Energy Analytics 4 Plan Proposal

Miguel Alvarez, Dominic Enriquez, Henry Koelling,
Minh Pham, Ben Preiss, Nacho Rodriguez-Cortes

Fall 2021

1 Summary

Entering the fourth iteration of the *Energy Analytics* computer science comps project, we aim to build on the foundation of previous work completed by the former groups in a variety of ways. Our ultimate goal is to produce an analytical tool with a user interface catered to the needs of the Facilities team whom oversee the usage of energy in Carleton affiliated buildings on campus.

Previous groups have done an excellent job in crystallizing an automated data pipeline which receives energy consumption data from *points*[1] Along with establishing the pipeline, they have made great strides in interpreting the naming scheme of such points, and have written descriptive documentation detailing how to query such points from the server where the data is stored. Initial steps have been taken to present a user interface displaying anomaly detection of points and energy usage trends capable of being sorted by characteristics of the points.

We propose to improve the professionalism of the data storage and website interface, and provide more extensive user features with the website that allow for the user to easily query points around campus and quickly identify anomalies around campus. Our improvements can be broken up into four

[1]A *point* is a value that can be observed remotely (e.g. a room's temperature) or controlled remotely (e.g. a fan's speed).

categories: **Website Interface**[2], **API**[3], **Data Management**, and **Data Analysis**. In the following sections, we will describe in depth what we plan to improve for each category, and we will propose an order in which we will complete the tasks.

2 Improvements

With the amount of work that's already been done on this project, it's important to be organized and focus on the areas where the most work can be done and be beneficial. Ultimately, we think improving our data wrangling and management, such as cleaning up the .CSVs and point names, to be an important start to our work. This should give us a strong foundation that will help streamline our work further down the line on things like data compression & backup, as well as the responsiveness and clarity of the front-end of the website. Additionally, becoming more comfortable and knowledgeable with Brick as our data gets more organized should help us take advantage of any of Brick's analytical tools that might be useful to Martha.

Another area of improvement that can be done throughout the project as an ancillary task is the cleaning up of the example queries and descriptions, as well as the syntax of the API documentation. While not the highest priority item given the lack of action steps that depend on this cleaning, as well as the time independent nature of this task, it is nevertheless important. Making sure that the API documentation is as clean and presentable as possible will ensure that we are able to provide the necessary professionalism to deploy this software on a production level, while also ensuring that those who come after us are able to understand the inner workings of this software as best as possible.
The biggest hurdle to preventing this task from being completed is that it is currently unknown how to access and edit the Postman. It is presumed that there exists a repository that stores the code for this documentation, however it has yet to be discovered. Once we know how to edit the Postman documentation, this task will be relatively straightforward.

[2] The current website interface can be accessed at `http://energycomps.its.carleton.edu` when connnected to the Carleton VPN.

[3] Documentation of the API can be found here.

2.1 Website Interface

As stated previously, our ultimate goal is to produce an analytical tool with a user interface catered to the needs of the Facilities team whom oversee the usage of energy in Carleton affiliated buildings on campus. Improving the current website is imperative in order to achieve this goal. Our ideas for improvement will help with up-scaling the front-end, contextualizing ambiguous data and general information, and provide and analytical tool that will be of use for Martha and the facilities team. We identified a few key goals and areas of improvements for the website:

1. Currently the front-end of the website could use some improvements in order to make the data more clean, understandable, and useful. For example, the graphs could be a bit more interpretable with things such as "average room temp" as a line on a graph that shows the room temperature of a room on campus in order to create a sort of standard for that particular measurement/building. Furthermore, we'd like to create a more responsive user interface when selecting data to plot which includes expanding/removing options that are not related to the current data we're attempting to plot.

2. Make sense of the anomaly algorithm and data by potentially talking with Owen and figuring out if continuing/building on his work would be of great use to Martha. The anomalies table is a bit hard to understand so we'd like to understand if this is something worth pursuing after meeting with Martha.

