

Data Analytics of Building Automation Systems: A Case Study

Gulustan Dogan¹, Ted BROWN², Eric Dagobert³, Marco Ascazubi⁴, Joseph Manorge⁵

Accepted : 13/04/2018 Published: 29/06/2018

Abstract: In today's technology, when costs of time, energy and human resources are considered, efficient use of resources provides significant advantages over many aspects. In light of this, role of building automation systems, which are a part of smart cities, become even more important. At the very core of building automation systems there lies the efficient use of resources and systems for providing comfortable living situations. With the advancement in network technology, systems can be programmed smartly and any malfunctions on the systems can be detected and fixed remotely. In addition to that, all data gathered during this process can be analyzed to create machine-learning solutions for a system to control and program itself. In this document we are presenting a Web application offering features of data analysis and most importantly predictive modeling in the context of building data energy management. As of today, the implementation is made from a CUNY building at John Jay College and contains thousands of data collected from hundreds of sensors over a period of two years, and regularly updated. That is a particular context but the tool can easily be adapted to any type of data environment based on time series. The system articulates around three concepts: visualization, and predicting statistics and forecasting. Visualization is made possible with powerful widgets, and statistics and forecasting based on Python modules. The web client server architecture has several purposes, including, of course, the ones related to any web application, but what is most important it allows transparency between users; every user being able to see each other works. Overall, the originality of this application comes from its high degree of customization: indeed it contains an on-the-fly python interpreter ready to be used with the data, itself encapsulated inside a python object. Therefore, all kind of formulation is allowed to be immediately displayed. The forecasting part is versatile as well, and it sits on python machine learning features, but adapted to manipulate time series.

Keywords: Big Data, Time Series, Machine Learning, Research tool, Building Automation Systems, Smart Building

1. Introduction

Smart buildings are systems where the building energy system is controlled from a single point. In the building automation systems, the front panel is keeping the user comfort at the highest level. Building automation systems are predominantly used in places such as airports, hospitals, universities, schools, classrooms where energy consumption is high [1]. The types of sensors used in these types of buildings vary. However, the sensor sector is rapidly evolving in line with existing needs, and this field allows us to work with the systems more efficiently each day while producing new solutions. The development of technology is a factor in reducing the hardware costs of sensor and building automation systems in proportion to the competition in this area. In this respect, the use of building automation systems not only in public institutions but also in other areas has increased significantly. [2]. In this study, the types of sensors used in the building automation system of New York City University (CUNY) and the data obtained from these sensors will be analyzed.

There are many systems these days that are generating a large amount of data, some of which to be analyzed in real time. Data mining is now present in every domain: marketing, finance and medical applications to name a few. Building management systems

are an example of such applications. They are capable of generating in short time increments millions of values, too much for any operator to examine even retrospectively. In the mean time this mass of data is often correlated, or has cycles, and sometimes has patterns or trends, upon which analysis should be based. In this paper we will present a tool having features to analyze aspects of the energy use in a building equipped with a modern BAS system. The tool is plugged on a database filled with thousands of data collected from hundreds of sensors located in the New York's John Jay Building and regularly updated over a period of two years and a half. That tool is based on open source software and can easily be adapted for any other uses involving time series. Architecture and design have been motivated by different constraints and certainly modified several times before reaching the actual shape. The final architecture sits on the following principles: speed, easiness to use and to share, and flexibility. We first opted for R Studio that offers points 2 and 3 but we found R a little weak in terms of performance. We opted for a client-server architecture mainly based on Python libraries.

Section 2 presents related work. Section 3 defines sensors and building automation systems. Section 4 defines the interfaces of the developed website. Section 5 presents the possibilities of immediate achievement in terms of graphical analysis. Section 6 defines how the system computes graphics. Then, in Section 7 we will describe the statistical methods offered, and finally in Section 8 we will detail machine learning features and their application in

¹ Yildiz Technical University – 34220, İstanbul, TURKEY

² City University of New York, New York USA

* Corresponding Author: Email: gulustan@yildiz.edu.tr

the domain of energy savings. A brief conclusion closes the paper.

2. Related Works

Some work has been done in the field of building automation. Previous research has focused on building structure and passive thermal storage in furnishings using computerized "white box" building models based on thermodynamic rules [17, 18, 19, 20]. Confirming that such models are more successful than sampled empirical data led various research groups to model "gray boxes" by relying on model variables and machine learning [20, 21, 22]. Studies related to building automation have been made especially in relation to energy saving [23]. The main objectives of sustainable design are; can be summarized as optimization of renewable and non-polluting environmentally friendly energy and ecotechnology, use of materials, conservation, saving and recovery in all sources including water and energy, potential of the plot and its surroundings, operation, maintenance and repair of the building. A large number of building examples that generate electric energy by using PV panels as shading device, façade and roof covering, which optimally utilize renewable energy sources such as sun and wind as the intelligent passive system in the world and minimize heating, air conditioning, ventilation and lighting energy loads [4]. By making smart network technologies become viable nowadays, many countries around the world have set smart network vision and implemented programs. The European Union has established the Intelligent Networks Technology Platform and has set the roadmap for the intelligent network.

EU countries have started their infrastructure work to establish 20-20-20 targets for 2020, 20% of energy generation from renewable sources, 20% reduction of CO₂ emissions and 20% increase in energy efficiency. In Europe, about 10% of houses are smart meters, and this ratio is targeted to be increased to 80% by 2020 when 200 million smart meters are active throughout Europe. The US government finances several intelligent network projects in the country at a certain rate. It is expected that the cost of intelligent network investments in the United States in the next 20 years will be between \$ 338 and \$ 476 billion. US Smart grid vision is to produce 80% of electricity from renewable sources in 2035 and to use 1 million electric vehicles by 2015. Japan has taken intelligent network projects one step further and launched smart city pilot applications. Apart from Europe, USA and Japan, China, South Korea, Canada and Australia have also started to be interested in smart networks [10].

Jong Jin Kim from the Department of Architecture and Urban Planning at the University of Michigan mentions the three essentials for achieving environmental sustainability. These are: economic use of resources, life cycle design and civilized design. Economic use of resources; reuse and recycling of natural resources entering the building. Life cycle design; provides a methodology for the environmental impact of building operations. In the case of civilized design; focuses on the interactions between people and natural life [11].

The first environmental assessment tool used internationally; the Building Research Establishment (BRE) in the United Kingdom established The Building Research Environmental Assessment Method (BREEAM) in 1990. Over the years following the emergence of BREEAM, many different environmental assessment methods emerged as a result of similar studies. Building Environmental Performance Assessment Criteria (BEPAC) established by the Government of Canada in 1993, the

Hong Kong Building Environmental Assessment Method (HK-BEAM) in Hong Kong in 1996, Leading in Energy and Environmental Design (LEED), created by the American Green Building Council in 1998, is a prime example of these [11].

- Istanbul World Trade Center

The hot fluid obtained from the winner is given to the air handling units and the air heated and humidity adjusted. In the stores, the hot air is sent to the thermostat-controlled heating coils at the VAV unit outlets, and the required ambient temperature is obtained at the stores.

Wastewater is collected in a treatment facility near Ayamama Creek and subjected to biological treatment. The treated water is used for garden watering according to the need. Too much water is discharged into the Ayamama River. Rain and water in the building and parking areas are discharged to Ayamama Derby via rain water pumps at the station [8].

- Yapı Kredi Bank Operations Center Building

Each floor of the building is ventilated using an airflow system underneath the upholstery. The air is provided at a low level from the raised upholstery and is directed to the ceiling with a natural upward. In order to reduce the temperature changes, reinforced concrete elements on the inner surface are left uncovered [8].

- Foreign Trade Complex

Lighting design in the Foreign Trade Complex is based on daylight. In the floors where the office spaces are located, the light is taken from the atrium to the interior spaces. The glass sections and office areas are also illuminated. Besides the daylight, artificial lighting elements are also used in large numbers. Daylight and dimmer controlled energy efficient artificial lighting system is applied. The lighting system is controlled by the building automation system [8].

- Commerzbank Central Building

Germany, which is Europe's tallest office building in Frankfurt building. Commerzbank, one of the few intelligent buildings in the world, is an ecological building that uses building and office automation systems to achieve maximum performance with minimum energy consumption. The building's plan consists of work areas arranged around a triangular atrium. The atrium was separated by a horizontal glass partition on every 12 floors, and the flow of air was directed so that the chimney effect was removed [8].

- RWE Tower

The world's first ecological tower RW cylindrical form of the tower facilitates the vertical circulation of the air flow on all floors and diagonal ventilation. The air from the building is sent centrally to the vertical pipes in winter to provide heat recovery. The facade provides good insulation and combined sun protection elements in winter and effective solar protection in summer [8].

