Big data collection procedure for on-site monitoring system of smart community with PV microgrid

Manote Tonsing and Worajit Setthapun*

Asian Development College for Community Economy and Technology, Chiang Mai Rajabhat University 50300, Thailand

***Corresponding author's email**: worajit@cmru.ac.th

Received: 26/04/2019, Accepted: 07/05/2019

Abstract

Energy data such as energy consumption from buildings and energy production from distributed generations are very important to determine the optimal sizing, design and configuration of renewable energy-based microgrid systems for a smart community. The main goal of this research is to develop an energy data collection and processing system for the large and complex energy data set generated by a PV-based micro grid powering a Smart Community. Data collected include direct and indirect energy data. Direct energy data for this work are direct energy data which covers energy consumption from buildings located in the model Smart Community, and energy generation data from the PV systems. Indirect energy data include data that affect energy consumption, such as; water usage, indoor temperature, humidity and waste generation. The data collected from sensors installed in the buildings were then processed through three steps: capture, verification and arrangement. Approximately 1,800 data files per month were processed and each data file has the maximum of 86,400 data records depending on the data category and collection interval. The installed sensors in real buildings faced several challenges such interruptions of data transmission due to blackouts, and sensor malfunctions from animals and insects tripping the devices. The data verification was demnostrated to be a very important part to screen the usable data and reject the bad ones. The processed data were then imported into the energy data management system database. The database was developed with MySQL program to systematically grouped and arranged the data in their category. The MySQL database could be integrated with other tools to conveniently manage and report large quantity of data. This big energy data collection and processing procedure, with an easy-to-understand reporting format can be applied to any energy data management system for any real-world, although small community.

Keywords:

Community, microgrid, energy database.

1. Introduction

At present, the demand for energy is continuously increasing due to growth in population, economy and rapid development of technologies [1]. The main energy sources are mostly from fossil fuels but which are now decreasing [2]. There is a need to develop more efficient ways to use alternative renewable energy sources. Production of electricity from solar, wind, biomass and hydro are focusing on distributed generation systems. These distributed renewable energy systems can produce stable and sustainable power for small communities through a micro-grid system. A microgrid system is a small power network that distribute electricity from various distributed generations to the different loads connected to the network. The loads can also receive power from the main utility if the power sources from the renewables connected to the micro-grid are not sufficient [3]. Microgrid systems have been continuously researched and developed especially in Europe, United States of America, Japan and Canada with the goal of increasing further the efficiency of distribution systems. There are now many micro-grid installations done for demonstrations and evaluations in these countries [4]. These microgrid

systems are composed of small distributed generations; various loads; and information and communication technology, energy storage and automatic control systems. The components are integrated comparably to the main power grid. In addition, the current technologies are smaller and more affordable than the past which has allowed for more variety of distributed generations to be installed for power distribution.

The present microgrid system technologies are more advanced and complex, and have more energy data resources that need to be managed. Use of electronic devices to measure and collect energy data are very important components of such micro-grid systems. Micro-grid systems using such electronic devices are called Smart Microgrid. Electronic devices now have very important roles in our daily lives. The word "Smart" has also now been widely used such as; Smart Device, Smart Grid, Smart Home, Smart Network, Smart Intelligent Transportation etc. RFID sensors are being embedded into the devices for identification and act as the brain of the devices. The devices can then connect to world via internet. The concept was created so all devices can communicate together through sensors. Therefore, the smart devices are able to connect to internet and other devices with sensors for data collection. The data from the electronic devices can be collected and devices can communicate with each other via internet through the Internet of Things (IoT) [5]. However, the data collected from all the device sensors are large and complex with and without structure. This create the issue during data compiling, data analysis and data usage.

The ultimate goal of this work is to develop Smart Community with PV microgrid. Applying smart grid for the efficient energy management in the community will be the fundamental structure for the smart community. Hence, data directly and indirectly related to energy production and consumption must be collected and analyzed. With proper data analysis, the data can determine ways to manage, control and optimize the energy system and smart community system. In this work, the Smart Community is the model community with sustainable living with self-energy production, energy efficient housing, community business, and organic agriculture. The Smart Community location for this work is in Chiang Mai Rajabhat University, Mae Rim Campus, Thailand. There are numerous data that is directly and indirectly related to energy in the community such as water consumption, waste production, community environment, etc. The energy data collected will be large and quite complex. Therefore, the data must be collected and sorted. The data with and without structure will be transformed to the same format to facilitate data analytics [6]. This procedure of data collection and Big Data management is the key to determine the relationship between PV microgrid systems and the Smart Community. With the understanding of the community data, the community resources can be efficiently managed to achieve sustainable consumption and production.

2. Methodology

2.1 PV Microgrid

In this research, a DC microgrid and an AC microgrid were installed to supply electricity for the model Smart Community (Fig. 1). The AC microgrid generates DC power from 25 kW PV system and then converted to AC by an inverter and then distributed to AC loads. The DC microgrid generated electricity from a 25.5 kW PV system which then power the DC loads. There was a 100kWh battery bank to which the AC and DC microgrids were connected, including a 40-kW generator. AC and DC electrical loads included electrical appliances found in the 12 buildings of the model Smart Community. The buildings included 6 houses, a minimart, a coffee shop, a restaurant, an office, and two battery bank buildings. The electrical appliances were grouped into four categories: 1) Cooling load such as air conditioner, refrigerator and freezer; 2) Heating load such as microwave, shower water heater, hot water pot, electric plates, and coffee maker; 3) Lighting such as light bulbs and lamp; and 4) Entertainment and office appliances/equipment load such as television, copy machine, printer, projector, computer,

and stereo. The energy consumption from the PV microgrid and the loads in each building were monitored and analyzed. Analyses of the electrical load curves or patterns is required to design an appropriate and efficient power generation and distribution system for the model smart community.

