influxdata®

AN INFLUXDATA CASE STUDY

How the Aquicore Solution Cuts Energy Waste and Improves Tenant Comfort Levels with InfluxDB Cloud

Mike Donovan

VP of Products, Aquicore

aquicore

MAY 2019

Company in brief

Aquicore is the leading IoT-driven smart asset operations platform for commercial real estate. The only industry player to be built from the ground up as a portfolio-wide solution, Aquicore offers a cloud-based platform to track, analyze and predict building and energy data in real time to unlock portfolio-grade insights.

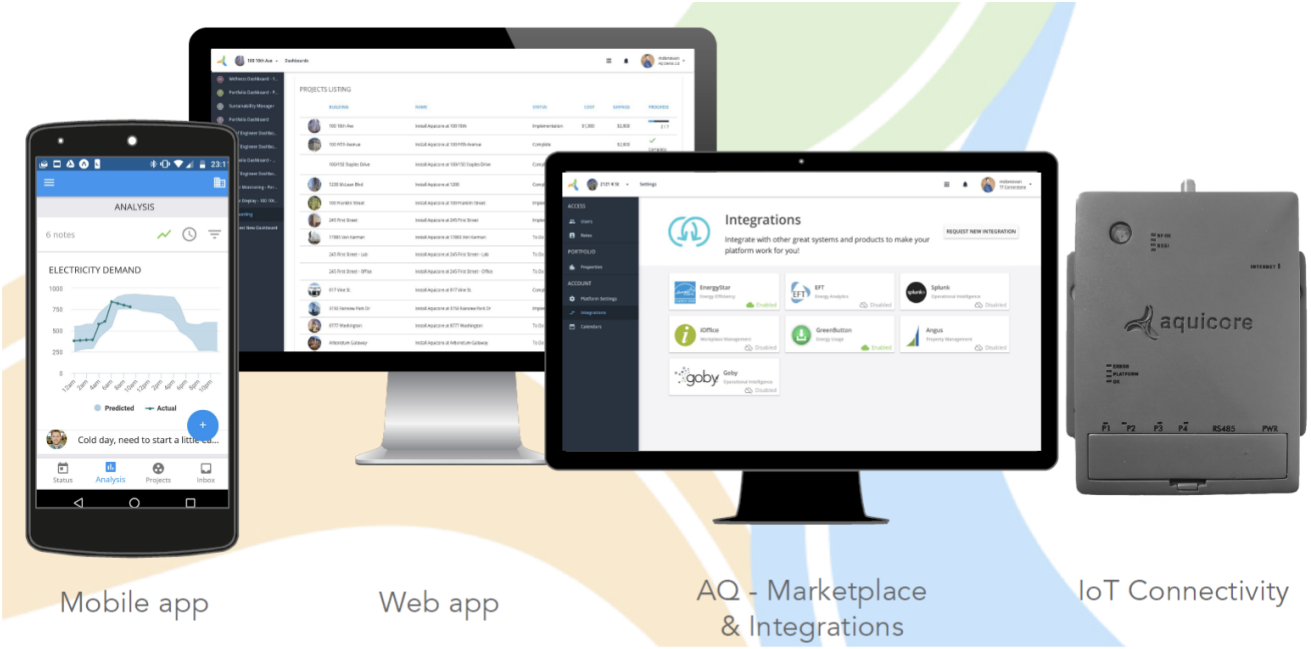
Aquicore's built-in collaborative tools and shared target tracking facilitate more robust lines of communication among executives, property managers and building engineers. With simple to install IoT sensors, Aquicore offers the only asset operations solution on the market that can be implemented in a matter of days — not months.

The Aquicore platform is deployed at more than 700 office buildings nationwide and over 150 MM square feet, including the John Hancock Tower, Salesforce Tower, National Press Building, Promenade and NASCAR Plaza. Aquicore's clients include Cushman & Wakefield, JBG Smith, Lincoln Property Company, MRP Realty, Under Armour, Facebook, The World Bank and Salesforce.

Case overview

Aquicore needed to enable centralized data collection and analysis in their cloud platform to monitor building systems for Commercial Real Estate (CRE) industry professionals in real time. With an industry that's behind the curve in data technology adoption and that faces the problem of siloed datasets across its various personas, it was critical for Aquicore to use the latest IoT data ingestion, storage and aggregation technology to automate billing, drive insights, optimize performance, and inform decision-making for property managers, building engineers, and accounting — all in a single platform.

Since IoT data is time series data, Aquicore chose InfluxDB Cloud as its managed time series database to collect and store metrics and event data from utility meters, submeters, building equipment, and environmental conditions of the buildings to help their SaaS solution deliver real-time and actionable insights. Using InfluxDB Cloud as its real-time IoT monitoring & analytics solution, Aquicore is revolutionizing "Asset Operations" for commercial real estate owners and operators and has become the backbone to IoT-data-enabled real estate decisions, paving the way for an autonomous building future.



Aquicore Technology Powered By InfluxDB Cloud Enables Data Centralization And Real-Time Monitoring To Improve Building Operation And Tenant Satisfaction

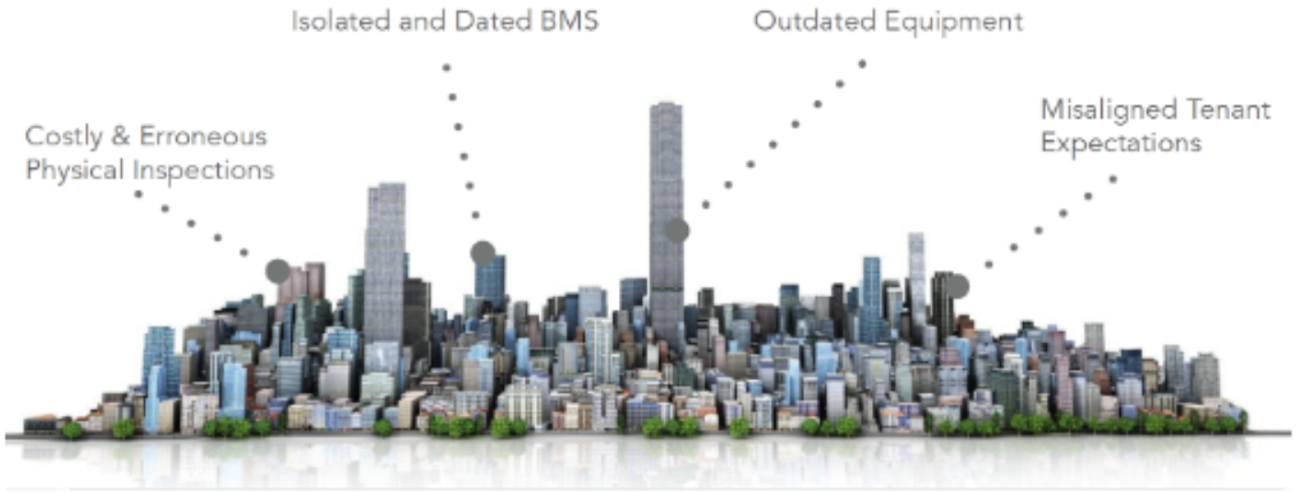
“The worst part about this setup in commercial real estate is that they’re very siloed data sets. The property manager has one reason to collect energy data while the building engineer has another reason, and they’re completely siloed. There’s no central platform that can give access to all this data.