- Bahrain World Trade Center

The structure consisting of twin towers is connected with 3 bridges at a height of 240 m. Each bridge has one wind turbine attached to it. These skyscrapers increase the efficiency of the project by directing the wind between each other and increasing its speed. The unique shapes of these skyscrapers, on the other hand, minimize the differences in pressure between the bridges, minimize the differences that can occur due to the increase in wind speed as the height increases, and provide an even wind speed distribution between the turbines. All of these features also make it possible to get extra efficiency in powering generators [8]

- Lion House Office Building, Alnwick Northumberland, UK

It is a very successful and highly awarded office building as a sustainable building example completed in England in 2009. DEFRA (Department for Environment, Food and Rural Affairs) with a use area of 1600 m², consisting of two blocks, zero emission (zero emission) is a building. BREEAM-Outstanding Awards for Offices 2008, New Building Project of the Year 2009-CIBSE Low Carbon Awards, Sustainable Achievement of the Year 2009, Property Week, Green Apple Award 2009, Sustainability - RICS North East Renaissance Awards, CIBSE Environmental Initiative of the Year - Constructing Excellence in the North East Awards, Project of the Year - British Construction Industry Awards and Environmental Initiative - Building Services Awards [9].

Increasing use of energy with industrialization has resulted in the reduction of available energy resources and increasing environmental pollution, energy efficiency. Effective use of energy has become one of the core strategies of many countries to support the use of renewable energy sources and to support policies that prolong the life of existing resources. Due to the energy they spend on heating and cooling systems, buildings are also known to have an important share in the energy struggle. The primary consideration for making buildings less energy efficient is to consider the impact of design decisions on energy use [12].

Design parameters that are effective in energy conservation can be defined as location selection, building spacing, orientation, volume organization and building envelope. By controlling these parameters, it is aimed to design the buildings that are least needed for the construction systems and therefore reduce the least use of energy resources [13].

The energy performance of the buildings as a passive system and the energy efficiency of the mechanical and electrical-electronic systems in the building are directly related to the architectural design parameters of the building. Among these parameters, the most important ones are the location of the building, the position, direction, form and the building shell according to the other buildings. Each of these parameters should be determined in relation to each other in such a way that each of them plays an important role in energy efficient building design and therefore intelligent building design and that the effects on the energy performance of the building are related to each other and that each of them makes optimum use of renewable energy resources. Since the most important goal of intelligent buildings is to ensure that buildings are energy efficient, the importance of these architectural design parameters in the design of intelligent buildings cannot be denied. Otherwise, the building can only be controlled by automation, mechanical and electrical-electronic systems, and cannot go beyond being a classic building [14].

All kinds of electrical and electro-mechanical equipment in today's buildings; comfort, economy, quality and safety. There are many different systems in the buildings, ranging from heating, ventilation and air conditioning (HVAC) systems to fire and security systems, lighting, emergency power distribution, elevators and process control systems. Central monitoring and control of these systems, management is an important requirement for operation and maintenance. A well-designed control scenario is required for the HVAC systems in the buildings to function as an energy efficient system for comfort in all conditions. The control strategies and parameters must be selected appropriately in order to obtain the highest expected yield from the HVAC system and to keep the system accuracy at the highest level [15]

One of the most important elements of providing good control is the presence of sensors that work perfectly in the system to be controlled. Recently, in the direction of "open protocol" concept developed in building automation systems, it has begun to be produced so that sensors can communicate directly with these protocols. Microprocessors added to the sensor without any change in sensor structure and operating principle can communicate directly with automation software and other nodes operating in the network. However, no matter which system is used, the sensors remain the indispensable component of automatic control and automation systems [16].

3. Sensors And Building Automation Systems

3.1. What Are Sensor And Transducer?

People perceive some physical quantities such as heat, light, pressure, and sound with their sense organs. These physical quantities are transducers and sensors, just like ours, and the elements that activate or deactivate some of the equipment as a result of the detection.

3.2. Sensors In Daily Life

Sensors are used in many areas as part of our everyday life. Numerous sensors for different and different purposes can be used for health, education, industry, etc. areas; It is used in vehicles, airports, automatic lamps, automatic doors, safety systems, gas and liquid measurement, alarm systems. The gas sensor shown in Fig. 1 is a sensor type that can prevent possible explosions and fires by alarming if the flammable and explosive gases such as natural gas, methane, propane, hydrogen, acetone exceed the limit values determined for the environment.

3.3. General Design Of Building Automation System

Class A (Dark Green): High Performance Building Automation Systems and Building Technical Management

Class B (Green): Developed Building Automation Systems and Building Technical Management

Class C (Light Green): Building Automation Systems

Class D (Yellow): Energy-efficient building automation systems [3].

3.4. Building Automation Systems

Mechanical System Control

- Air conditioning devices
- Pumps, Hydrofords, Fans and so on.
- Lighting System Control

Enlightenment of the light according to the development during the day

- Accurate and accurate illumination
- Maximum savings in lighting solutions

Electrical System Control

- CCTV systems
- Fire Protection
- Control of any kind of electrical control device (energy analyzer, compensation systems, separator, circuit breaker, relay, contactor etc.) can be done from a certain point.

Use either SI (MKS) or CGS as primary units. (SI units are strongly encouraged.) English units may be used as secondary units (in parentheses). This applies to papers in data storage. For example, write "15 Gb/cm² (100 Gb/in²)."

units are used as identifiers in trade, such as “3½-in disk drive.” Avoid combining SI and CGS units, such as current in amperes and magnetic field in oersteds. This often leads to confusion because equations do not balance dimensionally. If you must use mixed units, clearly state the units for each quantity in an equation. The SI unit for magnetic field strength H is A/m. However, if you wish to use units of T, either refer to magnetic flux density B or magnetic field strength symbolized as $\mu_0 H$. Use the center dot to separate compound units, e.g., “A·m².”

4. Properties Of The Building rscript.cisdd.org

In the building automation system of the John Jay College of Criminal Justice building of the University of Cuny (City University of New York), the building automation system was built in areas such as air conditioning, ventilation and energy efficient use of the building with sensors. In the system, a sensor is taken every 15 minutes from the sensors and stored in the database. The system software is set according to the usage times of the building and shut down the system during off hours. rscript.cisdd.org is a web application where the data from the sensors in the automation system of John Jay College of Criminal Justice building of Cuny University is stored in the database and all these data are opened. The data pertaining to the sensors located on the site are transferred to the database every 15 minutes. The obtained data can be queried from the site prepared using Python programming language, Django Framework and Sqlite database. All the sensors on the site, their locations, the sensors belonging to the sensors and the graphics of these sensors can be displayed. There is also a possibility to download and examine the data as a CSV file. rscript.cisdd.org has been developed for this study by a student of Professor Ted Brown who is the author of this article. Figure 1 is the main screen of the rscript.cisdd.org site.



Figure 1: The Website's Interface



Figure 2: Opening Window

4.1. Tools of website rscript.cisdd.org

The display in Figure 2 shows the types of sensors used in the building. Figure 3 also lists the respective location of the document sensors.

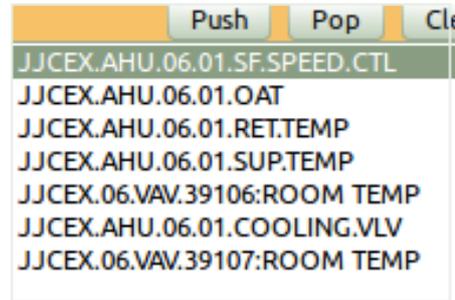


Figure 3: Sensor selection for the case study

Pop: Push is added to delete a desired sequence is provided. Push: Floor-PH selected with pop-up windows added. Clear: All the added data can be deleted. Permute: The data in the sensor list is presented in sequential order.

After making the selection, two buttons appear on the screen as shown in Figure 4. Use Filter and NoFilter. Their aim is to show the data without making any transfers.



Figure 4: Graph Raw Selection Window

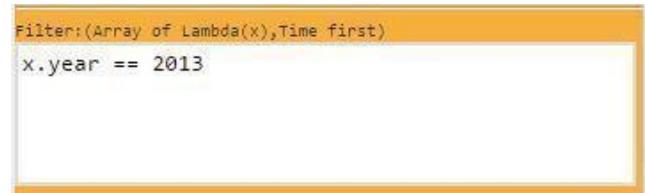


Figure 5: Graphic Filtering Equation

In Figure 5, after creating the graph, the relevant equation is displayed on the screen.



Figure 6: Graphic Types

In Figure 6, graphic type(x) selection section includes time series, table, moving std, xy, correl, frequencies options.



Figure 7: Configuration Selecting Area

In Figure 7, if you want to add something special while creating the graph, you can specify it.

Graph: Graph with button.
 Load: Loading.
 CSV: Dump of generated data.
 Save: The data is being saved.
 Time Series: The chart is based on time series.
 Table: Charting in tabular form.
 Moving Std: Animated graphics are being created.
 XY: The graphic is being created according to xy coordinates.
 Correl: Created in graphical lines.

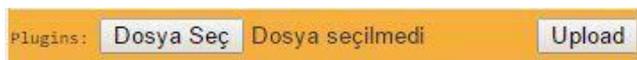


Figure 8: Plugin Selecting Window

In Figure 8, we see part of the installation process is done by selecting the plugin file. Figure 9 contains the section where we can run the codes we wrote with the machine language.

Plugins: Data is entered from the file in the machine language format available on the screen in Figures 8 and 9. When you create a graph, you can select log automation extensions that vary according to the selected graph type. The log screen is shown in Figure 10.