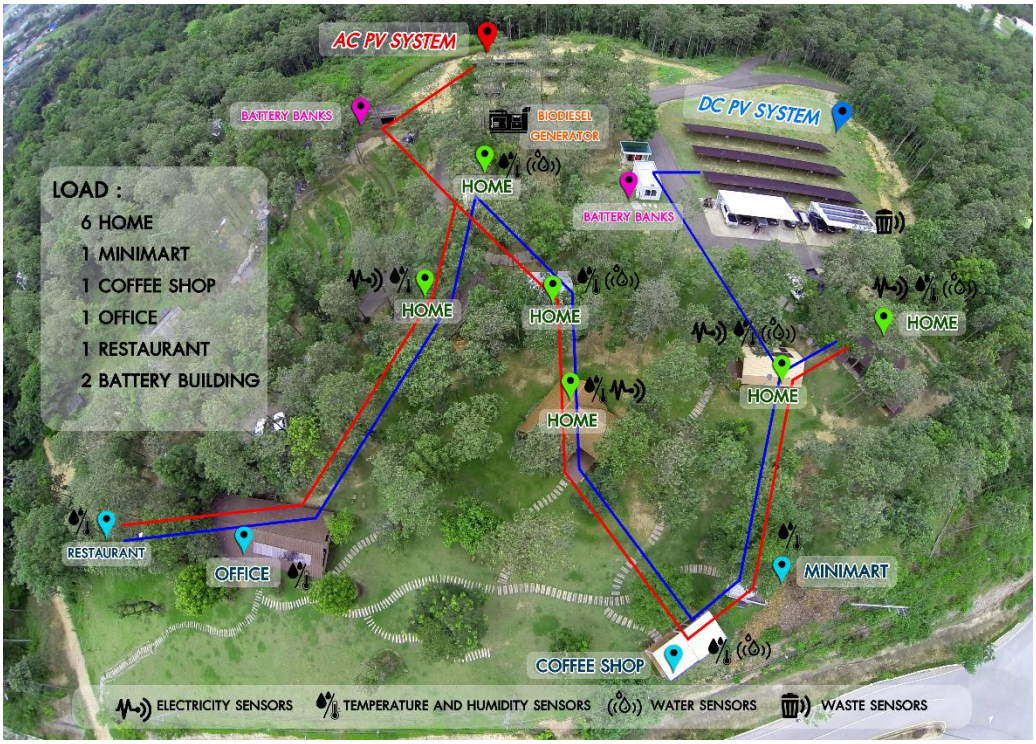


Fig. 1 AC/DC microgrid diagram for the model smart community.

2.2 Data collection process

To study electrification of a model smart community through smart grid, numerous data, including both energy and non-energy data need to be collected. The data categories (as shown on Table 1) include power consumption, temperature, humidity, water usage and waste generation from the building. The capability for real-time data monitoring with sensors installed in each building of the Smart Community was one important criteria is selecting which data to be collected and monitored. Data recorded were becoming highly variable with the increasing amount of data collected.

The data were collected daily and stored as comma separated value (CSV) files. The is the type of text file where the data are stored in a table form with comma to separate the column. The large amount of data recorded were grouped and inputted into the CSV files using the different categories shown on Table 1. The data collection system arranged the CSV files from each category to have the same format. The appropriate data storage format and categories were defined in order for efficient use and analysis of the data.

Table 1 Data collection categories and collection interval.

Data	Unit	Data collection interval
Power Consumption	kW	1 second
Temperature	degree Celsius (°C)	2 seconds
Humidity	percent (%)	2 seconds
Water	Liter (L)	1 second when in use
Garbage	kg	5 minutes

The data collection process is shown in the flow chart below (Fig. 2). The procedure includes data capture, verification, and arrangement to attain the dataset format that can be used in database design.

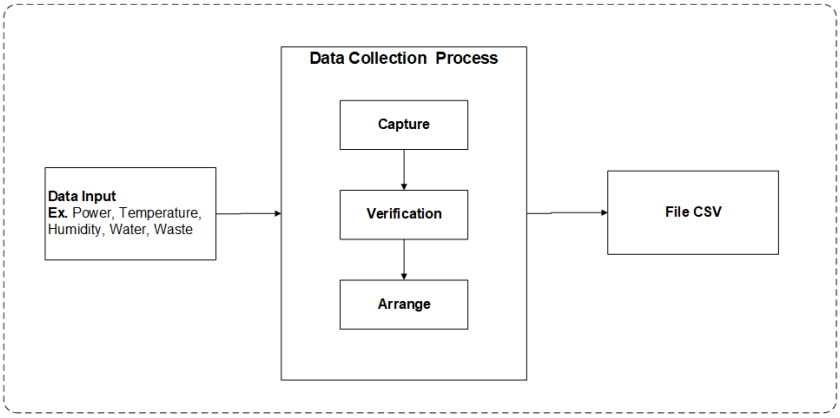


Fig. 2 Data collection process flow chart.