Mike Donovan, VP of Products

The business problem

People who manage and own office space face numerous challenges caused by outdated technology and inaccessible data, and these are the business challenges Aquicore sought to solve through its platform as their customer base grew:

1. **Costly and erroneous physical inspections**: Most people are only able to receive one point of data about their energy usage or water usage (their utility bill), which is never sufficient to make smart decisions. To gather more data, building engineers end up walking around the building and taking readings that are erroneous and time-consuming.
2. **Isolated and dated Building Management Systems (BMS)**: The building management systems in place comprise fairly robust, control systems and sensors throughout the building, but are very dated and old-school. They typically consist of an on-premise server with no historical capabilities (and therefore no budget-setting and forecast possibilities).
3. **Outdated equipment**: Office buildings served by Aquicore are often decades old, with outdated equipment, and need more data to understand how their equipment is being used.
4. **Misaligned tenant expectations**: Unmet tenant expectations impact the bottom line of real estate because leasing out spaces is what generates revenue. While the real estate industry is several years behind the curve in adopting the latest building management and monitoring technology, office tenant expectations have evolved. They expect on-demand real-time visibility into energy usage, utility spend and billing across web and mobile.



CRE Industry Professionals Lack The Technology
To Centrally Collect And Analyze Data About Their Physical Assets

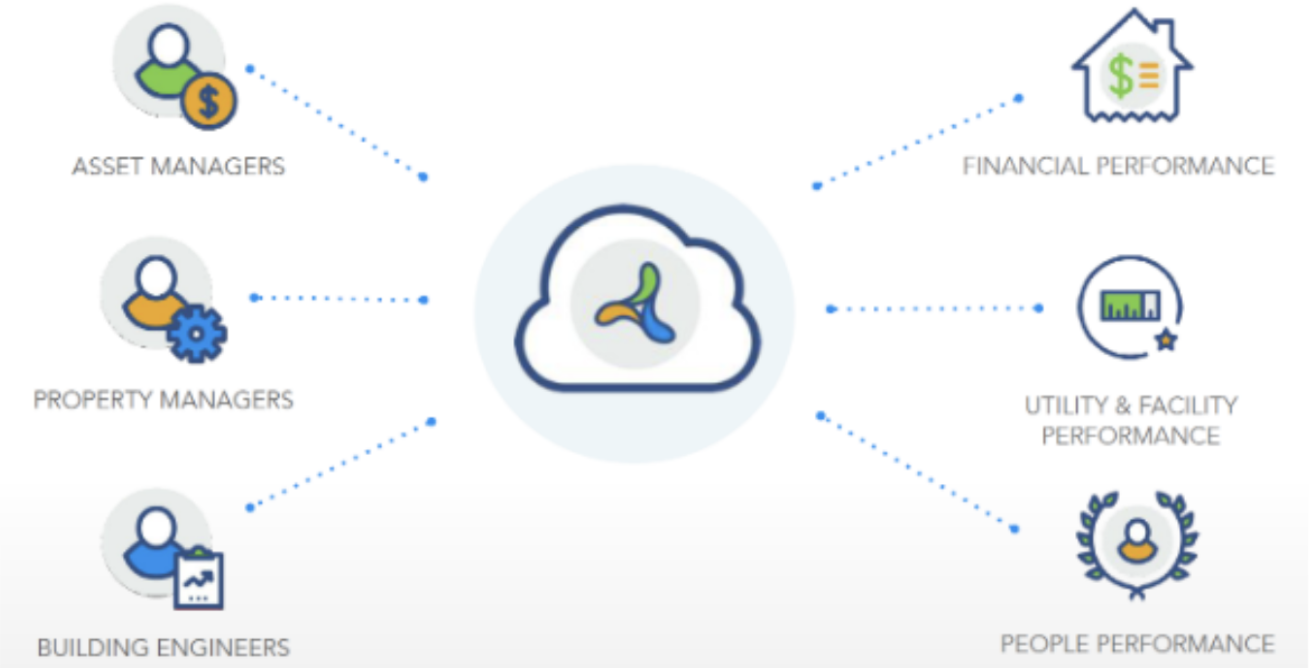
Lack of technology and data access create siloed datasets impacting the three different personas of commercial real estate:

1. **Property managers** (those who handle office tenants and lease office space)
2. **Building engineers** (on-site engineers who respond to complaints, handle equipment and conduct building systems' inspection for safety and energy usage),
3. **Accounting** (who handle lease accounts and billing).

Siloed Teams, Excel Sheets, Outdated Information,
And Decisions Made By Instinct Over Data

Aquicore therefor set out to build a platform that solves the problem of siloed datasets (by centralizing the operations of real estate for these three personas) and provides solutions for their distinct areas of work:

1. **Financial performance**: utility budgeting, tenant billing and understanding how utility spend impacts a building's financial performance
2. **Utility and facility performance**: understanding energy consumption in the building (such as water and gas) and understanding equipment usage
3. **People performance**: capturing and exposing real-time data and making it usable for CRE professionals that need it to do their job (such as the property manager running the tenant billing cycle every month or running a yearly budgeting process)



Aquicore Centralizes Commercial Real Estate Operations

The technical problem

Aquicore had originally built their platform using PostgreSQL (a relational database not designed to handle time series workloads) as a time series database, but they realized that it could not support their need for high data ingestion and for data aggregation at various levels. The data came in every minute, but they also needed to show 15-minute data and hourly data to users. They would have to sum up all the data points or take an average depending on the metric being shown or a daily value.

Since PostgreSQL had no built-in capability for data aggregation at different levels, they built a Scala application that would run and perform these aggregations. PostgreSQL was unable to handle this load, and the Scala application was too tough to maintain and wasn't the right solution to the problem of data aggregation.

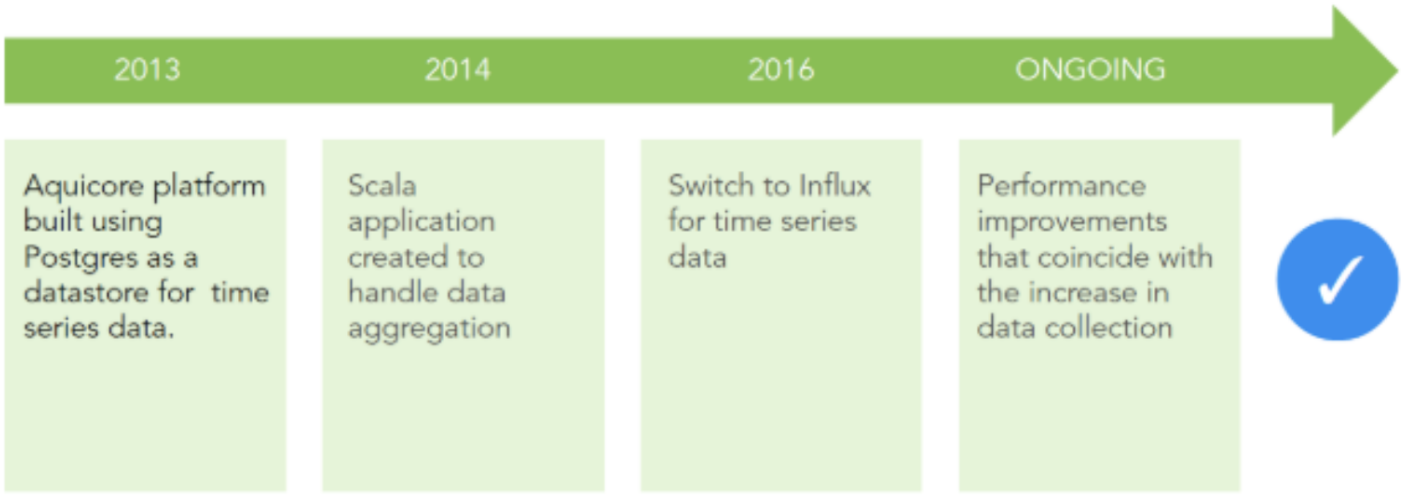
In 2016, recognizing the need to switch to a purpose-built time series database, they launched their migration from PostgreSQL to InfluxDB (a purpose-built time series database).