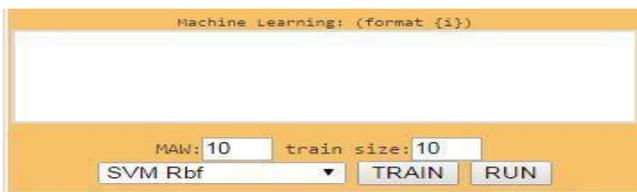


Figure 9: Machine Learning Algorithms Window

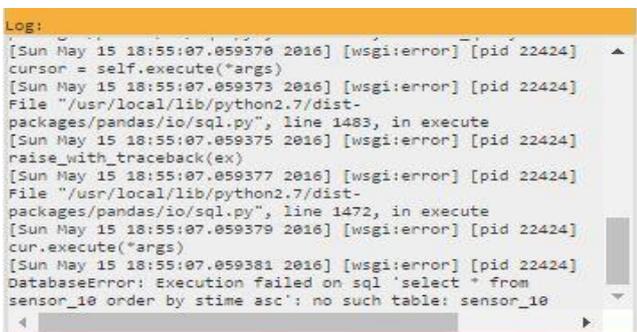


Figure 10: Log Window

Figure 11 contains a screen view of data from multiple sensors in the same graphic. Figure 12 contains a tabular view of the air volume data.

Figure 12: Sensor data in table view

Timestamp	JJCEX.06.CAV.39201:ATR VOLUME
01/01/2014 00:00:00	228
01/01/2014 00:15:00	252
01/01/2014 00:30:00	172
01/01/2014 00:45:00	0
01/01/2014 01:00:00	4

4.2. Sensor Types Used In Building

The air, water, steam or various streams in the systems are often regulated by automatic control instruments. The control devices that regulate water and steam flows in the systems are valves, while the dampers that regulate and control the air flow.

Five sensor systems were used in the analyzed building. These; Air Control System Sensors (ACS-Air Control System) Even when the equipment changes from system to system, the operations that are required to be performed by the automatic control system can be placed on specific main headings. The intent here is to control the whole system by dividing it into sub-particles that can control it, and then complete the system in terms of control by providing links to other parts of these particles.

Air Handling Unit Sensors (AHU-Air Handler Unit) An air handler, or air handling unit (often abbreviated as AHU), is a device that organizes and circulates air such as a heating element, ventilation, and air conditioning (HVAC) system. An air handler usually includes a large metal box bellows, heating elements, filter shelves or chambers, sound attenuators and cooling bumpers. Air handlers usually distribute air-conditioned air through the building and the AHU

is connected to the ventilation system by a duct system that turns it.

Air Control Valve (CAV-Control Air Valve)

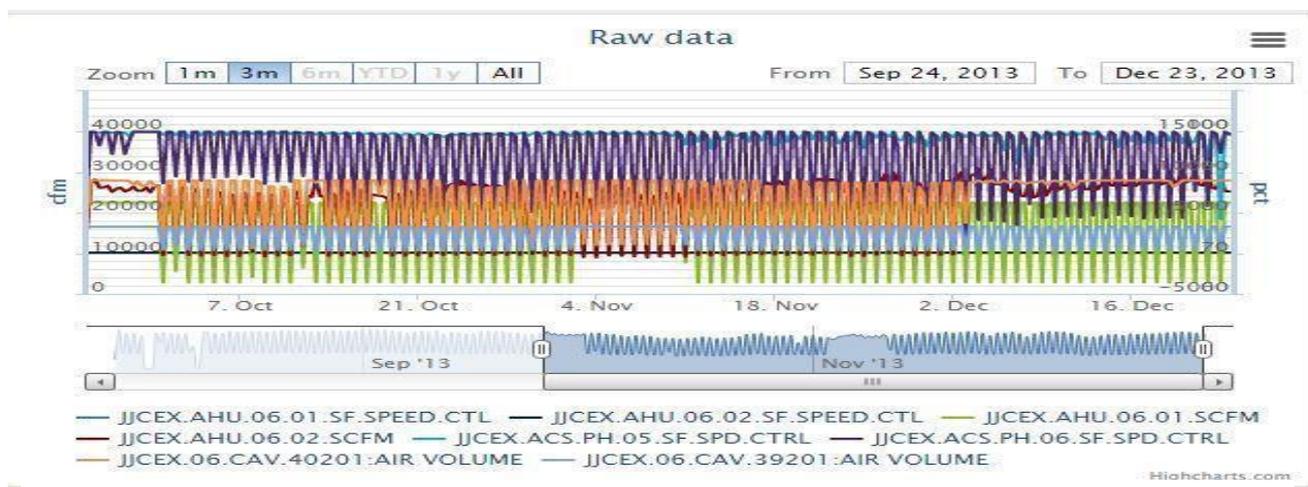


Figure 11: Graphic Comparing Multiple Sensors

Single-way valves, two-way valve or balance valve, three-way mixing valves, three-way separation valves.
 Secondary Pump Sensors (Secondary Pumps)
 Cooling Tower Sensors (Cooling Tow)

5. Data Visualization

In Figure 13, there is a sketch showing the positions of the sensors in the building. Most of existing systems present data collected under the form of a Dashboard which will at a glance highlight key features such as curve fitting, trending and alerts.

We wanted to follow the same direction by designing a one-page web site that would also allows sharing of work and information between researchers.

This Dashboard is articulated around three axis: a) Filtering and defining a data subset, b) building the model to analyze, and c) showing different statistical points of view. A fourth axis about learning and predicting is detailed in section 3. Figure 14 shows the top of the application's dashboard.

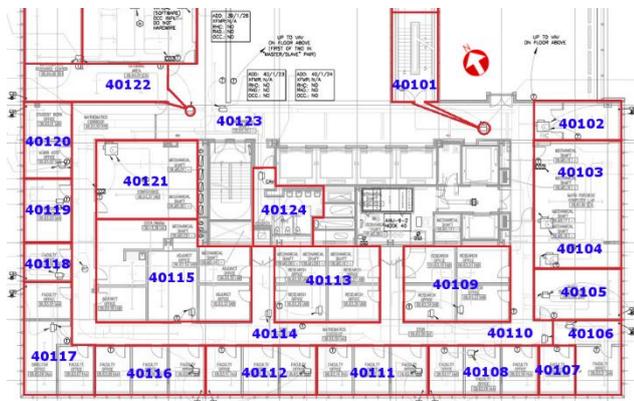


Figure 13: Building Plot

5.1. Showing Data

Graphing is the central feature of our system, under the form of time series curves and frequency histograms, as well as their statistical moments such as moving average, standard deviation as well as correlation, either in a function of time or in a function of other values. At this point, simple curve fitting for model verification is doable, a model being described as an expression, or function, of series.

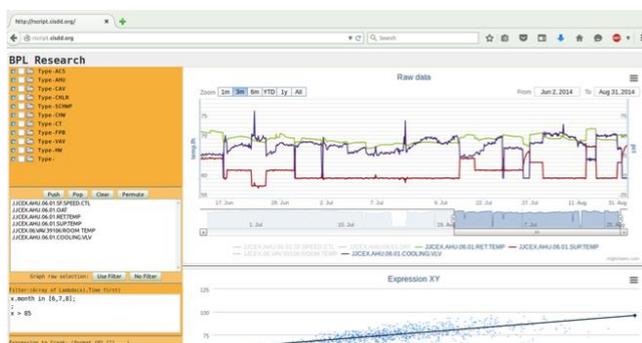


Figure 14: Main page: the Dashboard

All graphs can be tailored, an operator being able to display any type of relation. Indeed it is very important to determine relationships between data, therefore we offer tools to combine

series and display together different data sources mixed with arithmetic operations. Graph widgets are provided by Highcharts [?] to create interactive charts. The application simultaneously shows three graphs, coming from three different origins (from top to bottom):

Raw Data: graph of data not transformed coming straight from sensors, only filtering being possible.

Expression: graph an expression or transformation, by default being the same as raw data.

Forecast: graph an expression, its prediction and the mean square error of the difference.

5.1.1. Display Model

The set of data to be analyzed is reduced down to a set of sensors data projected onto the same time axis. Prior to display, variables can be re-ordered and filtered and only the adjusted set is graphed. Before analysis, a subset of sensors must be defined and each selected sensor is indexed for convenience. In other words, the subset of selected sensors can be seen as a table indexed with the time, and where columns are sensor data. Then every column of the working set is designated by its number. Sensor indexing is implicitly made from the working set by starting from item on top as number 0, followed by number 1, etc. Indexing has the major advantage of simplifying input because sensors are replaced with numeric identifiers, a number is always easier to input than a long sometimes meaningless string. A data series is therefore represented by a variable, statistically speaking. We will later see the other advantages of such a model. Finally, the data model is composed of time series represented by their identifier, such as #0, #1, #2 etc. In addition, a special variable t represents the time axis.

5.1.2. Selection

Sensors are categorized to form a hierarchy tree where nodes are physical location, category and types, and leaves are sensors names.

The working set defines a stack on which simple modifications such as reorder, delete, add and clear (Figure 15) are permitted. The indexing mechanism described above is achieved here, every sensor being substituted by its position within the stack.