3. After up-scaling the front-end we wish to add the rest of the data and continue adding more and more buildings to the current project.

The biggest goal we have as a group is to deliver features or an analytical tool for Martha and the facilities team that tailors to their needs. To begin, we need to upscale the front end with the data we have right now, and then add as much data as possible.

2.2 Data Management

2.2.1 ALC Data

Accessing data from ALC-based buildings on campus is a primary focus in our early steps of the project. Attaining this data will allow us to compare data analysis to that of Siemens-based data which may shed light on the importance of switching Siemens-based buildings over to ALC.

2.2.2 Brick

At the end of the day, the naming of points is an inconsistent, messy process, and understandably so. There is a plethora of information required to accurately categorize the thousands of points on this campus. This is where Brick comes into play. Rather than rely solely on the naming of each point to describe the location, type of measurement, and other such important information that specifies which point is being looked at, Brick utilizes a graph theoretic approach to establish connections between different entities within the Brick Schema, in which the *metadata*, rather than the particular point name, tells most of the story. An entity is the abstraction of any physical, logical, or virtual item. These are the actual "things" that make up the buildings. These entities are then *tagged*, which is to label the entities with atomic facts or attributes. These entities are also typed through the use of classes, which are named categories with intentional meaning. Any entity is an instance of one or more classes, and there is a hierarchy to the classes. Finally, these entities are tied to one another via relationships, which are any links between entities. In essence, each entity represents a node in a directed graph, whereas each vertex represents a relationship. This graph then allows one to not only get a sense for the organization of the building data from a more spatial perspective, but also allows for the efficient querying of points and entities based on their attributes. [4]

We are in the process of exploring different tools and technologies that we feel would further amplify the utility of making the switch to the Brick Schema. For instance, the website Brick Studio was explored by those who worked on this project before us. This website takes in the Schema in a *.ttl* file extension format, and then provides a visualization of the graph that the Schema describes.

Another tool that can be utilized is the python library py-brickschema, which allows for the analysis of the aforementioned .ttl files, specifically allowing for the querying of points based on the specified criteria. (building, point type, etc.) through the use of python scripting. This library in particular has potential as a way of uniquely conducting exploratory data analysis on

[4] Information regarding the general overview of the Brick Schema sourced from https://docs.brickschema.org/brick/concepts.html

the Carleton building system and its points.
Quoted from this website, https://www.esmagazine.com/articles/101566-how-brick-schema-defines-the-digital-landscape-of-smart-buildings, the Brick Schema is also promised to have these other important tools:
BEGIN QUOTE
With help from engineers, this sensor-created data, along with the Brick metadata, allows owners and operators to:

1. Create and use a building's digital twin – A digital twin is a virtual representation of a building and its assets that can be used to gauge a building's present state and its past benchmarks. Using artificial intelligence (AI), it can offer predictions of what is needed for the building in the future. A digital twin is created by combining both the metadata, ongoing sensor data, and other information into a digital representation that allows for deep analysis. Many digital twins are also developed into an adjustable 3D visualization of the facility, which has a variety of uses.

2. Analyze occupants as well as assets – As mentioned, the metadata used in Brick Schema allows information about the system to be grouped and sorted, which creates opportunities for analysis of efficiencies, operations, and functionalities. Using fobs, badges, or other individualized IDs that serve as a Brick asset, the actions and locations of occupants can also be gathered, analyzed, and acted upon. This allows the building operator to offer occupant-friendly services in real time, such as automatically opening doors or helping coworkers locate one another.

3. Recognize use patterns – The data generated by Brick-enabled building systems and occupants can show how parts of the building are being used, which allows for predictive planning of the building's maintenance, cleaning, and general operations, including the use of energy intensive assets like the HVAC system.