The solution

> *That's what we're always looking for: How can I get this information all in one place? Overall, I think we just really loved using Influx. It's been great. I think it's scaled with us and allowed us to reach all our goals as a company.*

Why InfluxDB Cloud?

Aquicore chose InfluxDB Cloud because they were familiar with it and because it was a managed solution that could handle IoT workloads and had built-in data aggregation capabilities through continuous queries. Their VP of Products had been tracking InfluxDB for a long time and had prior experience using it. Because they knew they were selecting a technology for the long haul, they sought a database that has a track record and a good community behind it. They were impressed by the InfluxDB community and consistent updates to InfluxDB Cloud. They use InfluxDB Cloud to run their production environment. In their QA and lower environments, they run their own managed version of InfluxDB on EC2 instance for the purpose of their own learning.



Timeline Of InfluxDB At Aquicore

The migration from PostgreSQL to InfluxDB took three months during which Aquicore ensured it was battle-tested before fully switching over. As more devices are added to their platform and as they collect more data for their growing customer base, they're constantly finding and resolving different bottlenecks in their platform.

Key features of the Aquicore platform fall into four categories:

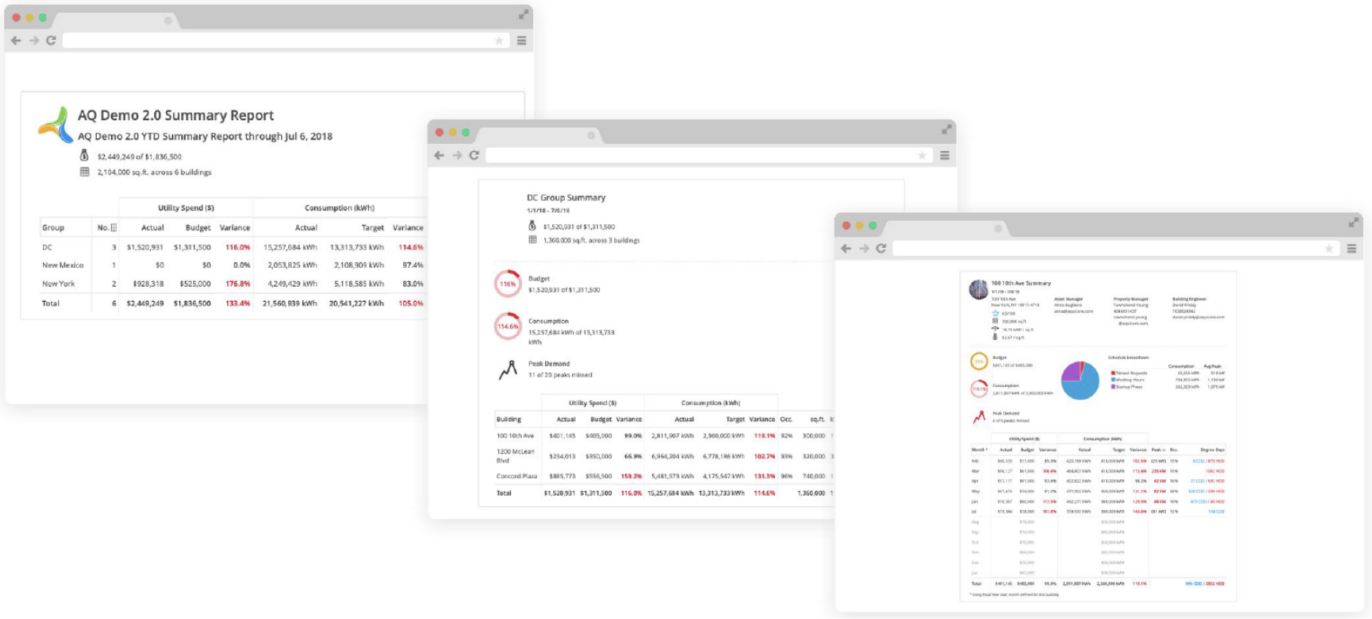
1. **Performance optimization** - Aquicore uses real-time equipment data and energy data to achieve hidden operational savings and grow portfolio value through live monitoring and prescriptive recommendations. Real-time monitoring shows what is happening in a building and enables adjustments (such as altering when the building comes online in the morning, verifying that the equipment is running as expected, or ensuring that the building doesn't operate on a weekend).
2. **Utility budgets** - The platform centralizes utility bills to gain intelligent budget recommendations, manage variance reporting, and track utility spend in real time.
3. **Tenant billing** - Aquicore automates, using real-time data, meter readings as well as tariff calculations and invoicing to maximize recoverables from tenants.
4. **Building connectivity** - Meter and sensor data are inventoried, streamed and centralized across a portfolio, in a single platform, thereby making critical information accessible to everyone. Building connectivity is why InfluxDB is critical to Aquicore's platform. While Aquicore is an asset operations platform, it is also an IoT company. It installs edge devices to web-enable buildings and meters and then sends that minute-by-minute data to Aquicore's cloud. Data is then ingested into InfluxDB. This applies for a building's energy sensors as well as for humidity, temperature and other sensors.

Aquicore's platform encompasses a mobile app, web app, an interface for real estate industry partners, and IoT connectivity devices:

- The platform's **mobile app** provides access to real-time data on the go, such as energy consumption and predictive analytics to forecast future energy consumption.
- The **web app** has all the dashboarding, visual insight, broad data and workflow tools housed within it.
- Aquicore also has a fairly robust **marketplace and integration platform** where they partner with other companies in the commercial real estate landscape. They consider their platform an open one that makes their data accessible and seeks to integrate with other platforms — a fairly new concept in the commercial real estate industry.
- For **IoT connectivity**, they build and deploy their own networking hardware. They connect their Aquicore Hub (AQ-Hub) device to one of the utility meters which then sends data to their cloud, turning "dumb meters" into smart meters.

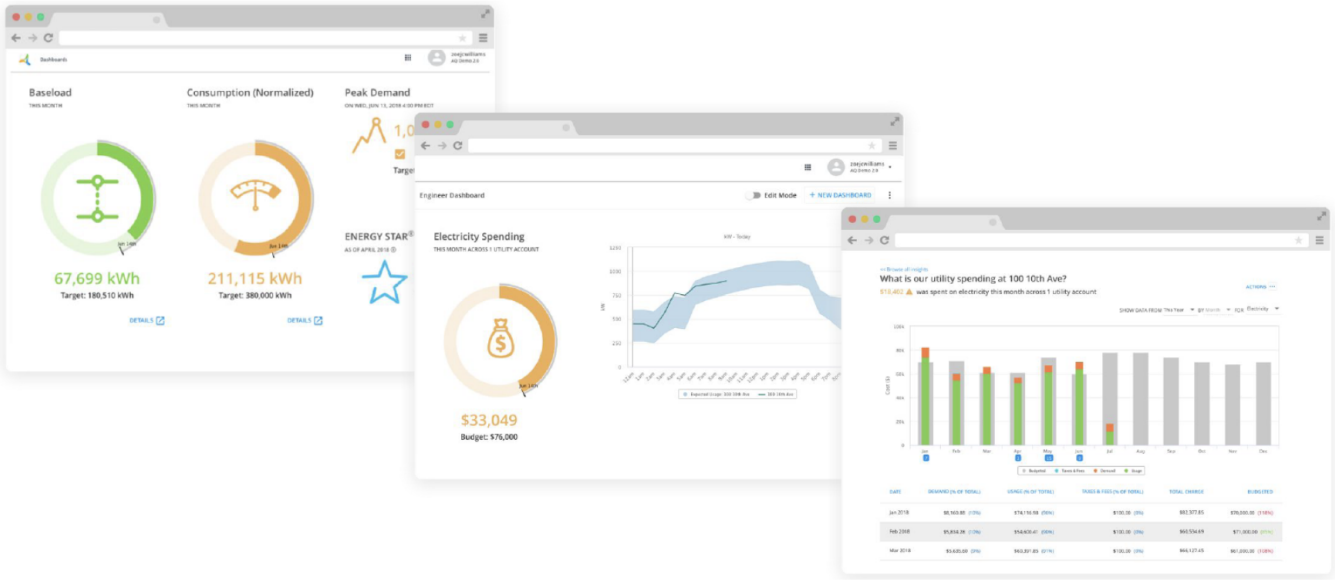
The platform ingests and runs calculations on real-time data through Aquicore's InfluxDB integration, to show energy usage or actual dollars being spent and thereby provide portfolio reporting.