Data capturing process: Each category of data was captured as one CSV file per day from 00.00 AM to 23.59 PM. Power consumption data were collected from the smart meter in each building at the interval of 1 data per second. The temperature and humidity data were collected from the temperature and humidity sensors installed inside the building at 2 second interval.

The waste generation data was collected at the community trash bins, where load cells were installed under each trash bin. There are 4 trash bins; one each for general, recyclable, hazardous and organic wastes. The weight data for the waste were collected every 5 seconds.

Data verifying process: For the data verification, all CSV files were examined for data completeness. In cases of sensor malfunction, the data were not collected and was shown as zero. The data verification process deleted the corrupted data leaving only the usable data to be stored in the database.

Data arranging process: The next step was arranging the data files into chronological order. The structure of each data file had to be arranged with the specified column according to the database. Each data category had different set of specified column ID.

2.3 Data analysis and database design

The data flow in the Smart Community were analyzed to determine the relationship between data input and data output. The data flow is shown in Fig. 3 as Entity-Relationship Diagram (ER Diagram) which described the structure and relationship of data in the database. Subsequently, the collected data will be easier to manage through DBMS database management system. The database was developed through MySQL open source software with SQL language.

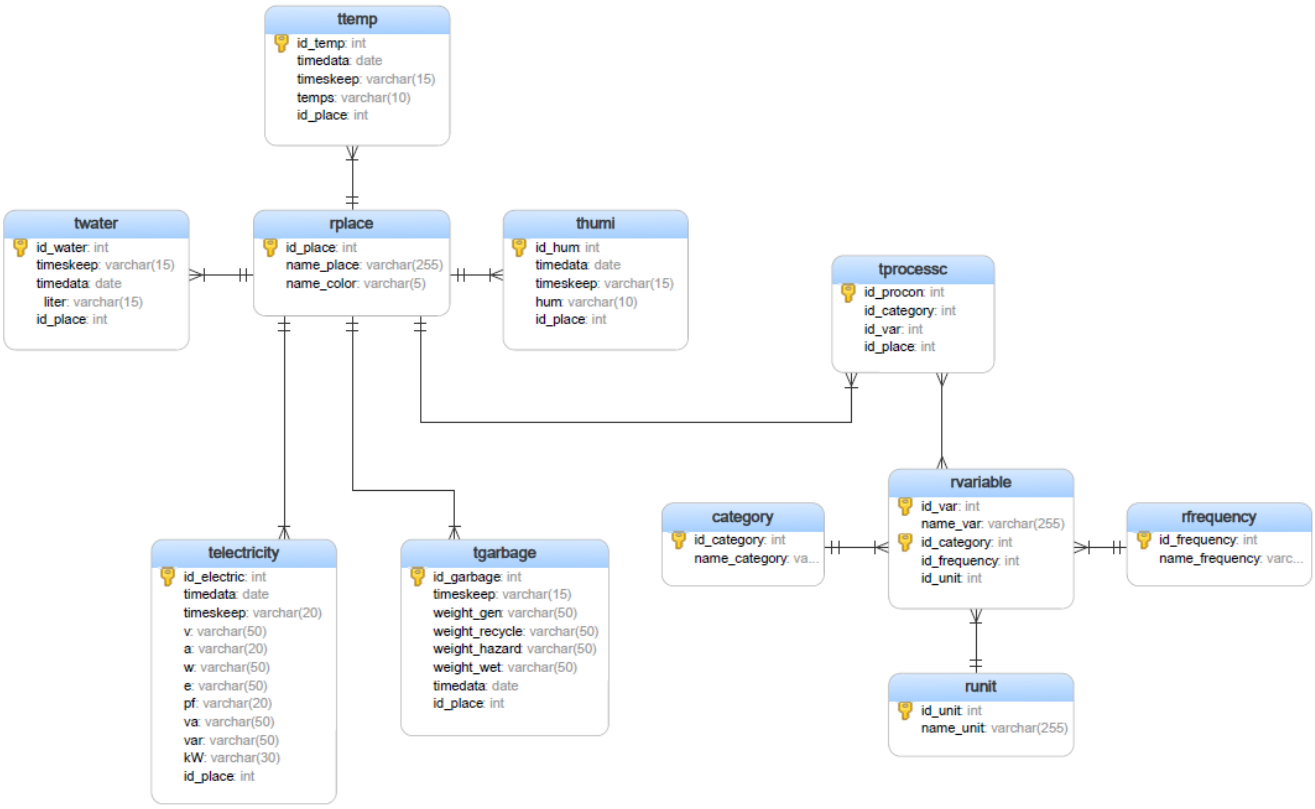


Fig. 3 ER Diagram for Data Analysis of Smart Community.

