Development of Extract-Transform-Load Processes and Visualization of Data Consistency in DustBoy Measurement System

Nat Weerawan[1] and Pruet Boonma[2]

[1]Master's Degree Program in Data Science, Chiang Mai University, Chiang Mai, Thailand
[2]Faculty of Engineering, Chiang Mai University, Chiang Mai, Thailand

nat_w@cmu.ac.th

Abstract. In this independent study, the extract-transform-load (ETL) system is designed and implemented to extract and transform large scale data from heterogeneous data sources and load into data warehouse efficiently, for ease of further usage. In addition, the visualization system conceptualizes the consistency of the data transmission in the DustBoy measuring system in order to demonstrate researchers or users who in the need of convenient analyzed information for analyzing purposes and debugging problems. Furthermore, the system will assist on decision making reference for maintenance each sensor device with a highly efficient approach.

Keywords: ETL, IoT, Data Visualization, DustBoy Measurement System

1 Introduction

Nowadays, the Internet of Things (IoT) technology has widely adopted[1] [2], coupled with the emergence of the Maker Movement [3] that has given rise to a culture of creating things and equipment that is more sophisticated to use. And today's technology also creates technology that are more convenient for the manufacturing things, such as 3D printer technology. With these tools and opensource software manufacturing does not need to rely on industrial factories to produce mass production. Technology developers can produce or create your own products[4]

Climate Change Data Center Chiang Mai University (CCDC) is an organization that has developed a climate change database system. It has also developed a DustBoy Weather Sensor [5], a sensor device capable of measuring PM2.5, PM10, temperature and humidity levels. By using internet of things technology which has been installed in more than 400 locations across Southeast Asia and set to expand to 2,000 - 3,000 locations throughout Thailand [6]

As a result, the authors has the opportunity to participate in a research project with the research team of the Climate Change Information Center. Chiang Mai University, the study results show that there was a problem with the transmission of the weather

sensor's data and the problem of uneven transmission of the sensor resulting in the loss of important information over time.
The author recognized that the intermittent transmission of uneven data was a major problem but difficult to detect because of the data moderator will only know the problem if it has been reported that the system has not been sent for a long time. The author is interested in using the ETL process [8][9][10][11] to prepare the data before use, as the data is available in multiple sources and in a variety of formats for ease of use. Also, visualization can be used to show consistency of data to make it easier to see trends in data. Because of the large amount of information and difficult to check consistency.

2 The proposed system

The system is separated into four parts: CSV conversion system (csv2line), Data loading to ETL Engine, ETL Engine and database persistent system, respectively. The systems is developed using Python and NodeJS language. The systems are illustrated in Fig. 1

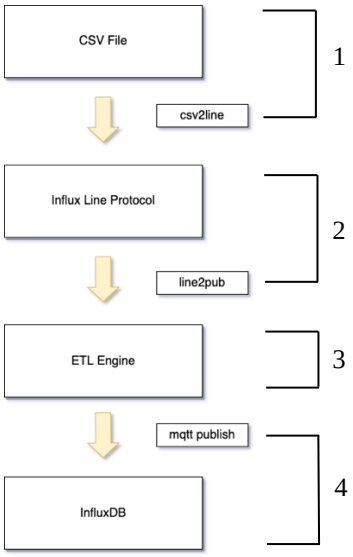

Fig. 1. The systems overview

2.1 CSV2Line System

CSV2Line system is designed to convert a csv file to a newline separated text in an influx line protocol data format that shown in **Fig. 2.**

```
dustboy,nickname=N-001 pm_10=82 1465839830100400200
   |    --------------------- ---------------  |
   |              |              |              |
   |              |              |              |
+-----------+--------+-+---------+-+----------+
|measurement|,tag_set| |field_set| |timestamp|
+-----------+--------+-+---------+-+----------+
```

Fig. 2. Influx line protocol data format

2.2 Line2Pub System

Line2Pub system is designed to sending the converted data in influx line protocol format to the ETL System. The system is written in Python as a cli program that shown in **Fig. 3.**

Fig. 3. Line2Pub command and parameters

2.3 ETL System

The ETL system is written as a dockerized NodeJS application that user can put ETL Configuration to the system by write a single NodeJS file to define fields need to be converted which is an example file shown in **Fig. 4.**

```
const f = {
  fields: [
    { field: 'd_pm10', as: 'pm10' },
    { field: 'd_pm2_5', as: 'pm2_5' },
  ],
}
```

Fig. 4. An example of configuration file in the ETL system

3 Experimental Result

The evaluation of the systems is using a success rate in each conversion steps and speed of the conversion in CSV2Line, Line2Pub, ETL Engine, respectively.