Product Screenshots: Portfolio Report



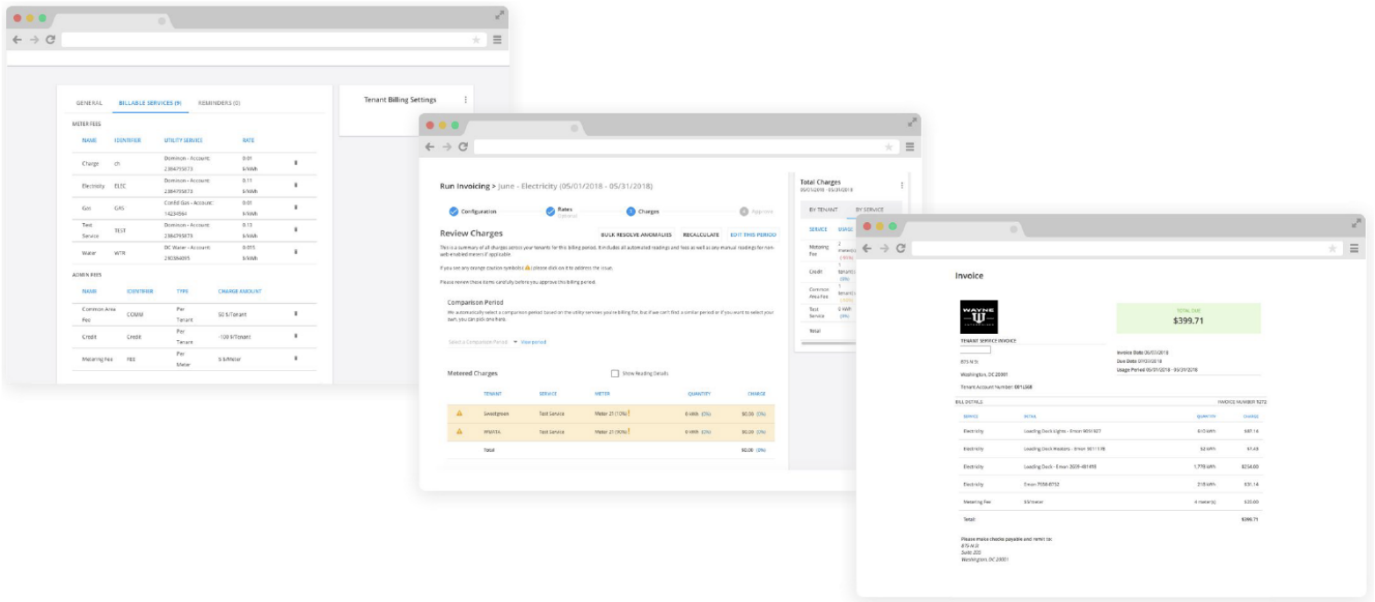
Most people don't know how to make a behavior change based on energy usage data. Setting budgets and performance targets makes data easily usable for commercial real estate professionals, enabling companies to reduce energy usage and utility spend while providing real-time insight into billing and consumption.

Product Screenshots: Budgets And Targets



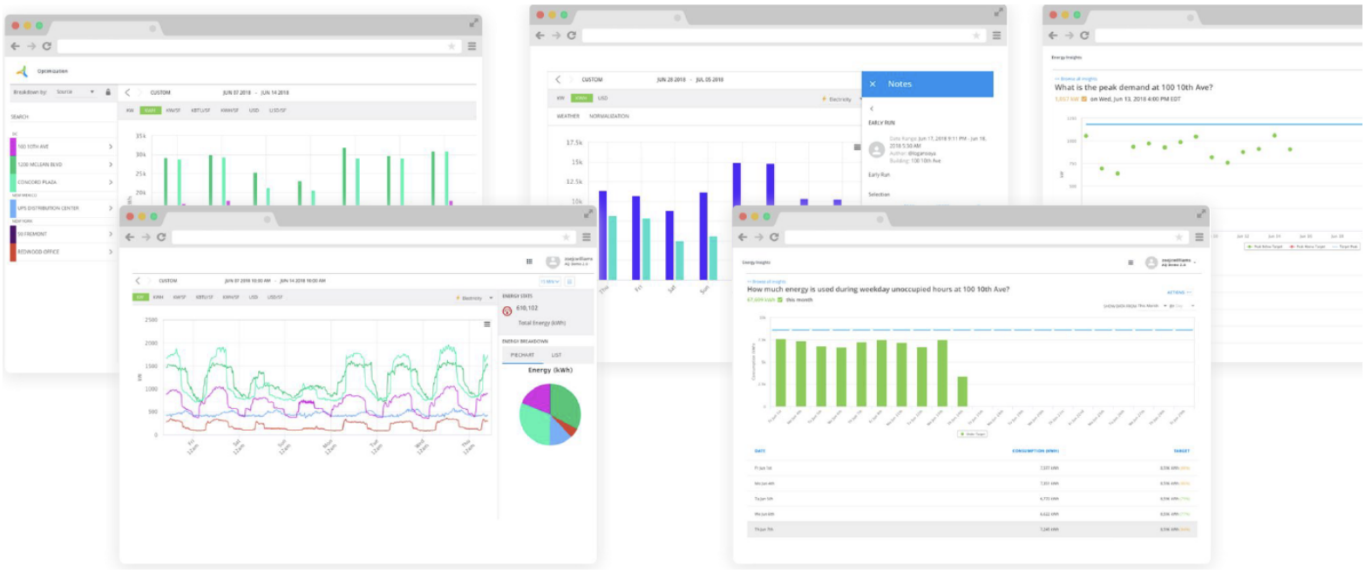
Tenant billing is another aspect where real-time data is revolutionizing the industry. Aquicore's platform eliminates what was a completely manual process — just by clicking a few buttons and accessing real-time data, you can now bill your tenants for energy usage.

