

Are you ready?



Software Tool Time



Unstructured Data Warehouse

- Tutor

- Jiheon Choi

- Objectives

- Introduction about Pandas, Node.js
 - How to implement data warehouse with unstructured data
 - Explain code & Project demo

- Preparation

- Python Pandas
 - Node.js & MongoDB
 - Visual Studio Code

Case Study: Autonomous Vehicle Data

- 자율주행 알고리즘 검증을 위한 시뮬레이션 소프트웨어에 대한 관심 증가
- 자율주행 차량은 장착된 많은 센서를 통해 주변 상황을 인식하고, 생성된 데이터 분석을 통해 주어진 상황에 대해 적절한 판단을 도출
- 이러한 자율주행 차량의 특성 상 실제 도로에서 주행을 하기 위해서는 사전에 탑승자, 보행자 및 다른 차량의 안전을 고려하는 엄격한 검증 절차 필요

Choose a Database system

- 데이터 적재를 위해 적절한 데이터베이스 시스템 선택 필요
- 서로 다른 데이터베이스 엔진을 대상으로 작업부하(Workload)를 변경하며, 질의-응답 성능 측정 (e.g. HBase, Cassandra, MySQL)
- 실험 결과로 HBase가 Read, Update, Insert Operation들에 대해 상대적으로 더 나은 성능을 보였으나, 적재하는 데이터의 특성과 질의 형태를 고려했을 때는 MongoDB가 적합 (key-value)



mongoDB®

Related Works

- 비정형 데이터 기반의 데이터 관리는 자율주행 차량 관련 중요 연구 주제
- 기존 연구에서는 정형 데이터를 기본으로 하며, 비정형 데이터를 입력받아 정형화하는 작업을 거쳐 적재하는 연구가 다수 (NoSQL 데이터베이스를 활용한 실험 및 연구가 활발)
- 이에 따라 비정형 데이터 자체를 적재하고 효과적으로 질의할 수 있는 연구 필요

Autonomous Vehicle Dataset

- 개발한 프로토타입을 실전적 상황에서 사용 가능하도록 연구 목적으로 공개된 데이터 조사 및 수집
- 구현에 사용된 데이터는 자율주행 분야 데이터 분석에 자주 활용되는 NGSIM US-101 데이터

	NGSIM US-101	KITTI	nuScenes	A2D2
Sensors	Camera*4 (Roadside)	Camera*4 Lidar GPS/IMU	Camera*6 Rader*5 Lidar GPS/IMU	Camera*6 Lidar*5
Location	freeway	Urban, one city	Urban, two cities	Urban, highways, country roads
Labels	Trajectory, vehicle type, size	3D bbox, 2D bbox, type	Localization, Trajectory, 3D bbox, 2D bbox,	3D bbox
Data size	1.66 GB	65 GB	292.78 GB	2.3 TB



ETL Process for unstructured data

- 연구자가 비정형 데이터를 편리한 화면 인터페이스에서 효과적으로 질의 가능한 시스템 구현
- 다양한 비정형 데이터(특별히 시계열 데이터)에 대한 ETL 프로세스를 진행하기 위해, 다양한 파일 형태의 데이터를 분석하여 데이터의 행과 열을 포함한 파일의 메타데이터를 적재하며, 이를 기반으로 효과적인 질의가 가능한 시스템 설계 제안

ETL Process for unstructured data

추출 (Extract)



1. Python Pandas 활용
2. 다양한 종류의 파일 입력
(e.g. Excel, TCSV)
3. 행과 열, 파일의 메타 데이터
(e.g. 파일 이름과 크기)를 추출



변환 (Transform)

1. 칼럼 값과 파일의 메타 데이터
정보에 대한 객체를 병합한 뒤
JSON Object 형태로 변환
2. MongoDB 데이터베이스 삽입
질의를 수행하기 적합한 형태로
만들기 위함

```
{  
  "file_name": "ngsim_us-101.csv",  
  "file_size": "1.66 GB",  
  "columns": {  
    "Vehicle_ID": "Integer",  
    "Frame_ID": "Float",  
    ...  
  }  
}
```



