT NULL,
  revision character varying(255) COLLATE pg_catalog."default" NOT NULL,
  path character varying(255) COLLATE pg_catalog."default" NOT NULL,
  location character varying(255) COLLATE pg_catalog."default" NOT NULL,
  lat numeric(9,6),
  lon numeric(9,6),
  azimuth numeric(9,6),
  resolution numeric(9,6) NOT NULL,
  width integer NOT NULL,
  height integer NOT NULL,
  origin_x numeric(9,6) NOT NULL,
  origin_y numeric(9,6) NOT NULL,
  orientation_x numeric(9,6) NOT NULL,
  orientation_y numeric(9,6) NOT NULL,
```

```
orientation_z numeric(9,6) NOT NULL,  
orientation_w numeric(9,6) NOT NULL,  
static_layer_revision character varying(255) COLLATE pg_catalog."default" NOT NULL,  
semi_static_layer_revision character varying(255) COLLATE pg_catalog."default" NOT NULL,  
exclusion_zone_layer_revision character varying(255) COLLATE pg_catalog."default" NOT NULL,  
layout_object_id character varying(255) COLLATE pg_catalog."default" NOT NULL,  
origin_z numeric(9,6),  
CONSTRAINT env_map_tbl_pkey PRIMARY KEY (revision)  
)  
WITH (OIDS = FALSE)  
TABLESPACE pg_default;  
ALTER TABLE IF EXISTS public.env_map_tbl  
OWNER to $ROLL_NAME;
```

(8) 作成したテーブルのオーナーを変更します。

```
rdr_db=# ALTER TABLE public.env_map_tbl OWNER to (ロール名);
```

(9) オーナーが変更されたことを確認し、ログアウトします。

```
rdr_db=# ¥dt  
リレーション一覧  
スキーマ | 名前 | 型 | 所有者  
-----+-----+-----+-----  
public | env_map_tbl | テーブル | (ロール名)  
  
rdr_db=# ¥q  
postgres@pcname:~$ exit
```

作成されたこと、指定した所有者であることを確認します。確認後、ログアウトします。

3.4. ライブラリのインストール

- (1) 仮想環境を起動し、RDRMapManagerに必要なライブラリをインストールします。

```
$ cd /**/**/RDRMapManager
$ source venv/bin/activate
(venv)$ pip install -r requirements.txt
```

```
~/RDRMapManager$ source venv/bin/activate
(venv) ~/RDRMapManager$ pip install -r requirements.txt
Collecting mpmath==1.2.1
  Using cached mpmath-1.2.1-py3-none-any.whl (532 kB)
Collecting numpy==1.22.4
  Using cached numpy-1.22.4-cp38-cp38-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (16.9 MB)
Collecting opencv-python==4.5.2.54
  Using cached opencv_python-4.5.2.54-cp38-cp38-manylinux2014_x86_64.whl (51.0 MB)
Processing /home/.../.cache/pip/wheels/6a/48/01/c895c027e9b9367ec5470fbf371ee56e795a49ac6a19aa4c9f/paho-mqtt-1.6.1-py3-none-any.whl
Processing /home/.../.cache/pip/wheels/f3/dc/e2/b8e0e2142eff7fd680295ecd2d92e3bfb90195523e43da161/psycopg2-2.9.3-cp38-cp38-linux_x86_64.whl
Collecting pymongo==4.1.1
  Downloading pymongo-4.1.1-cp38-cp38-manylinux2014_x86_64.whl (498 kB)
    |████████████████████████████████████████| 498 kB 2.7 MB/s
Collecting scipy==1.5.2
  Using cached scipy-1.5.2-cp38-cp38-manylinux1_x86_64.whl (25.7 MB)
Collecting sympy==1.11.1
  Using cached sympy-1.11.1-py3-none-any.whl (6.5 MB)
Installing collected packages: mpmath, numpy, opencv-python, paho-mqtt, psycopg2, pymongo, scipy, sympy
Successfully installed mpmath-1.2.1 numpy-1.22.4 opencv-python-4.5.2.54 paho-mqtt-1.6.1 psycopg2-2.9.3 pymongo-4.1.1 scipy-1.5.2 sympy-1.11.1
```

図 3-1. ライブラリインストール

- (2) 正常にライブラリがインストールされたか確認します。

```
$ pip list
```

```
(venv) ~/RDRMapManager$ pip list
Package            Version
-----
mpmath              1.2.1
numpy               1.22.4
opencv-python      4.5.2.54
paho-mqtt           1.6.1
pip                 20.0.2
pkg-resources       0.0.0
psycopg2            2.9.3
pymongo             4.1.1
scipy               1.5.2
setuptools          44.0.0
sympy               1.11.1
```

図 3-2. pip list ライブラリ確認

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。