The database relationship structure design has 11 tables which are 6 Transaction Database tables and 5 Reference Database tables. The Transaction Database is the database with constant changes (

Table **2**). It is used for SQL Insert Statement, SQL Update Statement and SQL Delete Statement. "telectricity", "ttemp", "twater", "thumi", and "tgarbage" are the Transaction Database Tables used for collecting energy consumption, temperature, water usage, humidity and waste data from each building, respectively. The Transaction Database Tables are linked to the "rplace" which is the Reference Database Table. "tprocess" is the table used for collecting category and variable data which has linkage between "category" Table, Rvariable Table and Rplace Table.

The Reference Database is the type database that does not change. It is the structure of the fixed database which is used as reference to the other tables in the same database (

Table **3**). "rcategory" is the Table used to collect data, to separate data into groups, and to find statistical analysis according to the data category from the community. It is also the Table used as reference to "tprocessc" Table. "rplace is the Table used to collect data of building number and name. It is also the referece to "tprocesc", "twater", "tgarbage", "telectricity", and "ttemp" Tables.

"rfrequency" Table is used for collecting frequency data in data collection and storage. It is also the reference to "rvariable" Table. "runit" Table is responsible for collecting unit data for energy value measurements and collection time period. It is also the reference to "rvariable" Table. "rvariable" Table is fused for collecting variable data for energy data category. It also collects frequency data, energy unit data and act as reference to "tprocessc" Table.

Table 2 Transaction database table.

Table Name	Name	Type	Length	Detail
telectricity	id_electric	Int	11	Used to sequence the data collected
	timedata	Date	0	Use to collect date for the electricity consumption data
	timeskeep	Varchar	20	Use to collect time period during electricity consumption data collection
	v	Varchar	50	Voltage (V)
	a	Varchar	20	Current (A)
	w	Varchar	50	Power (W)
	e	Varchar	50	Energy (kWh)
	pf	Varchar	20	Power Factor
	va	Varchar	50	Apparent power (VA)
	var	Varchar	50	Reactive power (VAr)
	kW	Varchar	30	Use to collect electricity consumption data from each building
	Id_place	int	11	Use as location reference from rplace Table
ttemp	id_temp	Int	11	Used to sequence the data collected
	timeskeep	Varchar	15	Use to collect time period during temperature data collection
	temps	Varchar	10	Use to collect temperature data from each building
	timedata	Date	-	Use to collect date for the temperature data
	Id_place	Int	11	Use as location reference from rplace Table
twater	id_water	Int	11	Used to sequence the data collected
	timeskeep	Varchar	15	Use to collect time period during water usage data collection
	liter	Varchar	15	Use to collect water usage data from each building
	timedata	Date	-	Use to collect date for the water usage data
	Id_place	Int	11	Use as location reference from rplace Table
thumi	id_humi	Int	11	Used to sequence the data collected
	timeskeep	Varchar	15	Use to collect time period during humidity data collection
	hum	Varchar	10	Use to collect humidity data from each building
	timedata	Date	-	Use to collect date for the humidity data
	Id_place	Int	11	Use as location reference from rplace Table
tgarbage	id_garbage	Int	11	Used to sequence the data collected
	timeskeep	Varchar	15	Use to collect time period during waste data collection
	Weight_gen	Varchar	15	Use to collect general waste data
	Weight_recycle	Varchar	15	Use to collect recyclable waste data
	Weight_hazard	Varchar	15	Use to collect hazardous waste data

	Weight_wet	Varchar	15	Use to collect organic waste data
	timedata	Date	-	Use to collect date for the waste data
	Id_place	Int	11	Use as location reference from Table rplace
tprocessc	id_procon	Int	11	Use to sequence data collection for process data
	Id_category	Int	11	Use to collect reference order from category Table
	Id_var	Int	11	Use to collect reference order from rvariable Table
	Id_place	Int	11	Use as location reference from rplace Table

Table 3 Table reference database.

Table Name	Name	Type	Length	Detail
category	id_Category	Int	11	Use to collect sequence order of the category
	Name_Category	Varchar	255	Use to collect category name
rplace	id_place	Int	11	Use to collect sequence order of building data
	Name_place	varchar	255	Use to collect building name
rfrequenc y	id_frequency	Int	11	Use to collect sequence order of frequency
	Name_frequency	varchar	255	Use to collect frequency name
runit	id_unit	Int	11	Use to collect sequence order of measurement unit
	Name_unit	varchar	255	Use to collect unit name
rvariable	id_var	int	11	Use to collect sequence order of variable
	Name_var	varchar	255	Use to collect variable name
	id_category	int	11	Use to collect sequence order of category data with reference from category table
	id_frequency	int	11	Use to collect sequence order of frequency with reference from rfrequency table
	id_unit	int	255	Use to collect sequence order of unit with reference from runit table

2.4 Smart community energy data management system design

Energy data management system design must consider the analysis of collection process of data flow including relationship between input data, process and data output.

2.4.1 Programming language

The selected computer language for this work are SQL (Structured Query Language) for system development. SQL was used to access and manage the system database. SQL is the database program with simple language structure, high efficiency, and able to work complexity. In this work, SQL was used in 4 commands.

1. Select query for select data
2. Update query for edit data
3. Insert query for add data
4. Delete query for delete data

The energy data management system was developed with large and complex database. The data is constantly changing every second and the data is stored everyday which create complexity. SQL

language has more functions to assist in problem solving of the system. The fuctions that were selected to manage the database for convenient of usage were SQL SUBSTRING(), SQL COUNT(), AVG() and SUM().