적재 (Load)

1. 파일의 메타 데이터 정보에서
획득한 칼럼명과 타입을
사용해 스키마 생성
2. 생성된 스키마 모델로
전체 데이터 행을 삽입

ETL Process: Memory Issue

문제점	해결 방법
최초에는 데이터 파일에서 행과 열 데이터를 추출하기 위해 Node.js 기반으로 구현했으나, 대용량 파일을 읽어오는데 메모리 이슈가 존재	이를 해결하기 위해 데이터 분석에 자주 활용되는 Python Pandas 로 데이터를 읽어 온 뒤, 데이터베이스 스키마를 자동으로 생성하는 부분을 Python으로 개발

```
import pandas as pd

df = pd.read_csv('./datasets/ngsim-data.csv')

columns = df.columns.values.tolist() # get columns name

for index, row in df.iterrows():
    print(index, row) # print the index of data including row content.
```

ETL Process: MongoDB Issue

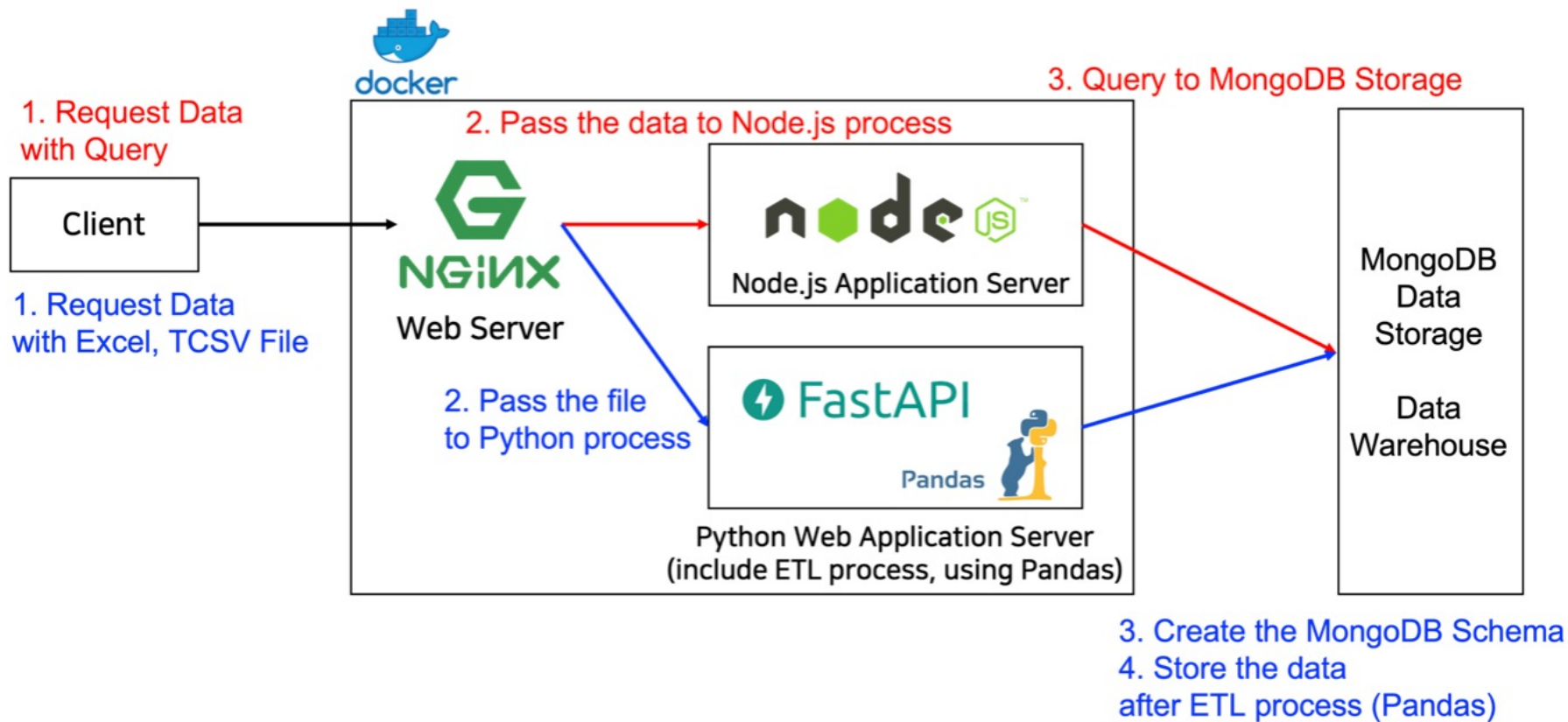
문제점	해결 방법
<ol style="list-style-type: none">1. MongoDB 질의를 사용하는 전형적인 Node.js 애플리케이션은 “mongoose” 모듈을 활용하여 개발자가 정의한 코드 상의 스키마 모델을 활용하여 질의하는 구조2. 하지만, Python 애플리케이션을 통해 이미 적재된 데이터를 Node.js 애플리케이션을 통해 질의 가능하도록 구현해야 하는 상황	<p>'mongodb', 'bson' 모듈을 활용하여 데이터베이스에 적재된 Collection, Schema 등의 정보를 가져오는 로직 구현</p>

```
const { Schema, model } = require("mongoose");  
const NgsimSchema = new Schema({  
  Vehicle_ID: Number,  
  Frame_ID: Number,  
  Total_Frames: Number,  
  Global_Time: Number,
```



```
db.listCollections({ name: { $not: { $regex: /system.+/ } } }).toArray()  
  .then((collections) => {  
    let proms = [];  
    collections.forEach(collection => {  
      proms.push(dataAccessAdapter.ConnectToCollection(  
        collection.name,
```