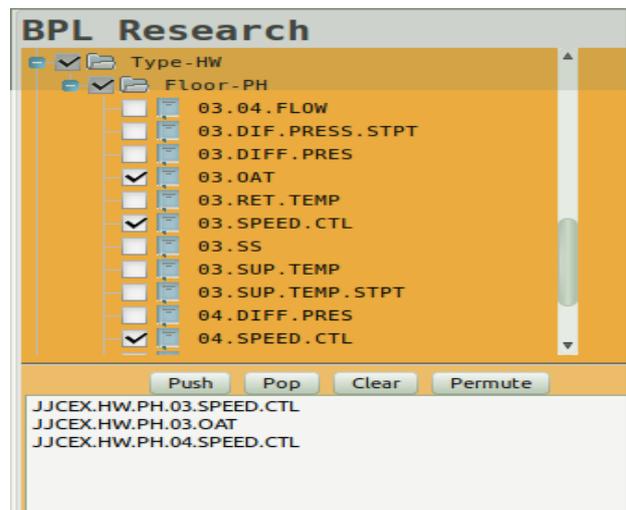


Figure 15: Sensor Selection

5.2. Filters

Behind the scene is Python which has the immense advantage of providing on the y code compilation and also brings a large number

of statistic and machine learning functions. Rather than building an environment from scratch and design an API, we offer users an entry point inside the application server code. Programming is not necessary, nevertheless being aware of the existence of widely documented open source Python modules can be useful. Users have access to a safe sandbox where they can design their own formulas. Among the existing Python computation tools we have included Pandas for the data model, and Scikit-Learn for Machine Learning.

Within this environment there exist filters to limit the size of the observed data set. Indeed, at a rate of a value every fifteen minutes for several years, the database contains hundreds of thousands observations. Dynamic filtering consists in a array of lambda functions, each of one to be applied to every single variable of the input set. Lambda functions are used here to define constraints with boolean expressions, as conditions for which the subset of data extracted must hold.

5.2.1. Timeline

Filtering is done thanks to an array of x lambda expressions, one per sensor, starting with the first element related to the time axis to set some constraints on the time scale common to all sensors.

In Python, a lambda function is a unnamed function of several named variables. By convention, x here is used to declare bodies of such functions. In this context, our lambda variable is a pandas.datetime object, and possesses methods and properties useful to manipulate date and time such as comparison at different levels, from microsecond to quarter. For example, restraining the data set to values observed every day from 2 pm to 5 pm can be expressed as:

```
x.hour in [2,3,4].
```

5.2.2. Data Filters

Same as for timeline, data filters are represented by constraints, on values rather time. They too are declared with boolean lambda functions defining some conditions for which the designated sensor will hold. But this is not all: in fact a filter on a given data will put a condition on the entire output, because the selected values hold for specific periods of time that will be the reference for the other values.

For instance, if item 1 has to be capped at 10: $x < 10$ then the entire graph shows the other sensors values only for the dates determined by sensor1's condition verified.

5.3. Python Expressions

Expressions have a slightly different implementation because they are not reduced to an operation per sensor but they both define the format and the nature of the output. An expression is a list of sub-expressions separated with commas and every sub-expression is represented in the graph. For instance, if the selection stack is composed of four sensors, the output can be any combination of four variables identified from 0 to 3. Thus:

```
{0};{1};{2};{3}
```

will display all the series.

```
But: {3};{1};
```

shows only sensor #3 and sensor #1. In other words, the result displayed is a set of functions of input variables. All kind of operations on variables are available. Through Pandas, any type of arithmetic operation is allowed, e.g.

```
{0} + {1}
```

will display for every time tick the sum of values from sensor #0 and #1. A mean of three values over time can be defined as:

```
(({0} + {1} + {3})/3.
```

Specific behaviour is given to the time axis, called $\{t\}$, typed as a pandas.datetime object, which is consistent with how filters are implemented.

5.3.1. Variables

Variables $\{i\}$ are representations of Pandas time series indexed with the common timeline and created from sensors values. Precisely, behind every sensor there is a two-column Pandas table containing observation dates and value. Sub-expression are computed from format strings where bracketed identifiers (variables) are replaced with their respective sensors tables and evaluated on the y by the server.

Expression evaluation consists of two steps. First a unique table is built from joining the sensor's tables on the time axis, then filters are applied through lambda functions. Afterwards, a format string is built from the expression treated as a string template, where identifiers are implicit references on the positional arguments, the latter being composed of sensor's table values.

For instance, `f0g + f1g` starts with the construction of a pandas table `p` having columns `datetime,svalue0` and `svalue1` loaded with values of sensors #0 and #1. A string `"p.svalue0`

+ `p.svalue1"` is built from the expression template, then the string is evaluated via the python interpreter and `numpy` result is returned to the web page to be drawn.

5.3.2. Pandas Python Module

Furthermore computations on sensors values are made either with Pandas builtins, otherwise Python lambda expressions applied to Pandas through methods such as `apply`.

Pandas also brings useful features for analysis: `shift` and `window`. First, if it is possible to display simultaneously several sensors data for the same time period, we added the possibility to individually shift every series along the time axis, in order to analysis the correlation, delayed in time, between one or several data sets. For example the Return Temperature is a function of the Supply Temperature, but the action of one on another may be delayed by the time it takes to propagate updates, so the correlation between these two variables can be computed from one value at t and the other at $t+(\text{delay})$.

Second, another interesting possibility is to provide rolling statistic moments such as moving average, covariance, skew, box, or user defined. One is therefore able to analyze the dynamic components of a system and the causes of evolution (see below). Last but not least, discrete integration and derivation, rather important features to describe thermodynamic models, complete this toolbox. Examples:

```
{0}.shift(1);{1}.shift(-1): displays series #0( $t+1$ ) along with series#1( $t-1$ )
```

```
{0}.rolling_window(size,operation) : compute a rolling operation on a window of given size.
```

```
{0}.diff(): 1st order discrete derivative with respect to time.
```

```
{0}.diff()/{1}.diff(): 1st order discrete derivative of {0} with respect to {1}.
```

```
{0}.cumsum(): discrete integration over time.
```

5.3.3. Python Plugins

A more advanced feature is the ability to develop Python functions for immediate integration to the application, with the advantage of being also reachable among users. In the mean time this is a simple

way to encapsulate complex expressions in order to make one's work more comprehensible. Above all, the main advantage is to enlarge the scope of analysis to a new level as the plugged-in functions can access to any combination of row/column data, as well as any computation library not limited to Pandas. A plugin is created by uploading a text file of python code onto the server, and can be called via a specific namespace from the expression box, directly applied to variables or as lambda function from a .apply() type of call. And if the signature allows it, the plugin can take several sensors as parameters.

5.4. Bulk Section

Bulk selection, another input box, is a way to add into the sensor stack a set or a list of sensors sharing a pattern of denomination. This is sometimes easier to do than a manual selection, especially if the number of targeted sensors is high. That specific kind of

request is made of three parts: a condition of selection, an operation, and a condition on values. Without going too deep into details of utilization, we can say this feature is used to highlight all the sensors from their characteristics of type and location, for which values have some particularities. For instance, a typical request would be: "get the list of sensors for every room of floor X for which the temperature is greater by Y to the dial temperature". But the output is sometimes difficult to visualize on a simple graph so we privileged the Comma Separated Values format for further analysis with spreadsheet applications. Unfortunately large requests in terms of number of sensors involved are huge CPU consumers.

6. Graph Analysis

Because of the paging, we put figures to the end of the paper. Below are the explanation of the figures.

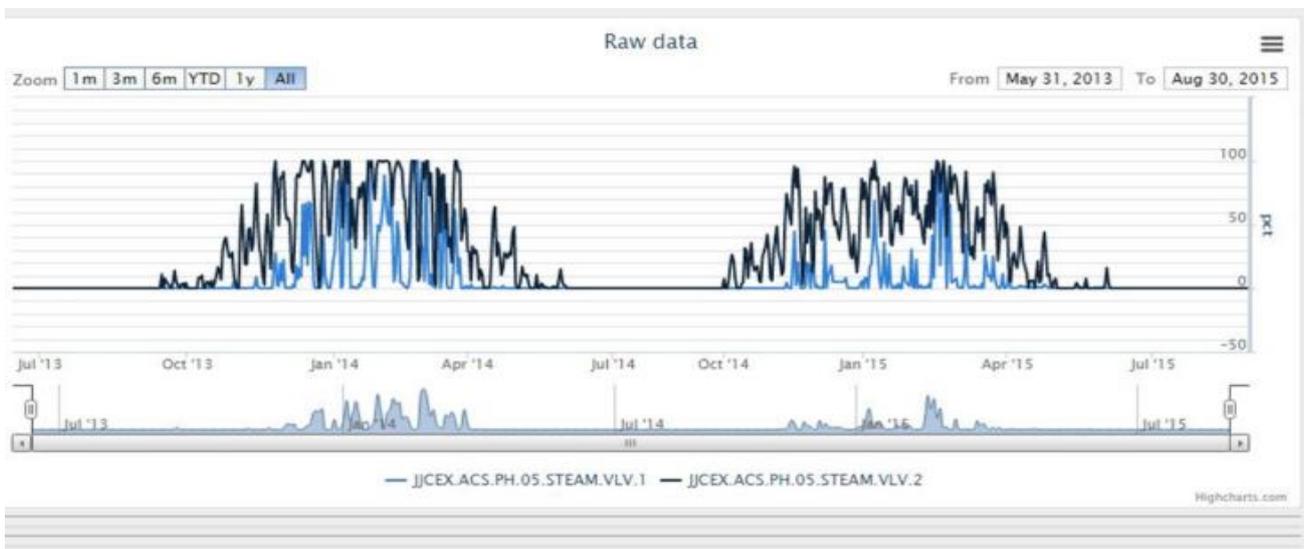


Figure 16: Steam Valve Graphic



Figure 17: Air Volume Graphic

Figure 17 is a graph of the sensor that measures air volume. This sensor was operated from January 26, 2015 to July 26, 2015. The sensor was run on a regular basis for 5 days a week and was not

run for 2 days. On the day it was run, the sensor's operation remained stable. Operation is not always 0 when not running. As you can see in the chart, some days have been run more.