Function SQL SUBSTRING() was used to gather and section the time range of the data collection as 1 hour range as shown in Fig. 4. The fixed time range made generating summary report for 1 data easier and more simple.

```
SUBSTRING_INDEX (string, delimiter, number)
```

Fig. 4 Function SQL substring

Function SQL COUNT(), AVG() and SUM() were used to count, average and sum data, respectively, according to the specified condition. The data were gathered and averaged for all 4 energy data categories as 1 hr interval. The syntax that were used for this command is shown in Fig. 5.

```
SELECT COUNT(column_name)
FROM table_name
WHERE condition;
```

(a)

```
SELECT AVG(column_name)
FROM table_name
WHERE condition;
```

(b)

```
SELECT SUM(column_name)
FROM table_name
WHERE condition;
```

(c)

Fig. 5 Function SQL (a) COUNT, (b) AVG, and (c) SUM.

3. Results and discussion

3.1 Data capturing

The five energy data categories, which included data that are directly related and indirectly related to energy consumption were collected through sensors. The sensors for electricity consumption, indoor temperature, indoor humidity, water usage and waste generation are shown on Fig. 6 Not all of the 12 buildings were installed with sensors. Electricity consumption sensors were installed in four buildings that were occupied dwellings. Temperature and humidity sensors were installed in only ten buildings, however one sensor setup malfunctioned. Thus, only nine buildings generated data on indoor temperature and humidity. The water sensors were installed in five buildings, however, there were difficulty with real-time data recording, that water usage were recorded only in two buildings. The Smart Community had one central location for waste disposal, from where waste generation data were recorded.

The data collection process is described in Section 2.2 above. The data is collected as 1 CSV file per day per energy data category. The picture of CSV file is shown in Fig. 7. It has different file name format depending on the data category and type of sensor. The variety of data formats posed made data verification and arrangement complex.

For electrical data, an example CSV file is stored as LOGGER01.CSV for 1 file per day. Each file composed of data for DATE, TIME, Voltage (V), Current (A), Power (W), Energy (kWh), Power

Factor, Apparent power (VA), Reactive Power (VAr), and Total kWh (kWh). The data is recorded every 1 seconds for 86,400 data records per day (Table 4). The file name for energy data showed only LOGGER and a number. It does not contain information for date or type of energy data category.

For the temperature and humidity data, the sensors are bundled and installed together. The examples of CSV files for temperature and humidity are shown as 7-18-4-2018-temp.csv and 7-18-5-2018-hum.csv, respectively. The file name for water usage sensors is the same format as the temperature and humidity data file names such as 5-18-4-2018-water.csv. The file name format indicates the date and data category. However, the file name formats for temperature, humidity and water were different from the electrical data file name.

For the waste generation data, the sensors are installed in four waste bins placed at the central waste disposal location of the Smart Community. There are four waste bins, each for general, recyclable, hazardous and organic wastes. The data from the 4 sensors are grouped together into one file per day. The example of the CSV file name was all4-1-7-2018-kg.csv which indicate the date and type of data.

From Table 4, the results from the data collection revealed that the real data that were collected was less than the expected number of record per day for each data category. This was due to an external factor that could not be controlled. The microgrid power condition was unstable, as there were internittent blackouts and voltage fluctrations. These conditions caused sensors to fail. In addition, the Wifi was not stable as well due to the weather thus, the data could not be recored continously. Therefore, it was very important to verify and screen the data.

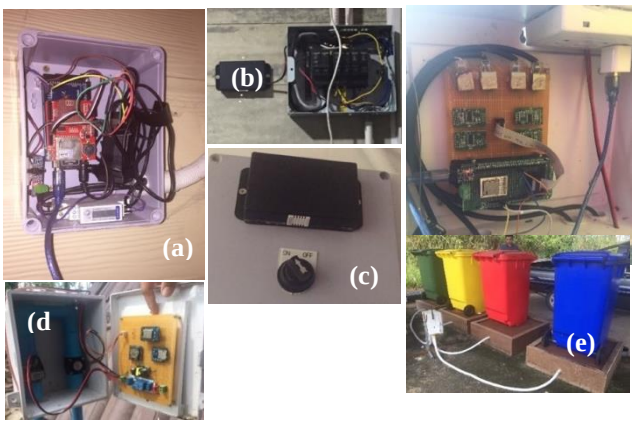


Fig. 6 Example of sensors installation to collect data for (a) electricity, (b) temperature, (d) humidity, (c) water usage and (e) waste.