System Architecture



Frontend Interface

Driving Data Warehouse

Search for collections

admin

config

driving-data

ngsim

ngsim-partial

local

ngsim

Simple

Key

:

Value

String

Find

Add

Export

Reload

driving-data → ngsim

Displaying 1 - 10 of 11850526 documents

1 2 3 4 5 ... 1185053 > Goto

```
_id:ObjectId("6282b40e8b2426f7a5b38196")
Vehicle_ID:515
Frame_ID:2330
Total_Frames:1123
Global_Time:1118848075000
Local_X:30.034
Local_Y:188.062
Global_X:6451203.729
Global_Y:1873252.549
v_length:13
v_Width:6.9
v_Class:2
v_Vel:23.31
v_Acc:2.05
Lane_ID:3
O_Zone:NaN
D_Zone:NaN
Int_ID:NaN
Section_ID:NaN
Direction:NaN
Movement:NaN
Preceding:500
Following:523
Space_Headway:119.1
Time_Headway:5.11
Location:"us-101"
```

```
_id:ObjectId("6282b40e8b2426f7a5b38197")
Vehicle_ID:515
Frame_ID:2330
Total_Frames:1123
```



Software Tool Time (소프트웨어 툴 타임)

(CC-BY-NC 4.0) Jiheon Choi and Ajou University

Visit “Software Tool Time” channel in YouTube : <http://goo.gl/remxrw>

This video was supported by the Khronos Group

All product names, trademarks, and/or company names are used solely for identification and belong to their respective owners.

“Software Tool Time” video is licensed to the public under a Creative Commons Attribution 4.0 License (<https://creativecommons.org/licenses/by/4.0/>)

Twin Musicom’s African Drums (Sting) is licensed to the to the public under a Creative Commons Attribution 4.0 License (Artist: <http://www.twinmusicom.org/>)