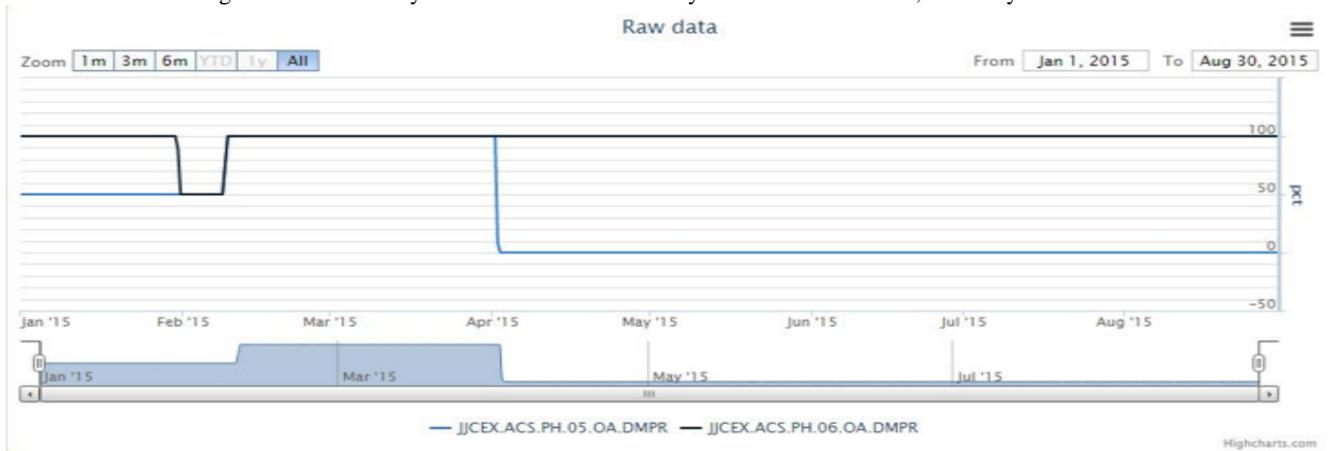


Figure 18: Outside Damper Graphic

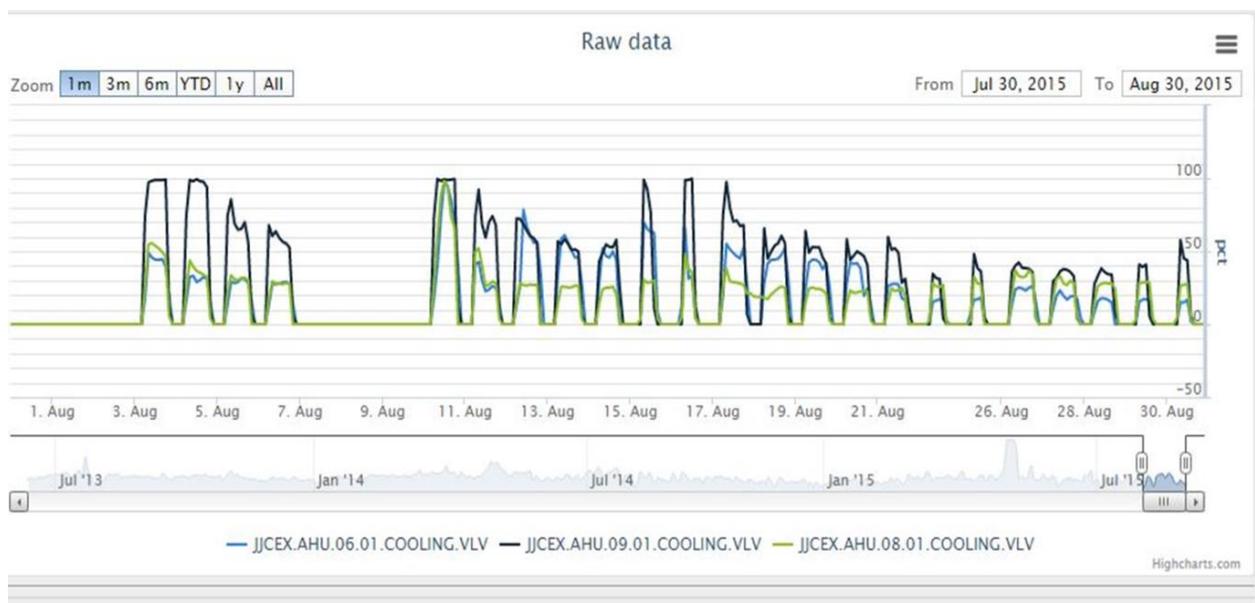


Figure 19: Cooling Valve Sensor Graphic

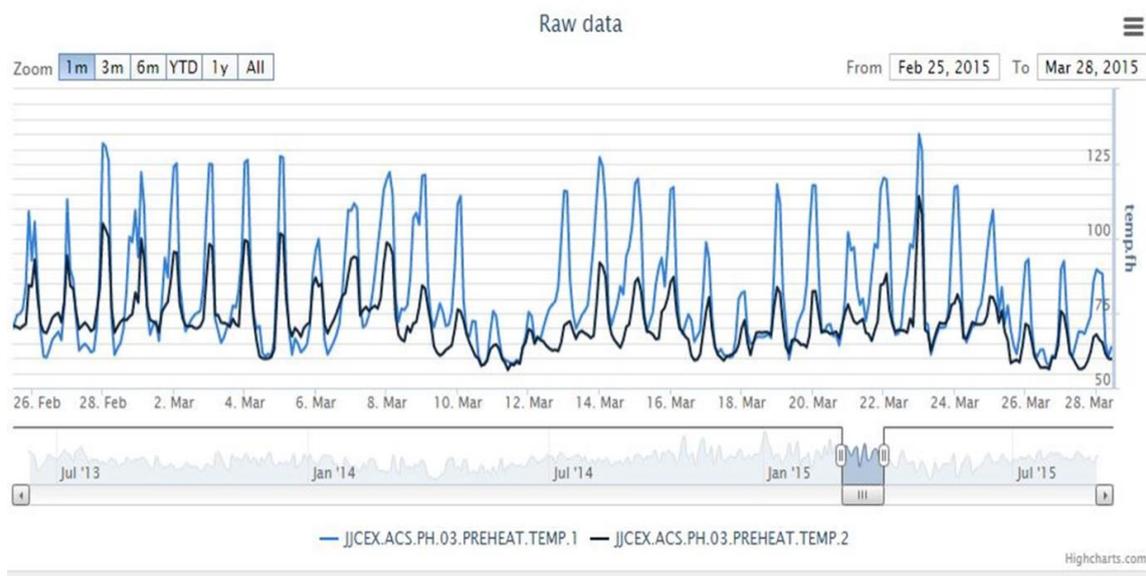


Figure 20: Preheat Temperature Graphic

Figure 19 shows the cooling valve located on floors 9, 8 and 6. The above chart shows the values for August. The sensor on the 9th floor is the most active sensor. The reason for this may be that the summers are warmer than the top floor. Sensors on floors 6 and 8 have obtained data that can be counted at some time. Between 13 August and 22 August, the data of the sensor belonging to 6th kata have

higher values. After August 22, the situation was reversed. Figure 20 belongs to the front temperature sensor. The graph above shows the values of 2 separate pre-temperature sensors belonging to 3rd

column. According to the data from February 25, 2015 to March 28, 2018, the 1st sensor is operated more than 2nd. The reason for this is that the area where the 2nd sensor is located may be seeing more sun than the 1st. Figure 21 shows the data of air volume sensors on the 7th, 8th, 9th and 10th floor. Based on data from April 28, 2015 to May 28, 2015. Sensors generally have the same level of data. On May 10, the increase in data display of all sensors came to fruition.