3.1 CSV2Line

The evaluation in CSV2Line system is using a success rate of converted data from csv file to an influx line protocol format. The success rate of the results is 100% with no missing data after CSV2Line system. The results are shown in **Table 1.**

Table 1. The result of processed lines of data when use CSV2Line system

Sensor Model	Processed Lines of data	Converted Data (Lines)	Files	Success Rate (%)
Model-IV	4,764,698	4,764,680	18	100
Model-N	2,755,091	2,755,083	8	100
Model-N-NB-IoT	6,697,226	6,697,220	6	100
Model-PRO-1	25,818,648	25,818,640	8	100
Model-PRO-2	32,981,268	35,746,619	7	100
Model-T	1,139,043	1,139,034	9	100
	76,921,332	76,921,276	56	100

3.2 Line2Pub

The evaluation in Line2Pub system is using a success rate of publishing data from Line2Pub system to the ETL system. The success rate of the results is 100% with no missing data after Line2Pub system. The results are shown in **Table 2.**

Table 2. The result of processed lines of data when use Line2Pub system

Sensor Model	Published (Lines)	Received (Lines)	Success Rate (%)
Model-IV	4,764,698	4,764,698	100
Model-N	2,755,091	2,755,091	100
Model-N-NB-IoT	6,697,226	6,697,226	100
Model-PRO-1	25,818,648	25,818,648	100
Model-PRO-2	32,981,268	32,981,268	100
Model-T	1,139,043	1,139,043	100

3.3 ETL Engine

The evaluation in ETL engine is using a success rate and average of an experimental data rate in messages per seconds. The invalid data from source has been removed from ETL Engine are less than 0.25% and the average date rate of experiment data rate is 4718.5 msg/s.The results are shown in **Table 3.** And **Table 4.**, respectively.

Table 3. The result of processed lines of data when use ETL system

Sensor Model	Processed Data (Lines)	Removed Data (Lines)	Changes (%)
Model-IV	4,764,698	19,793	0.0724
Model-N	2,755,091	155,640	0.4307
Model-N-NB-IoT	6,697,226	52,264	0.107
Model-PRO-1	25,818,648	1,265,652	0.3747
Model-PRO-2	35,746,626	871,734	0.1254
Model-T	1,139,043	49,482	0.3951
Summary	**76,921,332**	**2,414,565**	0.25

Table 4. The result of processed lines of data when use ETL system

Sensor Model	Average 1st experimental data rate(msg/s)	Average 2nd experimental data rate (msg/s)
Model-IV	4,674	4,650
Model-N	4,674	4,674
Model-N-NB-IoT	4,701	4,657
Model-PRO-1	4,801	4,807
Model-PRO-2	4,650	4,652
Model-T	4,855	4,827
Average data rate (msg/s)	4,725.83	4,711.17

3.4 Data Visualization

Data in each DustBoy Measurement System shows the data rate per hour. The researcher uses scatter plot to visualize the data it will illustrate the consistency and data rates that shown in **Fig. 5.**