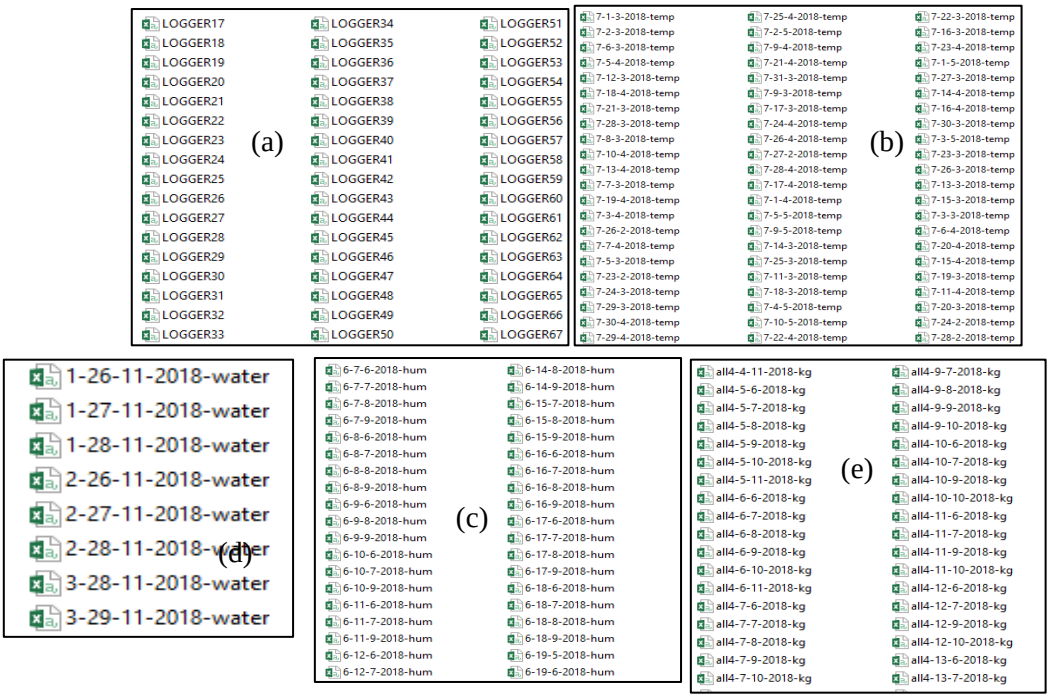


Fig. 7 CSV files from installed senstors for (a) electricity, (b) temperature, (c) humidity (d) water usage and (e) waste.

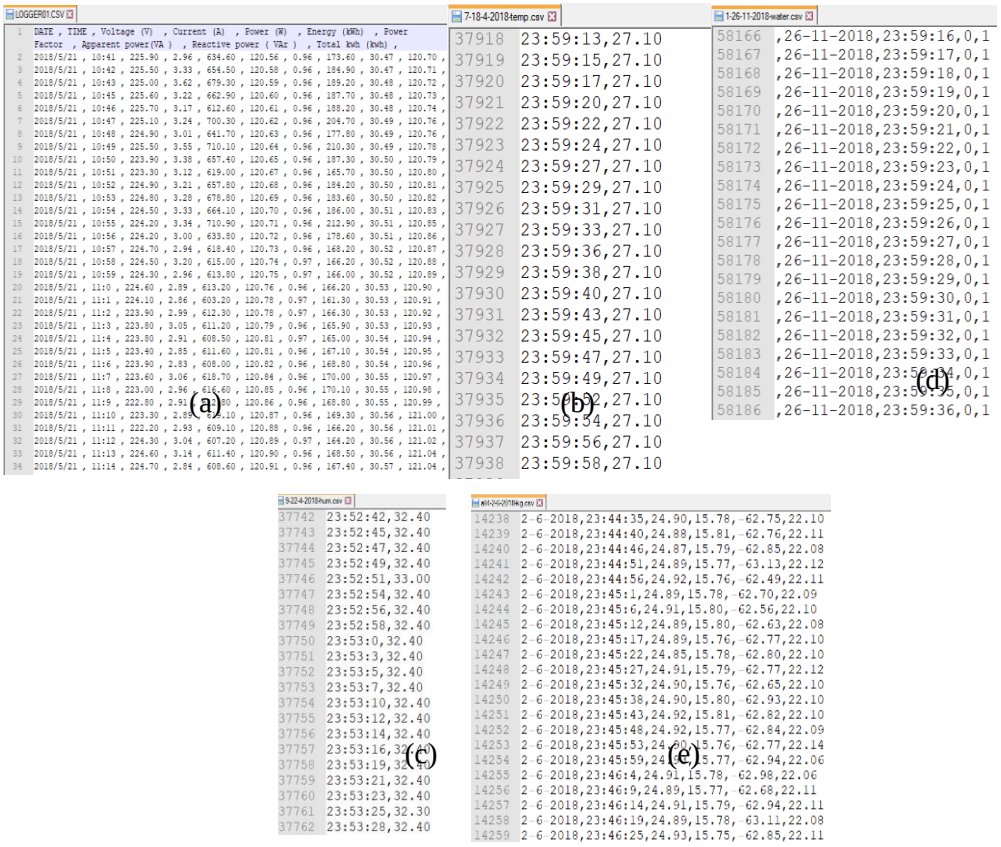


Fig. 8 Raw data from CSV file for (a) electricity, (b) temperature, (c) humidity, (d) water usage and (e) waste.

3.2 Data verification

Table 4 indicated the types of data unit for each data category. The electricity data file contained information for Date, Time, Voltage, Current, Power, Energy, Power Factor, Apparent power, Reactive power, Total kwh, and kW.

For temperature, humidity and water data files, the information format for Date, and Time were similar. The specific information format used for data collected for temperature, humidity and water volume, were Celsius (°C), percent humidity (%), and Liter (L), respectively. Wastes data file had information on Date, Time, Weight of waste (in kilograms) by type, i.e.; General waste, Hazardous waste, Recycle waste, and Organic waste (kg).