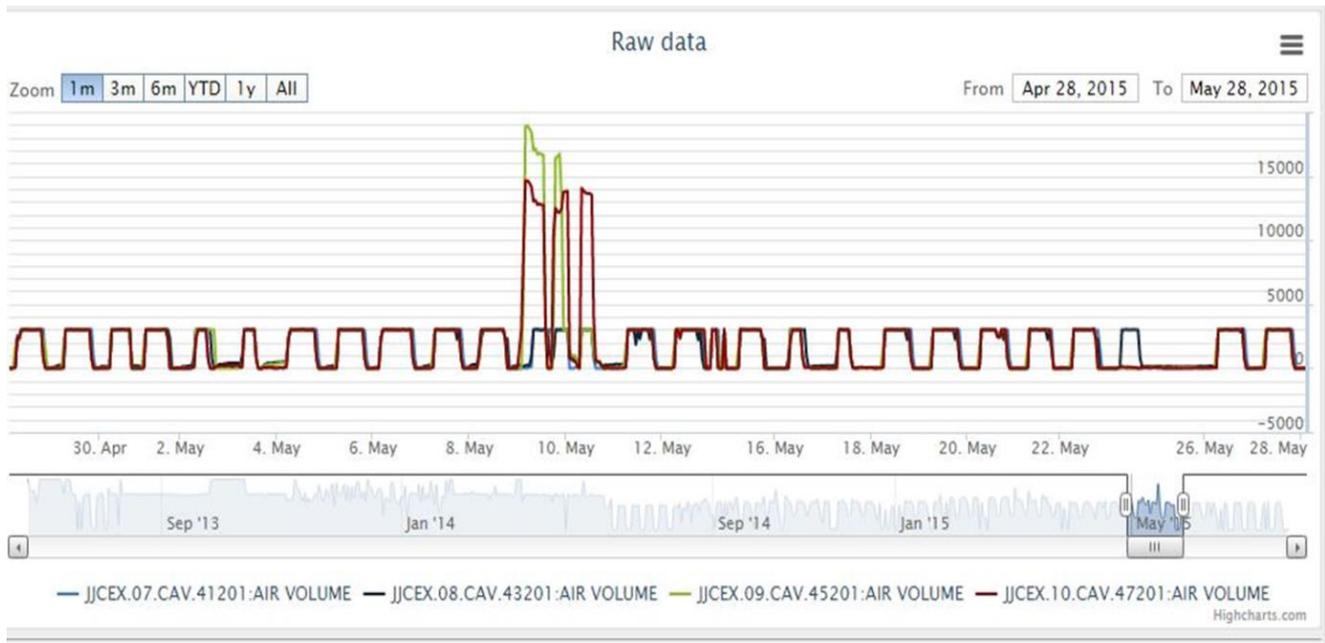


Figure 21: Air Volume Graphic

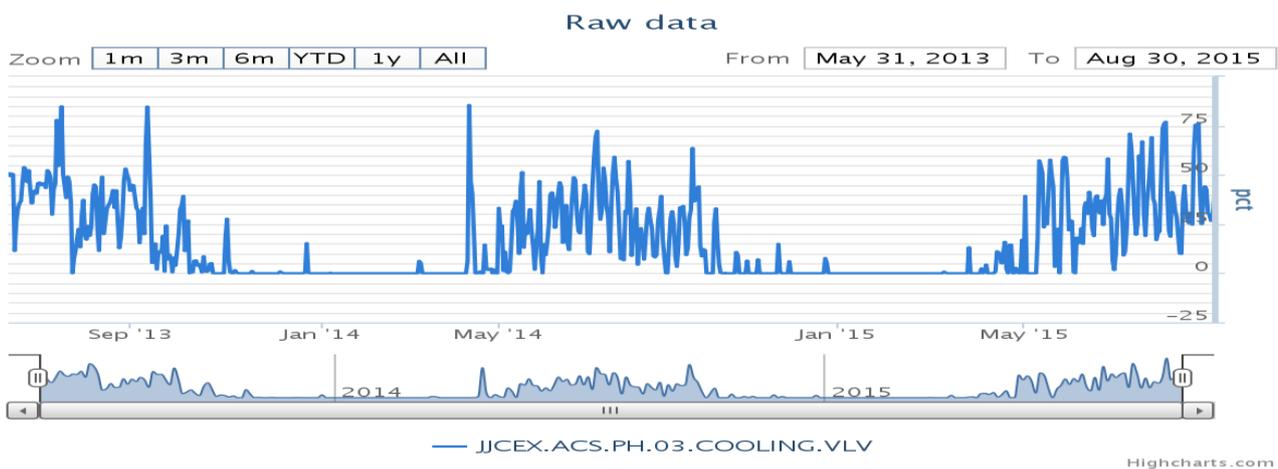


Figure 22: Cooling Unit Sensor Graphic

Figure 22 shows an image of the sensors in the cooling units of 31.05.2013 and 30.08.2015 period. It can be said that the values obtained from the sensors in April and October are higher in the data of 3 years, while the values obtained in the other months are lower. Figure 23 shows the engine operation chart of the No. 1 cooler. It is seen that in the first period when the values of the existing system are transferred to the graph in the graph, the work value of the motor is irregular. Once the current regulations have been made, it appears that the system is working steadily. Figure 24 shows the operating values of the fan of the air control system

according to the hours. According to this chart, the fan speed is increasing by one near values and by value.

In Figure 25, the value of the increase in fan speed is much higher as the clock progresses. Although it is not a fixed value, the change in speed is much more than 2 hours ago. In Figure 26 the fan speed continues to be constant after 07:30. The value is both higher and stable compared to nighttime. Looking at the 3 time and speed values above, we can say that the fan does not stop working at night, but works at the same speed and higher values in the morning hours.

Raw data



Figure 23: Chiller Motor Run Graphic

Timestamp	JJCEX.ACS.PH.05.SF.SPD.CTRL
01/01/2014 02:30:00	75.26
01/01/2014 02:45:00	77.4
01/01/2014 03:00:00	79.38
01/01/2014 03:15:00	81.16
01/01/2014 03:30:00	84.48

Figure 24: Fan Speed Control Graphic

Timestamp	JJCEX.ACS.PH.05.SF.SPD.CTRL
01/01/2014 05:00:00	99.79
01/01/2014 05:15:00	99.36
01/01/2014 05:30:00	100
01/01/2014 05:45:00	99.84
01/01/2014 06:00:00	99.64

Figure 25: Fan Speed Control Graphic

Timestamp	JJCEX.ACS.PH.05.SF.SPD.CTRL
01/01/2014 07:30:00	100
01/01/2014 07:45:00	100
01/01/2014 08:00:00	100
01/01/2014 08:15:00	100
01/01/2014 08:30:00	100

Figure 26: Fan Speed Control Graphic

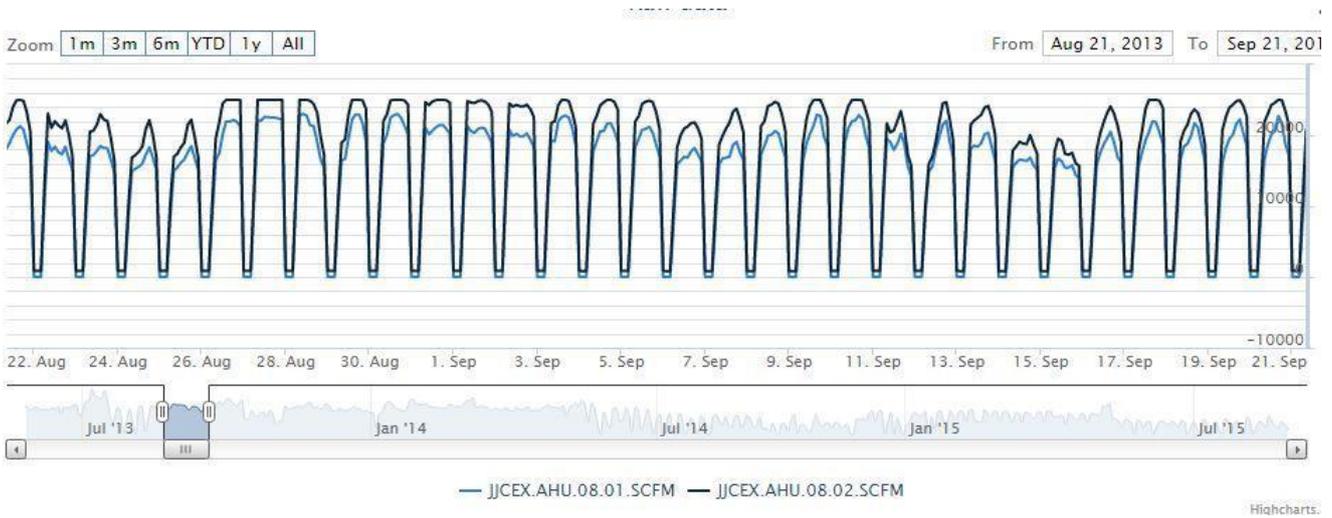


Figure 27: SCFM Graphic

Figure 27 shows the standard feet per minute values. Calculated for flow and speed control. In the above chart, the values of two different sensors are visible. 1. The values of the SCFM graph are higher than those of the 2. SCFM graph.

TimeSeries: regular time series.

XY: merges two data series along time axis and displays. one as function of the other, adding linear regression to the picture.

Correlation: This is a quick access to correlation computations, the graphs displayed here are the rolling correlations of every sensors compared to the first one

Histogram: Displays the distribution relative to the rst variable.

Moving Std: moving average accompanied by two standard deviations. This can be used to detect an abnormal behaviour of a sensor; for instance, when current values move far from the moving average and cross over the two standard deviations curve.