Product Screenshots: Tenant Billing



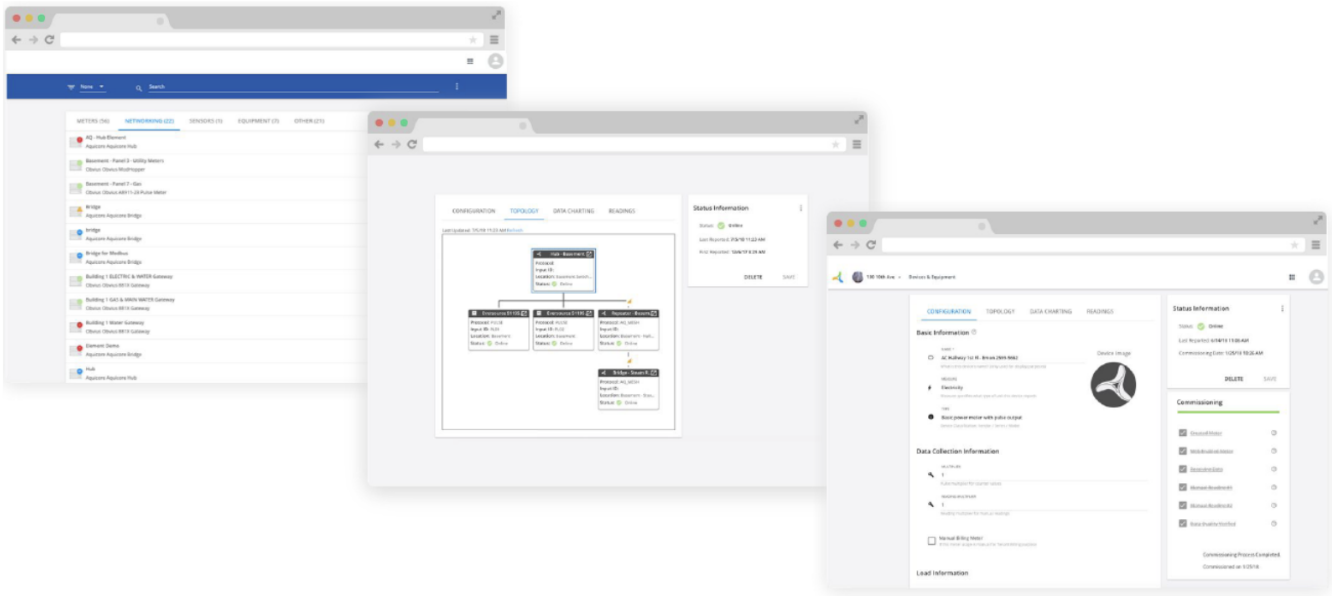
The platform also provides energy use visualization, showing real-time data coming into the platform throughout an entire building broken down in multiple ways — for the whole building, an individual piece of equipment or an individual piece of meter. Through an API, data can be exported as well.

Product Screenshots: Energy & Optimization

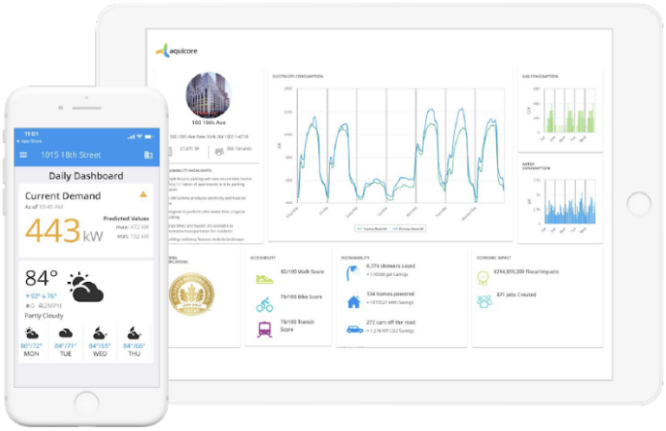


The platform has capability for Aquicore to manage their fleet of devices and hubs. They have a devices and equipment module which not only lets them manage their devices and site-installed networking equipment, but also inventory the entire building with the mobile app. Knowing how many meters and pieces of equipment you have is made easy by the Aquicore platform.

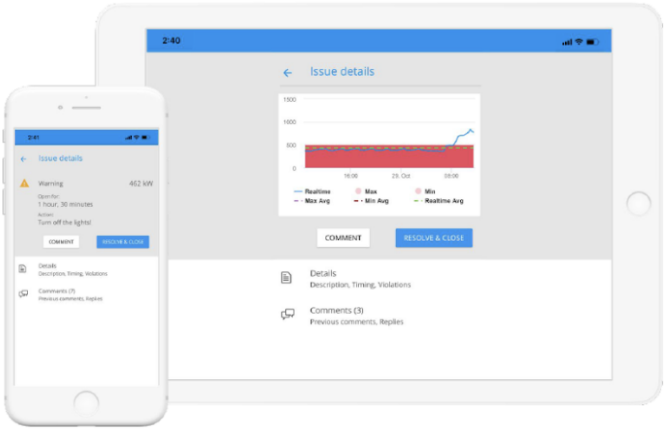
Product Screenshots: Devices & Equipment



Product Screenshots: Mobile



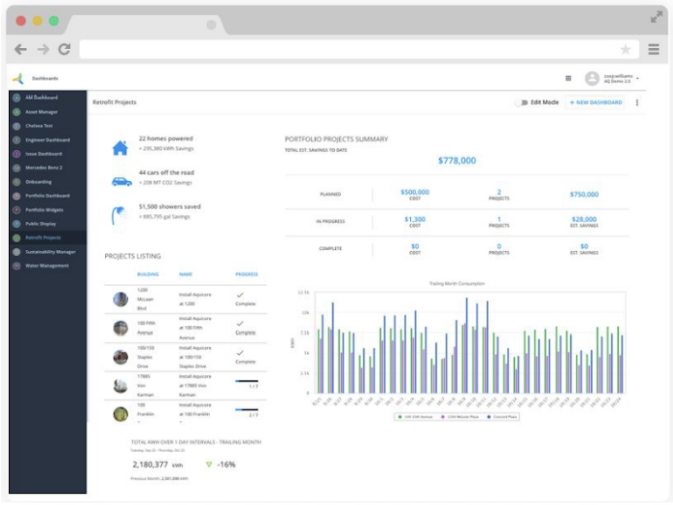
Mobile home screen



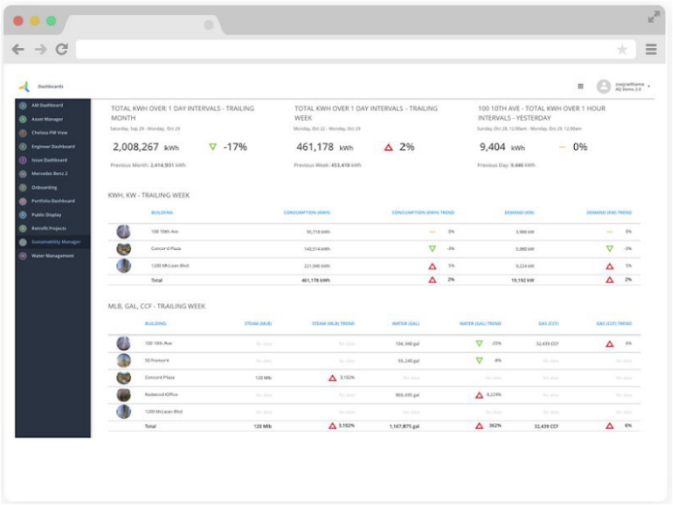
Issue management

As every real-time tool needs robust dashboarding capabilities, Aquicore platform dashboards are very configurable and flexible.