There was also discrepancy from the information data file of the same data category. For example, in humidity files, some files had columns for date, time, and % humidity, however, some files only had time and % humidity. A large number of data collected. For example, the temperature raw file had approximately 34,753 records per file which was time consuming to verify. The main issues during the verification process were; first, the large amount of data in the raw csv file were not of the same format; and second, the data were not clearly separated or grouped into columns, thus not ready for examination.

The verification process involved on generating good and usable data files to be imported to the database. In this step, all CSV raw files were examined via Notepad. Files that did not have data caused from sensor error or electrical blackout/instability were deleted.

Table 4 Number of Collected Data Records and Units (Calculated and Real Data Collection)

Energy Data Category	Time interval of data record collection	Calculated Maximum Number of record per day	Type of collected data unit for each data category	Number of data unit collected per record	Calculated Maximum number of data unit collected per day	Number of buildings that sensors were installed	Calculated Maximum Number of data unit per day for each category	Calculated Data: Maximum number of record per day for each data category	Real Data: Number of total record collected per day imported to database	
									Minimum	Maximum
Power Consumption	1 second	86,400	Date, Time, Voltage, Current, Power, Energy, Power Factor, Apparent power, Reactive power, Total kwh, kW	10	864,000	4	3,456,000	345,600	244	16,056
Temperature	2 seconds	43,200	Date, Time, degree Celsius (°C)	3	129,600	9	1,166,400	388,800	30	265,627
Humidity	2 seconds	43,200	Date, Time, percent (%)	3	129,600	9	1,166,400	388,800	1,353	37,951
Water	1 second when in use	86,400	Date, Time, Liter (L)	3	259,200	2	518,400	172,800	6,344	74,764
Garbage	5 minutes	17,280	Date,Time, General waste, Hazardous waste, Recycle waste, Wet waste (kg)	6	103,680	1	103,680	17,280	1,986	16,731

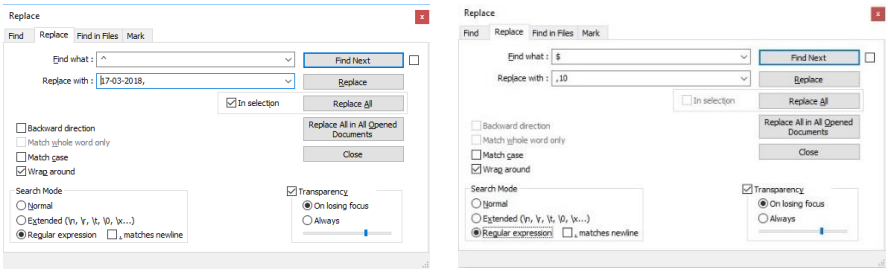
3.3 Data arrangment

The data arrangement was the most difficult part because the files had different formats and different number of records. The files must be in the structure compatible to be imported to the designed database.

Each file had been arranged in chronological order and the structure set to similar row and column via Notepad++. The files were processed with Replace with Regular Expression with "^" command by inserting date in the first column of each file (Fig. 9 (a)) and then inserting the building code at the end of the record with "$" command (Fig. 9 (b)).

After file structure arrangement, all the files per one month must be combined according to the data category and then uploaded to the database. The Batch file command was used to combine the files.

Fig. 10 shows an example of file combining process into a new file called combinehum.csv. After all the files collected in one month were captured, verified, arranged, and combined, this combined monthly file was uploaded to the database of the energy data management system.



(a) (b)

Fig. 9 Replace with Regular Expression (a) Insert first text of line and (b) Insert text at the end.

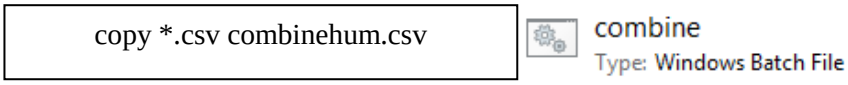


Fig. 10 Command and symbol Batch File.

3.4 Database file input

The time for uploading the data files depended on the number of records in the data file. On average, it took approximately 1 second to upload 2,088 records as shown in Fig. 11.

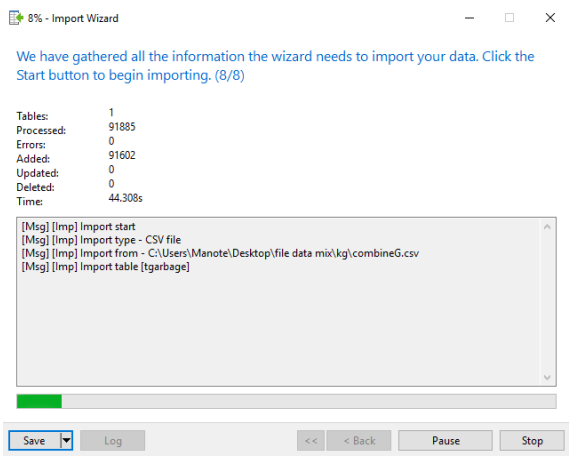


Fig. 11 Data transfer window.