7. Computational Tools

7.1. Graphic Tool

By default, the application doesn't require a deep knowledge of Python, because it allows to create several views on data without having to define an expression. Different types of graphs are available to highlight different type of information. A graph type is configurable through a menu containing these choices:

Expression

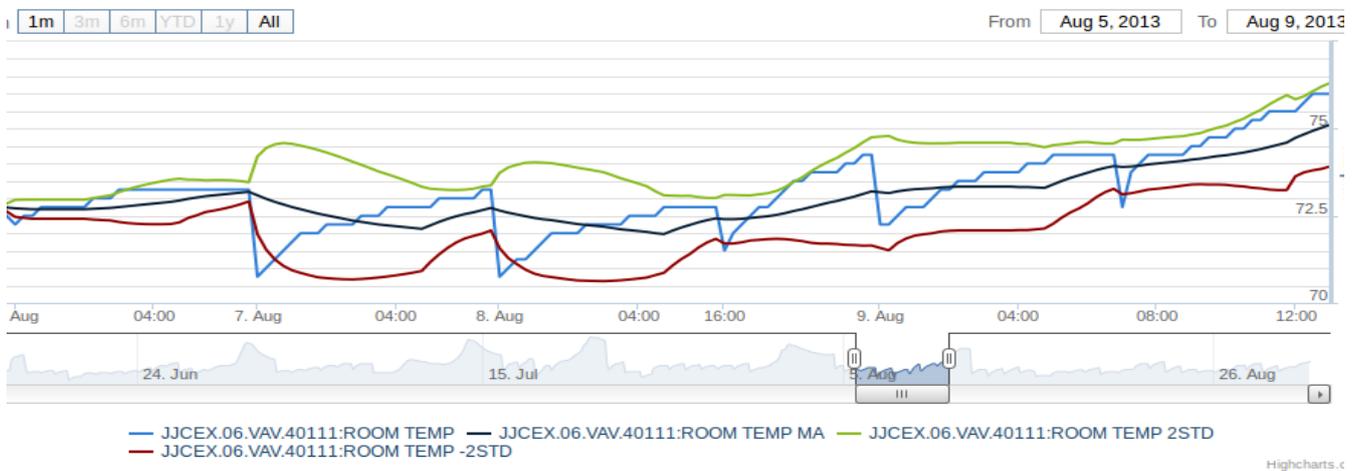


Figure 28: Example of MA and Standard Deviations: Blue curve is a room temperature, the Black one is the MA, the Green and Red are respectively +2STD and -2STD

Table: Displays a table, that can be sorted or exported into a spreadsheet for a different type of analysis.

7.2. Moments

One of the great advantages of Pandas is to provide rolling moments. Moving average, covariance, skew, and generic window are available. Users are therefore able to analyze the dynamic components of a system and analyse the causes of evolution. All pandas computation tools are accessible directly from the application through expressions. To mention a few, statistical

moments such as mean, variance, skewness and kurtosis as well as correlation or pairwise moments can easily be used. For example:

`pandas.rolling_var({0},10)`: Displays the variance on a rolling window of 10 values.

`pandas.rolling_cov({0},{1},50,True)`: Displays the rolling covariance of {0} and {1} on a rolling window of 50 values.

{1};pandas.ewma({1},10): Displays sensor {1} along with its exponential moving average.

7.3. Time series analysis with ARCH

We wanted to add auto regression features by importing ARCH python module for two reasons: rst of all, if auto regression is normally used for nancial series analysis, but we found interesting to apply this model to sensors, because of the similarities in terms of time series properties. The other reason was to prove it is quite easy to extend the set of computational tools offered by the application. For now, we are only using a very simple model called GARCH (1,1), from which we compute the conditional variance. The implementation was very straightforward and took a couple of minutes: a few lines of Python to create a function immediately exportable into the expression box.

Directly using GARCH (1,1) with building sensors is not necessarily giving a lot of information, but we noticed that given as input to machine learning system improve sometimes a lot the predictability. More details on this in the next section.

8. Machine Learning, Forecast And Modeling

One interesting thing about this application is that users have a quick access to some machine learning capabilities, to rapidly determine a relationship or a dependence between different types of thermodynamic entities, or also to validate any physical model, without prior knowledge of Python programming. Plus those two aspects of the Research process can as well easily shared to other researchers.

Machine learning consists in two phases: training and forecasting. The training phase is done with a subset of parameters using the sensor selection and the ltering mechanism describe above, the result of training being persistent. From there, a forecast or run is performed either with different parameters, as long as it is consistent with what has been used for training, and/or different set of values. Results of training or forecasting are displayed on a speci c graph along with the MSE of the difference between observed and predicted values. Figure 29

shows an example of training on a subset made of sensors #0 and #1 for the month of June 2014.

8.1. Input and Output

Two types of input are necessary for machine learning: parameters and values. The parameters set is quite succinct, a moving average window size, a percentage representing the training set size and a machine type. Values are defined through a series of Python-like sub-expressions similar to what is described in section 3. Similar because the syntax is the same, but now the order of the sub-expression has a meaning.

Data input is therefore represented as a list of sub-expressions separated by commas, where the rst sub-expression is the output target, and the following ones are the observations. For example: if the expression typed into the machine learning input box is:

```
{2};{0}+{3};{1}.diff()
```

The rst sub-expression is {2} meaning sensor #2 is the object of learning; then follows: {0}+{3};{1}.diff() which means we want to fit sensor #2 as a function of a) the sum of sensors #0 and #3 and b) along with the differential of sensor #1 (with respect to time).

Now this model can be fine tuned by playing on MAW, i.e. the moving average window size, as well as the machine type, and the size of the training set.

8.2. Machine Learning models

The machine learning capabilities are still in a trial phase so for now we limit the set to four regression types of predictors.

SVM rbf: support vector machine with radial kernel.

SVM Poly3: support vector machine with cubic polynomial kernel.

Logistic Regression: log-linear regression.

Gaussian: Naive Bayes based on a Gaussian likelihood.

With this sample of different techniques, results so far are quite promising. All estimators or classifiers come from the Python Sci-Kit libraries and are based on default parameters for simplicity. Adding more algorithms to the server is pretty straightforward, as the Sci-Kit API is very consistent.

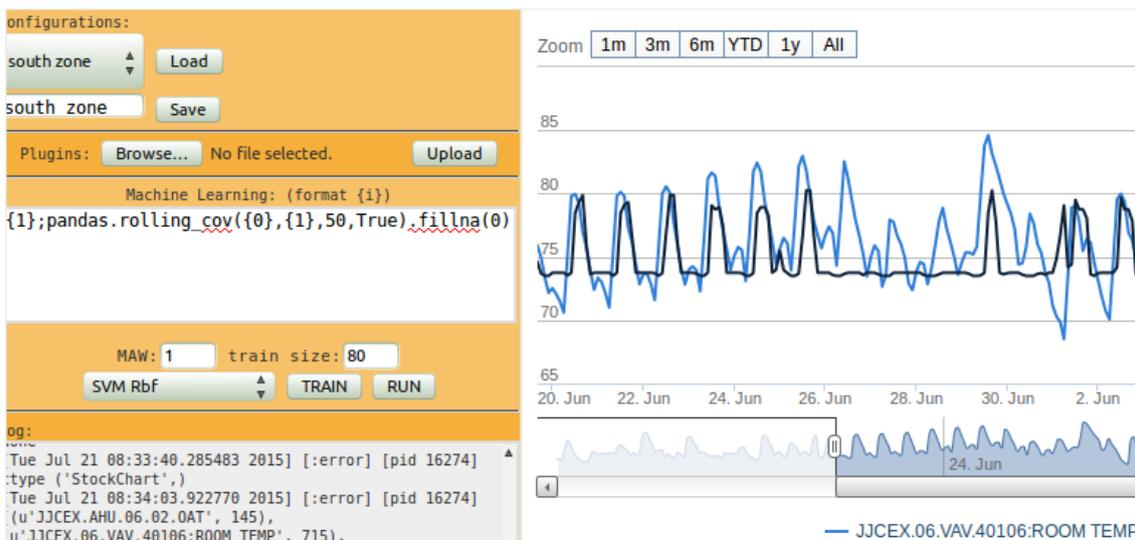


Figure 29: Machine Learning parameters

8.3. Forecasting and results

To do prediction, we look at the data as time series, and, similarly to quantitative finance models themselves based on Markov models,

future values are estimated from past data. Indeed, a rolling window of fixed sized observed at time t determines an estimation of next value, at t+1. To reduce the number of parameters, past data are reduced to one point, a moving average. Therefore, a prediction

$p_{e0}(t+1)$ computed with sub-expression e_0 can be seen as a function:

$$p_{e0}(t+1) = M(\text{maw}_{e1}(t; s; t); \text{maw}_{e2}(t; s; t); \dots; \text{maw}_{ei}(t; s; t); \dots)$$

where $\text{maw}(t; s; t)$ is a moving average of size s . For a given Machine Learning session, three curves are drawn into the 'Forecast' widget: The observed value to learn, the actual output computed by the machine, and the moving mean square difference. As said before, the machine learning process has a training phase and a prediction phase. Once the training done, its parameters are saved and a prediction can be made from there with a different set of data, either from different variable or a different time period.