Product Screenshots: Dashboards



Dashboard w/ project listing



Dashboard w/trends

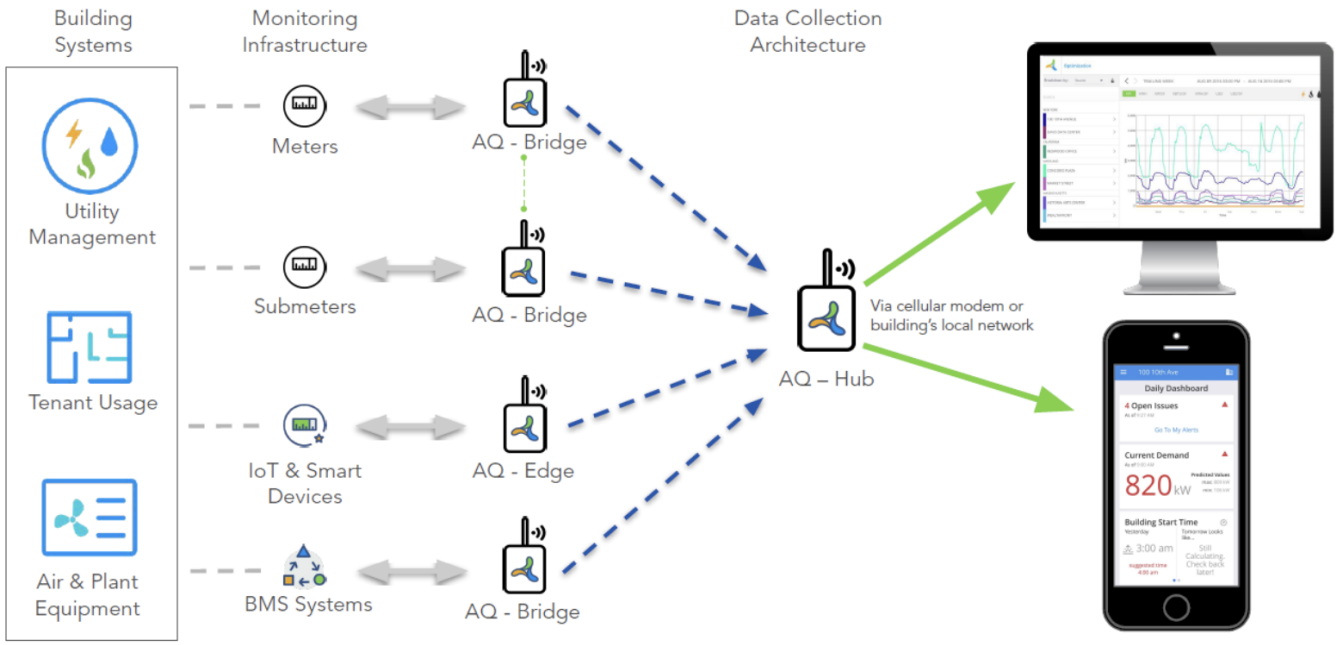
For Aquicore platform to scale with their growth and ingest higher data ingestion rates, it all came down to the real-time data that they're able to collect about these buildings. All that real-time data flows through InfluxDB. To meet the demands of their unique business use case, they've had to:

- **Build in support for various types of devices with different protocols**: They've iterated on the types of IoT devices to install in a building. As a young company, they began by using off-the-shelf products and hadn't yet built their own hardware. Being installed in hundreds of buildings, they'd run across different use cases and had to keep their platform data-source agnostic to remain focused on ingesting the data however they can.
- **Install and read from meters and sensors that exist in very hard-to-reach places within a building**: A six-foot slab of concrete will block any Wi-Fi connection or any wireless network connection between the meter and where you might have internet access or cellular access. Their sensor runs off a normal 12-volt plug. In most cases, they need to plug it into a nearby outlet, but the sensor can also be powered by the meter itself. So they sometimes connect their AQ-Bridge to the actual sub-meter and then pull off a little bit of electricity just to power it.
- **Provide real-time minute-by-minute data on-demand**: Since their platform is, in one respect, a real-time energy monitoring platform, customers use it on a daily, hourly, and sometimes few-times-per-hour basis to access their real-time data. One of the platform's differentiators is user access to very granular, minute-by-minute data, enabling tracking individual pieces of equipment, operational runtime, and the building's sequences. This creates a need for data quality and on-demand access.

Technical architecture

> *Once we moved to InfluxDB Cloud and got really good at it, it has allowed us to scale the amount of data and the varying types of data that we are able to collect much faster.*

Hardware Solution Architecture



The above diagram shows the data collection architecture inside of a building for Aquicore platform.

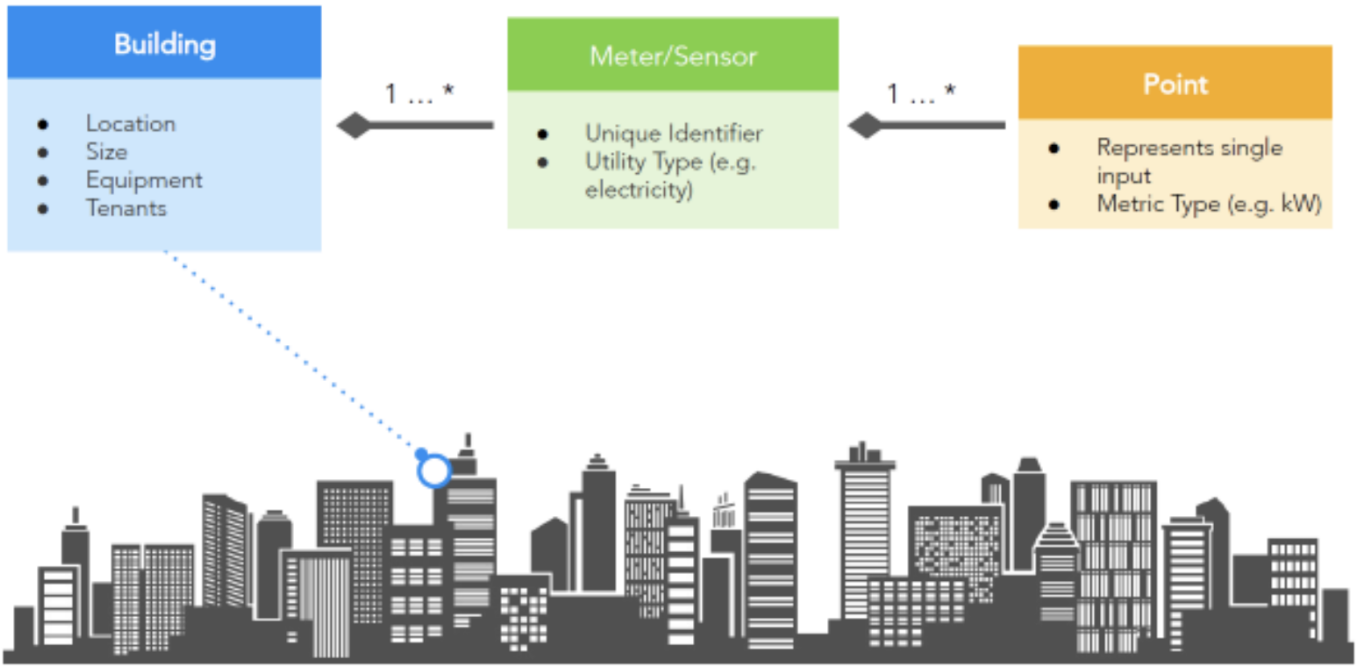
- The hardware solution is a critical component. An office building is equipped with utility meters and sub-meters (collecting information about a tenant's energy usage, not the whole building's), with such sub-meters scattered throughout the building.
- New developments are occurring in the space management area involving use of sensors to measure humidity, internal temperature, noise, and indoor air quality (IAQ). Tenants want a very clean and safe space to work in. It's time series data that they want to access. Aquicore devices can connect to that.
- A BMS system is still very prevalent, and where it's applicable and where they can connect to it easily and fast, then they can also access a host of information from it.
- The AQ-Bridge devices build a meshed network. It's not Wi-Fi or Bluetooth but a 900 megahertz proprietary network that can reach through the six-foot concrete slabs. And they all communicate eventually to an AQ-Hub, which is the cellular connected device that sends data to Aquicore's cloud.