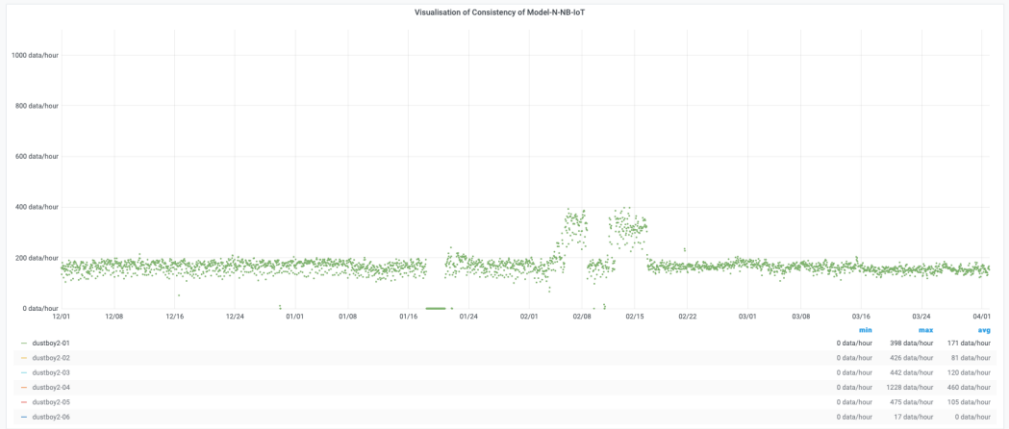

Fig. 5. An example of data visualization in the system

4 Conclusion

From the overall system design that is split into subsystems This gives us a system that is highly flexible and can easily increase the processing power in parallel programming architecture. A measure of the efficiency of the system developed in this research is measured by measuring the transmission rate in the system data extract – transform - load. And the consistency of the data was displayed in the DustBoy Measurement System to help see the frequency of each sensor's data transmission. Moreover, the developed system is also compatible with the original DustBoy Measurement System. Therefore, it can be said that this research is a very practical research because it can be used to connect and enhance the original system as well.

References

1. S. Ziegler, "Considerations on IPv6 scalability for the Internet of Things — Towards an intergalactic Internet," 2017 Global Internet of Things Summit (GIoTS), Geneva, 2017, pp. 1-4.
2. G. Tanganelli, C. Vallati and E. Mingozzi, "Rapid Prototyping of IoT Solutions: A Developer's Perspective," in IEEE Internet Computing, vol. 23, no. 4, pp. 43-52, 1 July-Aug. 2019.
3. H. Cha, S. Lee and H. J. Kim, "Development and Deployment of ICT DIY Standards in Support of Internet of Things: A Qualitative Approach," 2016 IEEE 18th International Conference on High Performance Computing and Communications; IEEE 14th International Conference on Smart City; IEEE 2nd International Conference on Data Science and Systems (HPCC/SmartCity/DSS), Sydney, NSW, 2016, pp. 805-810.

4. 5 REASONS WHY THE MAKER MOVEMENT WILL DRIVE IOT, https://www.digital-newsasia.com/insights/5-reasons-why-the-maker-movement-will-drive-iot, last accessed 2019/03/13
5. What is DustBoy, https://www.cmuccdc.org/aboutus, last accessed 2019/10/30
6. DustBoy Measurement System, https://www.cmuccdc.org/newsdetail/66, last accessed, 2019/02/26
7. Extract, transform, load, https://en.wikipedia.org/wiki/Extract,_transform,_load, last accessed, 2019/02/26
8. H. Agrawal, G. Chafle, S. Goyal, S. Mittal and S. Mukherjea, "An Enhanced Extract-Transform-Load System for Migrating Data in Telecom Billing," 2008 IEEE 24th International Conference on Data Engineering, Cancun, 2008, pp. 1277-1286, doi: 10.1109/ICDE.2008.4497537.
9. Y. S. Chang, K. Lin, Y. Tsai, Y. Zeng and C. Hung, "Big data platform for air quality analysis and prediction," 2018 27th Wireless and Optical Communication Conference (WOCC), Hualien, 2018, pp. 1-3, doi: 10.1109/WOCC.2018.8372743.
10. A Comparative Review of Data Warehousing ETL Tools with New Trends and Industry Insight," 2017 IEEE 7th International Advance Computing Conference (IACC), Hyderabad, 2017, pp. 943-948, doi: 10.1109/IACC.2017.0192.
11. H. Agrawal, G. Chafle, S. Goyal, S. Mittal and S. Mukherjea, "An Enhanced Extract-Transform-Load System for Migrating Data in Telecom Billing," 2008 IEEE 24th International Conference on Data Engineering, Cancun, 2008, pp. 1277-1286, doi: 10.1109/ICDE.2008.4497537 .