8.4. A case study

The following case study is an attempt to forecast the temperature in a given room during the day. Precisely, as the time scale consists of 15 minute ticks, the goal is to guess what will be the room temperature in the next period given present values of: fan speed, outside temperature, temperature measured at the AHU level (supplied and return), the actual room temperature, the cooling valve opening in percent. Figure 3 shows the selection stack.

Predicting room temperature can be helpful for energy saving: the actual AHU system on which that study is based tends to over correct, and the target temperature is reached after several oscillations above and below. Forecasting temperature is a way to approximate the thermodynamic model and directly obtain a target value. The data study is made over the months of June, July and August, over which temperatures are in the higher range of values. Next figure illustrate how the data set is restricted to a summer period of time.

The next step is to define input and output. Here we would like to know if room temperature can be determined from the following parameters:

- Output is {4}: it is the sensor #4 i.e. temperature in room 39106.
- First input value is {3} meaning the AHU return temperature.
- Second input value is {t}.hour%4: it is the hour of observation modulo 4, a period of the day roughly mapping the sun course over 24 hours.
- Third input value is {4}
- {1} is sensor #1 so outside temperature. {0} corresponds to the AHU fan speed.
- $\text{cond_vol}\{3\}$ for the conditional volatility on a GARCH(1,1) of the AHU supplied temperature.
- $\{5\}.\text{fillna}(0)$: cooling valve percent where NaN values are replaced with 0; small problems with raw data may happen.
- {4}: previous period temperature in target room.

The parameters selection is defined by an expression shown Figure 6, as well as the moving average, and the training set size; MAW here is set to 1 because the conditional volatility used as input is already a moving value, and it is comparable to a variance, so taking its moving average doesn't make too much sense. A value of 70 for train size means the training will use the first 70% of the data set. Finally, a SVM with radial kernel will be the machine used for this experiment. After training, machine learning parameters are automatically saved and can be used to predict either the temperature of a different room, or for a different time period. For example running the forecast using room 39107

is depicted Figure 30. Graphic prediction seems pretty accurate, which is not surprising as the new room is very close to the one used for training thus has very similar thermodynamic properties.

9. Conclusion and Future Work

Thanks to the developing Internet of Things (IOT) technologies, it is possible to intervene in advance in case of problems that may arise from analysis of long term data, as well as being able to find faulty remote disturbances in the system. On the other hand, it can be possible to eliminate the problems that may occur by interfering with the energy expenditure statistics of the system, intervention processes to the problems that occur, repetition frequency of the same problems, how long the solutions are produced in the problems that arise, and problems that can be anticipated in the system by analyzing these resources appropriately. Especially in large-scale organizations, it seems that such measures provide significant advantages to institutions in terms of cost and time, in situations where effective use of human resources is required. From the data in this system belonging to CUNY, by way of modeling the working structure of the system with machine learning algorithms, a more energy-efficient automation system can be created. The data can be intelligently programmed by combining the parameters of each room with the number of windows, size, geographical location, and weather. The temperature, humidity and pressure values at the site can be reached by referring to the system. With the industry 4.0 process, building automation systems are getting more and more important in this process, where renewable energy and efficient use of human resources are valued. In our next work, we plan to determine the data transfers and calculations that the building automation system can automatically start and repeat.

This paper presented a system used for analyzing and building models from data observed on thousands of sensors. Results are encouraging as we could explain and approximate a few thermodynamic behaviours, but yet that system still lacks of intensive exploitation. Lots of improvements are necessary, especially in terms of user friendliness and performance. But we think it is flexible enough to offers a large choice of possible utilization and would necessitate from now on only minor developments. We think still a few concepts are missing: for instance, to store sub-expression results into variables for later reuse could be very efficient. Same thing with results of training or forecasting, to generate pipelines of machine learning computations and then combines models together. Another point is performance wise some SVM needs a lot of calculations and could benefit of parallel architecture. Same remark about bulk selection, which implementation needs to be optimized if not completely rethought. Finally, we may now focus on the machine learning part and modeling based on the possibility of energy savings using the system.

After this work, we will try to predict how the system will behave with machine learning algorithms using the system's data and weather data. In the next step, we will investigate whether or not energy saving can be done as a result of the behavior of the AHU fans in the system being programmed according to these results.

Acknowledgement

We would like to thank Rojda Bozkurt, Kemal Atalı and Betül Aydın, students from Yildiz Technical University and Zahide Nur

Ercan from Istanbul Sabahattin Zaim University for their help with the formatting of the paper.

References

- [1] Çimen, L., (2005) Bina Otomasyon Sistemleri, Türk Tesisat Mühendisleri Derneği Dergisi.
- [2] Karaca, H., (2002) Otomatik Kontrol Ve Otomasyon Sistemlerinin Vazgeçilmez Elemanları: Sensörler, Türk Tesisat Mühendisleri Derneği Dergisi.
- [3] Ercan, M. S. (2009), Bina Otomasyon Sistemleri İle Devreye Alma İşlemleri , IX.Ulusal Tesisat Mühendisliği Kongresi, 1121-1127.
- [4]Yılmaz, Z. (2005). Akıllı Binalar Ve Yenilenebilir Enerji. Teskon Konferansı, İzmir. Teknokulis. 01.03.2016. [Türkiye Akıllı Şehirler Değerlendirme Raporu](#) Yayınlandı.
- [5]<<http://www.teknokulis.com/haberler/guncel/2016/03/01/turkiye-akilli-sehirler-degerlendirme-raporu-yayinlandi>> Erişim :31 Mayıs 2016
- [6]Kaşıkçı,S.(2015). Sensörler. Slideplayer. <<http://slideplayer.biz.tr/slide/2792904/>>. Erişim:31 Mayıs 2017
- [7]Necoinside. (2015). Ds18b20 Dijital Sıcaklık Sensörü.<<http://necoinside.com/>>. Erişim:31 Mayıs 2017
- [8]Günacar, G. (2011). Türkiye Ve Dünyada Akıllı Binalar.<<http://www.pckopat.net/turkiye-ve-dunyada-akilli-binalar/>> Erişim 31 Mayıs 2016
- [9] Utkuğ, G. (2011), Sürdürülebilir Bir Geleceğe Doğru Mimarlık Ve Yüksek Performanslı Yeşil Bina Örnekleri.
- [10]Kırmızıoğlu, E. (2014). Ülkemizin 2023 Stratejik Vizyonu Doğrultusunda Akıllı Şebekeye Geçilmesi İçin Öneriler. 2. Uluslararası İstanbul Akıllı Şebekeler Kongre Ve Fuarı . 143-147.
- [11] Civan, U. (2015) Akıllı Binaların Çevresel Sürdürülebilirlik Açısından Değerlendirilmesi. Diss. Fen Bilimleri Enstitüsü.
- [12]Gazioglu, A., Akşit,Ş.F. ,Manioğlu,G. (2013). Enerji Etkin Bina Tasarımında Isıtma Enerjisi Tüketimini Azaltmaya Yönelik Bir İyileştirme Çalışması
- [13] Manioğlu, G.(2007). Geleneksel Mimaride İklimle Uyumlu Binalar: Mardin'de Bir Öğrenci Atölyesi, VIII. Ulusal Tesisat Mühendisliği Kongresi Ve Sergisi, 79-92
- [14] Yılmaz, Z.(2006).Akıllı Binalar Ve Yenilenebilir Enerji, Tesisat Mühendisliği Dergisi,91,7-15.
- [15] Canbay, C.S. , Akkurt, G.G. , Hepbaşlı,A. (2003), Bina Yönetim Sistemleri Ve Hvac Sistemlerinde Enerji Tasarrufuna Yönelik Kontrol İlkeleri, V1. Ulusal Tesisat Mühendisliği Kongresi Ve Sergisi, 653-671
- [16] Karaca, H. , Otomatik Kontrol Ve Otomasyon Sistemlerinin Vazgeçilmez Elemanları: Sensörler, Türk Tesisat Mühendisleri Derneği Dergisi.
- [17] Cai, J. and Braun, J. (2012) “Efficient And Robust Training Methodology For Inverse Building Modeling And Its Application To A Multi-Zone Case Study” International High Performance Buildings Conference, Paper 98. <http://docs.lib.purdue.edu/ihpbc/98>
- [18] Inno, Y. and Masakiko, M. (2013) “Physical And Jit Model Based Hybrid Modeling Approach For Thermal Load Prediction” Electrical Engineering In Japan V.85 No.2
- [19] Zhou, Q. And Shangwei, W. (2008) “A Greybox Model Of Next-Day Building Thermal Load Prediction For Energy Efficiency Control” International Journal Of Energy Research V.32 1418-1431
- [20] Oldewurtel, F. And Andreas, U. (2010) Reducing Peak Electricity Demand In Building Climate Control Using Real-Time Pricing And Model Predictive Control. Proceedings Of The Ieee
- [21] Oldewurtel, F. And Alessandra, P. (2012) “Use Of Model Predictive Control And Weather Forecasts For Energy Efficient Building Climate Control” Energy And Buildings V.45 15-2
- [22] Li, P., O’neill, Z. and Braun, J. (2013) “Development Of Control Oriented Models For Model Predictive Control In Buildings” Ashrae.
- [23] Zhao, H. X., & Magoulès, F. (2012). A review on the prediction of building energy consumption. Renewable and Sustainable Energy Reviews, 16(6), 3586-3592.