4. See the big picture. With a detailed metadata map, an ongoing collection of data from building assets, and the ability to communicate with those assets, control and analysis of the building as a whole can be centralized through an operations suite, such as Johnson Controls' OpenBlue. This is the ultimate promise of Brick – that data coming from assets with Brick allows operators to see the big picture of their

buildings and act upon it in a unified, highly efficient manner to build occupant satisfaction, achieve sustainability goals, and improve budget performance.

END OF QUOTE
At its core, it appears as thought the overall benefits of the Brick Schema and the subsequent analytical tools that we can utilize from implementing the Brick Schema can be divided into two different groups: data visualization, and point querying. The question then becomes: are the uses of point querying and point visualization impactful enough to warrant the implementation of the Brick Schema on the production level. This question is one that we must explore further to truly see the benefits of implementing the Brick Schema.
In summary, want to standardize our data utilizing the Brick Schema for the following reasons:

- Clean up messes in the data
- Organize the data systematically, so people working on the project in the future can understand it better
- Allow us to utilize analytical tools out there that can process data in Brick formats. Specifically, we want to

2.2.3 Data Compression/Backup

What we aim to do is process the data right away after obtaining it, process it in the format for live tool, then save the raw data. Currently, we are not using most of the data, but we want to.
Our main goal for the compression and backup of the data is first to backup the data that we do have, followed by developing a strategy for backing up data in the future. New data is fed in regularly to the archive folder. We will leave it that way, so to backup the data we currently have access to, we believe that we should create a new directory /var/data/backups where we can store tar files for each building. Additionally, we should either move the uncompressed data files to a new directory or remove them altogether after they have been backed up. This will keep the uploads directory up to date with the files that have not been backed up, making it easier to manage which files need to be backed up. Moving forward, we need to implement a

strategy for future backups. One option is to compress the files for a given building in uploads to a tar file once every month or few weeks and simply add the tar files to backups. However, if we want to limit the amount of tar files in the backups directory, then we could decompress a given tar file, add the new files, and re-compress them. Having a multitude of different tar files in backups will likely not be as space efficient as larger tar files; however, a multitude of different tar files lends itself to easy retrieval. Either option will be an improvement upon what is currently in place and accomplish the goals we need it to accomplish.

2.3 Data Analysis

2.3.1 Understanding Current Figures

Currently, the website produces graphs that take point data and graph selected metrics over time. This is a great feature, however, the labeling of the units are not present and it is hard to discern the scale of such units as there is no comparison feature. Additionally, there is an anomaly detection feature on the website that uses k-means clustering to detect anomalies which is in need of contextual labeling. Additionally, the implementation of the k-means clustering algorithm should be investigated to better understand what exactly is being displayed.

2.3.2 Heat Recovery System Analysis

Once data is obtained from ALC buildings, we can apply data analysis methods to determine whether the *Heat Recovery ventilators* are working adequately. To do so, research will have to be done in order to understand how these HRVs work and how we can analyze the data to determine the status of their functionality.

2.4 API

Ideally, our website will be robust enough so that users may enter advanced queries to their liking, and a query-based website must be accompanied by adequate API documentation. Maintaining and improving upon the clarity of the documentation is key as we move forward in implementing new features to the codebase. Right now, the biggest barrier for this gaining access to the **Postman** account where adjustments can be made.

3 Order of Tasks

- Obtain ALC data from all buildings currently using that system and familiarize ourselves with the data analysis options it offers.
- Compress and backup the data we currently have access to and define our procedure for backing up files as we move forward.
- Develop non-alignment heuristics (heat recovery, dcv, slider/data disconnect)
  - Focus on top 5-10 most frequent/large anomalies rather than trying to catch them all.
- Check out Skyspark and Smart Energy Analytics Campaign
- Clean up data .CSVs (maybe not as high priority with ALC switch)
- Improve website query responsiveness and data visualization.
- Explore Weitz system (Desigo), maybe improve upon it, maybe ignore it until they get it on a better system?

4 References

https://docs.brickschema.org/brick/concepts.html