After the data were uploaded, the data could be retrieved for use by SQL language. Function SQL SUBSTRING() and AVG() were used to gather and split the data interval time period to be 1 hour (Fig. 11). Query time was applied in 12.148s to achieve Crosstabs by apply command such as IFELSE, SQL COUNT(), AVG() and SUM(). The data were displayed horizontally which was easier to understand and report (Fig. 12). In this work, Query Data method was kept in View for ease in usage, reduce the operation of Query and faster data retrieval.

The technique developed for data capture, verification, and arrangement provided the data with the same format to be transferred to the database. The data management process was based on theresearch of Yang, et. al. [7] on the effective management of large data to reduce the uncertainty by verification of raw data and reformatting the data structure.

Data cleansing is necessary to generate usable and correct data [8]. After data cleansing, the data in the database can now be extracted for analysis or reporting. The functions from SQL were used to retrieved data for display. The query was stored in view to reduce the operation of query data. This is similar to the work of mithani, et. al. [9] on modifying the Query for faster and more efficient retrieval of large and complicated data.

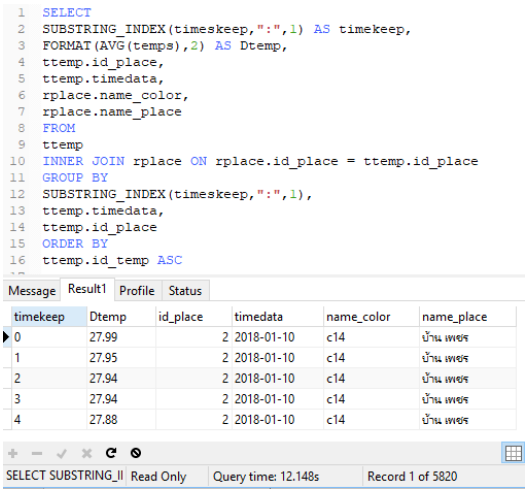


Fig. 11 Function SQL SUBSTRING ().

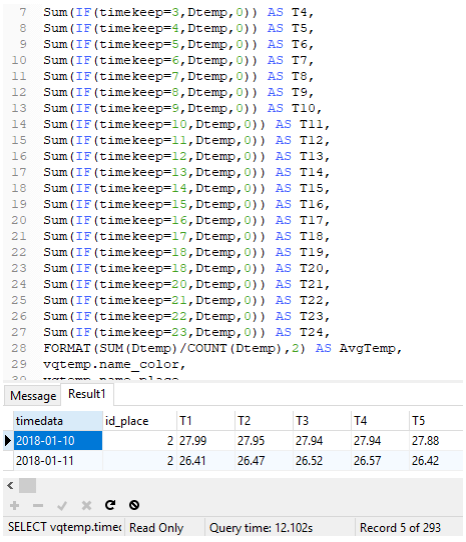


Fig. 12 Crosstabs for the horizontal data.

4. Conclusion

This paper focused on the procedures for energy data collection and processing. Data collected included both direct and indirect energy data. Direct energy data cover energy generation and consumption data, while indirect energy data include those data affecting energy generation and consumption such as water usage, indoor temperature, humidity and waste generation. Data were collected via sensors installed in the buildings located in the model Smart Community. The data were then processed through three steps: Capture, Verification and Arrangement. The structure of the datasets were formatted to be specifically compatible with the database. Approximately 1,800 data files per month were processed. Each data file has the range of 17,280 – 86,400 data records depending on the data category and collection interval. The data collection from sensors installed in buildings under real-world conditions has several challenges. The sensors, and thus data transmission, were interrupted by recurring blackouts. Some sensors were also damaged by animals such as ants and geckos.rs. Finally, this research demonstrate that data verification is a very important part of data processing, as in this process, the data are screen, to collect the usable data and reject the bad ones.

Acknowledgements

We would like to thank asian development college for community economy and technology, Chiang Mai rajabhat university for the energy data and facilities. The funding for this research was provided by the energy conservation promotion fund, Thailand.

5. References

[1] Chassin, D.P., Fuller, J.C., & Djilali, N. (2014). GridLAB-D: An agent-based simulation framework for smart grids. *Applied Mathematics*, 12-24.

[2] Jin, S., & Chassin, D.P. (2014). *Thread Group Multithreading: Accelerating the computation of an Agentbased Power System Modeling and Simulation Tool – GridLAB-D*, in *International Conference on System Science*. Hawaii.

[3] Barnes, M. (2007). *Real-World MicroGrids-An Overview*. in *System of Systems Engineering, 2007. SoSE '07. IEEE International Conference on*.

[4] Hatziargyriou, N. (2007). *Microgrids.* Power and Energy Magazine, IEEE, *5*(4), 78-94.

[5] Mohanty, S. (2016). *Everything You Wanted to Know About Smart Cities*, *5*, 60-70.

[6] Taylor-Sakyi, K.(2016). *Big Data: Understanding Big Data.*

[7] Y, X.U. (2015). *Knowledge Management in Big Data Times*. in *2015 IEEE Fifth International Conference on Big Data and Cloud Computing.*

[8] Data cleansing, National Science and Technology Development Agency. (2012). Retrieved from https://www.nstda.or.th/th/nstda-knowledge/2910-data-cleaning

[9] Mithani, F., Machchhar, S., & Jasdanwala, F. (2016). *A novel approach for SQL query optimization.* Paper presented at the 2016 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC).