

UNIQ-VIS 설치

작성자: 구연현

2022년 11월 30일

사전준비

- Docker 설치

- <https://docs.docker.com/engine/install/ubuntu/>

- MongoDB 설치

- <https://www.mongodb.com/docs/manual/tutorial/install-mongodb-enterprise-with-docker/>

- Node.js 설치

- <https://nodejs.org/>

- MongoDB 설정

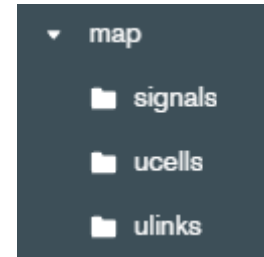
- 공유 디렉토리 설정

MongoDB 설정

- 데이터베이스
 - map
 - link, cell, signal 정보 저장을 위한 데이터베이스
 - 가시화 서버 구동전에 미리 생성하고 필요한 데이터를 추가해야 함
 - simulation_results
 - 시뮬레이션 결과 저장을 위한 데이터베이스
 - 가시화 서버에서 자동으로 생성 함으로 별도로 생성할 필요 없음

MongoDB 설정

- MongoDB Compass 를 이용해서 설치된 MongoDB 에 접속
- 데이터 베이스 생성
 - MAP
- COLLECTION 생성
 - SIGNALS
 - UCELLS
 - ULINKS
- 생성한 COLLECTION 선택 후 Import Data
 - signlas.json, ulinks.json, ucells.json



Mongo DB 설정

- 자동화 되어 있지 않아서 수동으로 설정 필요
- Compass 어플리케이션을 설치 사용하면 편리함 (compass 기준으로 설명)
 - 데이터베이스 생성
 - map 데이터베이스 생성
 - map 데이터베이스 선택 후
 - ucells, ulinks, signals 세 개의 Collection 생성
 - signals.json, ucells.json, ulinks.json 파일을 각각의 Collection 에 import 시킨다.
 - 파일위치: 시각화 서버의 /home/ubuntu/uniq-sim/map
 - 해당 Collection 선택 후
 - Compass 의 메뉴 > Collection > Import Data

데이터 Import

Import To Collection map.ulinks

Select File

BROWSE

Select Input File Type

JSON

CSV

Options

☐ Stop on errors

CANCEL

IMPORT

3. import 하고자
하는 파일 선택

2. 메뉴 > Collection > Import Data 선택

1. 마우스로 선택

Local

11 DBS 39 COLLECTIONS

☆ FAVORITE

HOST

localhost:27017

CLUSTER

Standalone

EDITION

MongoDB 4.0.10 Community

Filter your data

> admin

> chatdb

> config

> lms

> local

> map

signals

ucells

ulinks

+

> _MONGOSH

데이터 Import 화면

Local

11 DBS 39 COLLECTIONS

☆ FAVORITE

HOST

localhost:27017

CLUSTER

Standalone

EDITION

MongoDB 4.0.10 Community

Filter your data

> admin

> chatdb

> config

> lms

> local

> map

signals

ucells

ulinks

+

> _MONGOSH

map.ucells
Documents

map.ucells

DOCUMENTS 524.2k 187.4MB 375B INDEXES 2 9.0MB 4.5MB

Documents

Aggregations

Schema

Explain Plan

Indexes

Validation

FILTER { field: 'value' }

OPTIONS

FIND

RESET

↺

⋮

ADD DATA

↑

VIEW

≡

{ }

⌘

Displaying documents 1 - 20 of 524204

< >

REFRESH

```
_id: ObjectId("5f9a43d53796f9b480f8c426")
type: "Feature"
> properties: Object
> geometry: Object
```

```
_id: ObjectId("5f9a43d53796f9b480f8c427")
type: "Feature"
> properties: Object
> geometry: Object
```

```
_id: ObjectId("5f9a43d53796f9b480f8c428")
type: "Feature"
> properties: Object
> geometry: Object
```

```
_id: ObjectId("5f9a43d53796f9b480f8c429")
type: "Feature"
> properties: Object
> geometry: Object
```

```
_id: ObjectId("5f9a43d53796f9b480f8c42a")
type: "Feature"
> properties: Object
> geometry: Object
```

데이터베이스

Collections

MonboDB

8 DBS28 COLLECTIONS

★ FAVORITE

HOST
localhost:27017

CLUSTER
Standalone

EDITION
MongoDB 5.0.6 Community

My Queries

Databases

Filter your data

signals

ucells

ulinks

+>_MONGOSH

Indexes
map.ulinks

map.ulinks

151.2k2
DOCUMENTSINDEXES

DocumentsAggregationsSchemaExplain PlanIndexesValidation

RefreshCreate Index

Name and Definition	Type	Size	Usage	Properties
> _id_	REGULAR	1.5 MB	3 (since Thu Nov 17 2022)	UNIQUE
> geometry_2dsphere	GEOSPATIAL	4.1 MB	1688 (since Thu Nov 17 2022)	

JSON 임포트 후 반드시 인덱스 설정

공유 디렉토리 설정

- 특정 디렉토리에 uniq-sim.zip 파일을 압축 해제
- 디렉토리 구조

/home/ubuntu/uniq-sim/

- └─ data (교통 시뮬레이션 설정, 신호 최적화 설정, 신호 최적화 결과)
- └─ output (교통 시뮬레이션 결과)
- └─ routes (수요 파일)

- 압축해제 후 /home/ubuntu/uniq-sim/ 은 가시화 서버 소스 디렉토리의 config.js 파일 수정 해야 함

시나리오 및 수요파일 확인

- 시뮬레이션을 위한 수요 및 시나리오 파일 확인
 - 수요파일
 - yuseonggu.rou.xml → 유성구 시뮬레이션에 필요
 - doan.rou.xml → 도안 시뮬레이션에 필요
 - 시나리오 파일
 - scenario_dj_doan.zip → 도안 시뮬레이션에 필요
- 신호최적화를 위한 시나리오 파일 확인
 - 신호 최적화에서는 수요 파일이 시나리오 파일 내에 포함되어 있음
 - scenario_doan.zip → 도안 시나리오
 - scenario_cdd3.zip → 연구단지 시나리오

웹 UI 빌드

- 웹 클라이언트 소스 디렉토리로 이동
- 빌드
 - `npm run build`
- 생성된 결과물은 가시화 서버의 `public/` 디렉토리에 생성됨

가시화 서버 실행

- 가시화 서버의 소스 디렉토리로 이동
- 라이브러리 설치
 - `npm install`
- 가시화 서버 실행
 - `node ./bin/www` (싱글 프로세스 실행)
 - 또는
 - `npx pm2 start ./bin/www` (PM2 를 이용한 실행)

실행 확인

- <http://localhost:8080> 접속 후 UI 확인

Backup Slide

- 교통 시뮬레이션 결과 디렉토리
- 신호 최적화 결과 디렉토리

교통 시뮬레이션 결과 디렉토리 구조

/uniq_sim/output

- SIMU_202011_00700
 - 2020_dj_sample-PeriodicOutput.csv
 - bar-data.json
 - grid-data.json
 - histogram-data.json
 - pie-data.json
- SIMU_202211_00184
 - SIMU_202211_00184_PeriodicOutput.csv
 - bar-data.json
 - histogram-data.json
 - pie-data.json
 - progress.txt
- SIMU_202211_00828

신호 최적화 결과 디렉토리 구조

신호 최적화 결과는 /uniq_sim/data 에 생성된다.