From a high-level perspective, this is how the data is structured:

- The platform stores metadata about the building, meters, and individual points. So for every building they get installed in, there is a record of a building and its location, its size, the type of equipment installed in it, the tenants that occupy it and any information they can collect about the building.
- They configure one to many meters or sensors within the building. Each meter gets a unique identifier. And it represents some type of utility. So there might be four electricity meters that they will configure in the platform, each with its own unique identifier, as well as gas, water and sometimes an IAQ sensor. All of that is configured and associated with the building.
- Each meter or sensor, then, gets one to many points. Points represent a single input from this meter.

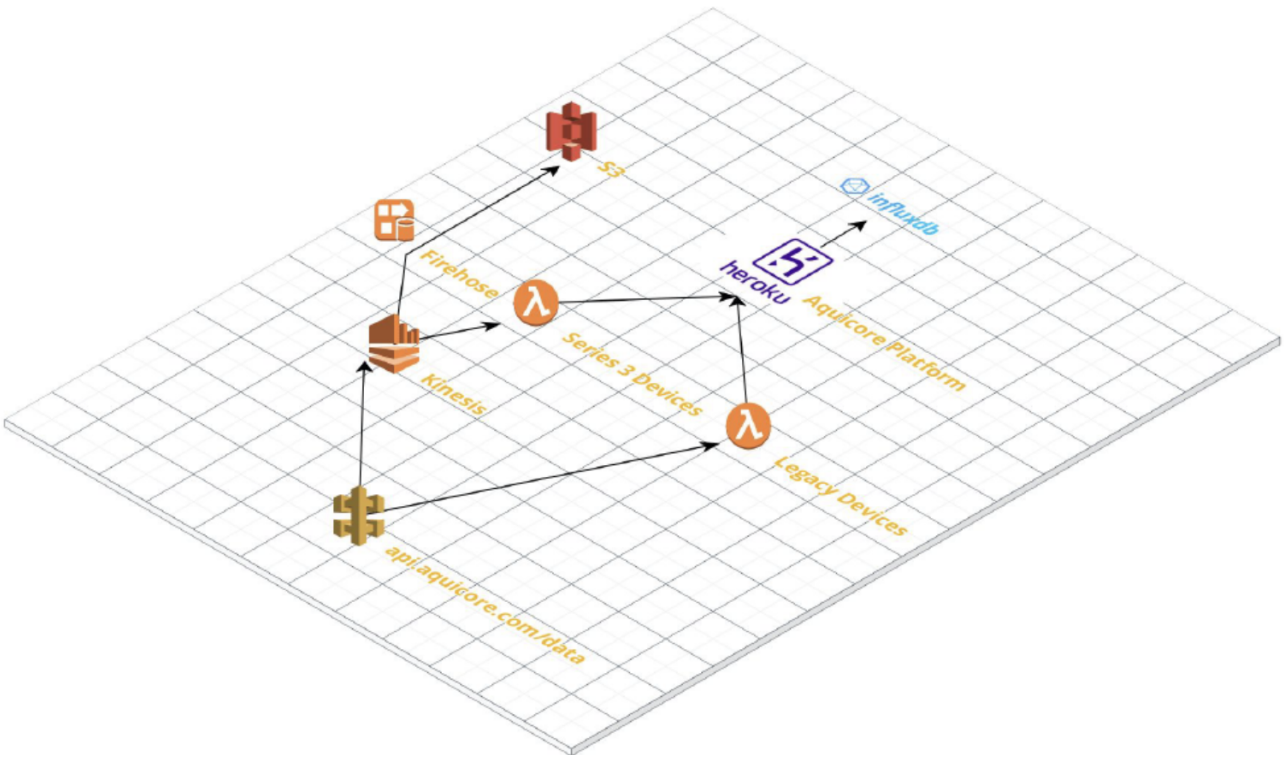
There's a lot of variability in the metering and sensor world. One specific meter might be able to send this data about multiple points of information (amperage, voltage, kW usage, the rate that water flows through the meter, the actual amount of gallons used). There's a host of metrics that can be sent to the platform from these points. Each of those different types of metrics is represented as a single point.

Data Structure Overview



With that data model in mind, this is how the data flows into the system:

Data Structure Overview



- Aquicore's hub sends data to their AWS cloud, which then routes it correctly.
- They have a big data processing, data ingestion stream that comes in through the API gateway.
- From the API gateway, there are two routes.
- One route sends the data to Kinesis (a fully-managed streaming processing service on AWS) as their queuing mechanism. Aquicore have a few consumers off that Kinesis stream. One of them is Firehose just for long-term storage in their data lake.
- A Lambda reads off of the Kinesis and does the appropriate forwarding to the Aquicore platform. The platform does all the business logic around reading the point. Each packet of data represents one of those points (like a kW value or a water-gallons-per- minute value) and associates it with the right metadata in Aquicore's database, which then stores it in InfluxDB.
- Another route from the API gateway leads to Legacy Devices (anything that's not Aquicore's own hardware).
- The Series 3 Devices are the AQ-Hub box.

Once the data is ingested by InfluxDB, it is stored based on the point ID. There are multiple tables or measurements set up for each utility type: electricity, gas and water measurement. For example, for electricity, there is the kW value, voltage value and power factor value — they create tags for each of the data values they have. They also include information about the time zone.

InfluxDB Cloud Data Structure

Electricity

point_id	timezone	historical	kW	voltage	power factor
123456	America/New York	false	8.3	270.4	0.8
123456	America/New York	false	6	250.3	0.7

Gas

point_id	timezone	historical	ccf
123456	America/New York	true	10
123456	America/New York	true	12

Water

point_id	timezone	historical	gpm
123456	America/New York	false	2
123456	America/New York	false	3

Here is the overall structure for how real-time data flows and gets sorted into InfluxDB:

- Point_id, time zone, and historical are all tags.
- The time series data streams in at minute intervals, which meets Aquicore customers' need. Some of their customers don't even need minute data but rather 15-minute data because that relates to how they are billed for energy (how much energy is costing them).
- Predictive analytics are stored in InfluxDB. In their Java application, they query InfluxDB for all the data, query all the building context information about weather and occupancy, and then store both queries' results in a separate measurement in InfluxDB just for predictive analytics.

Aquicore decouples building metadata from real-time data — because a meter that was once representing electricity or once representing a certain tenant will change over time, so they don't want their data in InfluxDB to be dependent on that. That allows them to adjust how they visualize the data.

Building metadata is stored in PostgreSQL database. There's a table for buildings, one for all the meters, and one for all the points. There are relationships built between them all. The "What's the usage on this meter" query renders all the points under that meter and they can build a query in InfluxDB that selects all the right points and then aggregates the data.

The only measure of data quality stored in InfluxDB is a tag to indicate whether or not the data is interpolated. There are times where a meter won't report for a few minutes but they don't want the user to have a bad experience so they'll interpolate the missing data, only up to so much time. Other data quality efforts occur external to InfluxDB and to this overall data processing. This is where Aquicore's data science ties in. Aquicore has a support process around performing data quality analysis and fixing issues.

As for the framework used to build Aquicore's dashboards:

- The main backend (integrating with InfluxDB and bringing the databases on API calls) is a Java application currently running in Heroku. As big proponents of serverless, however, they're slowly breaking apart that application into various microservices that run in Lambda.
- The front-end application on the web app is an angular application, and on the mobile app, it is a React Native application across iOS and Android.

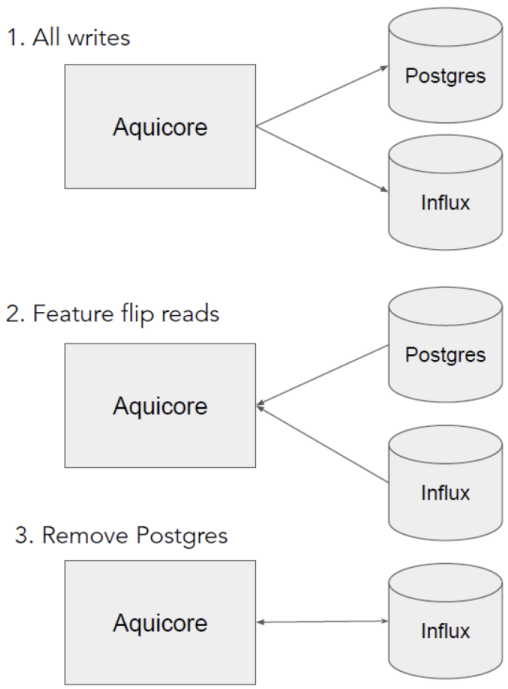
Technical challenges

Throughout the three years of using InfluxDB and relying on it for their production environment, Aquicore has faced and resolved several challenges:

1- PostgreSQL to InfluxDB migration

Problem:	PostgreSQL was unable to handle the load of Aquicore's growing customer base.
Solution:	Migrate to InfluxDB using feature flippers and replicating data between both databases for a period of time.

One of the first steps — once they had Influx up and running (using InfluxDB Cloud for production) — was to duplicate the data across, sending it to InfluxDB and PostgreSQL. They did that for a few months to ensure that they could spot-check the data, and that applied to every single write. Their platform has a feature-flipping mechanism, for certain customers, that allowed them to serve the platform using InfluxData. Once they were far enough along in the implementation and felt comfortable flipping this on for a few customers — that allowed them to incrementally add more risks. Eventually, they turned this feature on for all customers. After a few weeks of waiting and triple-checking everything, they were able to completely remove the PostgreSQL instance as well as the Scala application running it and just replace them with InfluxDB for their time series data.



2- Data aggregation challenge

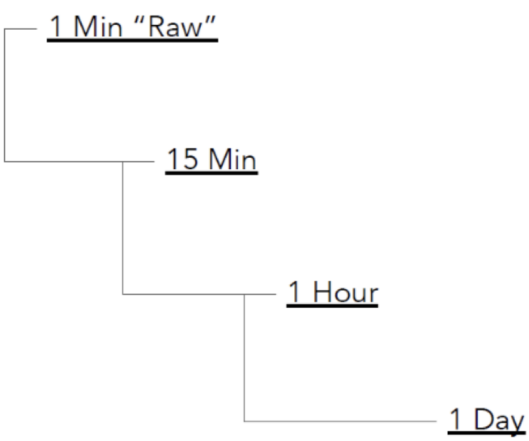
Problem:	Devices send Aquicore data every minute, but users want to see data aggregated at various intervals including 15 minute, hour, and day.
Solution:	Implement a set of CQs (Continuous Queries) for each interval

Originally, depending on the type of unit being calculated (such as average energy usage or total water usage), different math was run on each in the Scala application. This was too much of a load for Aquicore's system. One main reason why they wanted to move to InfluxDB was that continuous queries were built-in, which meant they could have something internal to the database that would perform these calculations to the real-time data ingested.

Below is a sample continuous query used to create separate measurements per aggregation. Their query logic, depending on what the user wants to see, will query the right measurement to look at.

```
CREATE CONTINUOUS QUERY cq_15m_water ON aq RESAMPLE
FOR 30m BEGIN SELECT mean(flow_rate) AS flow_rate,
sum(volume) AS volume INTO aq."15m".water FROM aq.raw.water
GROUP BY time(15m), * END
```

They implemented this two and a half years ago — yet the daily aggregation still had an issue with time zones.

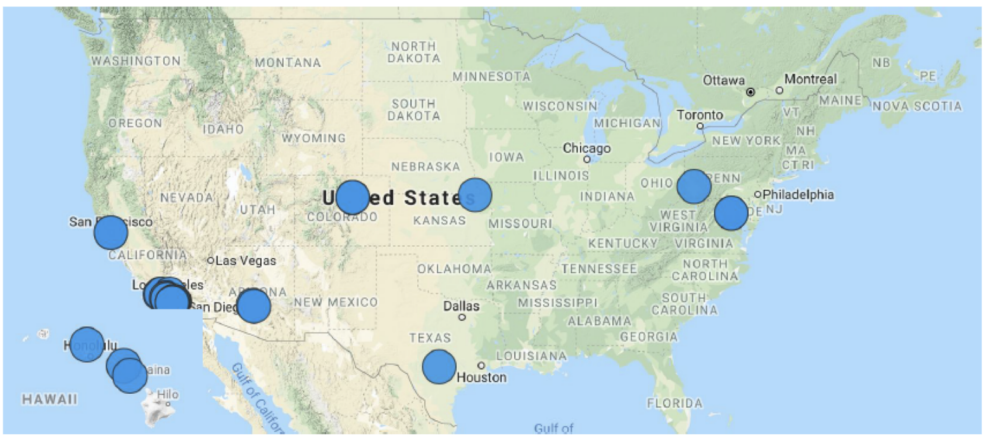


3- Time zones

Problem:	At the time of implementation, InfluxDB was unaware of time zones which made it impossible to use CQ's to create a daily value.
Solution:	Create a timezone-aware job in Aquicore's platform that performed the data aggregation (that would run for all the buildings in the right time zone and then execute the InfluxDB query to do what a continuous query would do instead).

Aquicore's platform is used in hundreds of buildings, across the US, including Hawaii, as well as in some buildings internationally, so Aquicore wanted to provide a correct daily aggregate value for all their units. At the time, InfluxDB didn't have enough support for time zones to allow them to use continuous queries to create a daily value.

Today, InfluxDB support for time zones is much better, but Aquicore haven't yet refactored how that product improvement could work for them. Since their solution has been successful, time zone support hasn't been that big of an issue.

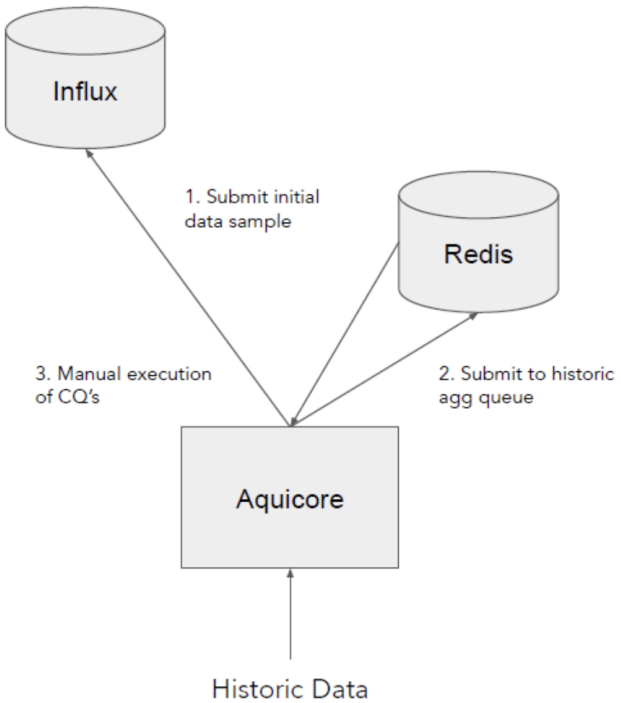


4- Historical data

Problem:	Aquicore's devices and users submit historical data frequently which CQ's won't process.
Solution:	Aquicore built a historic data submission and aggregation queue that executes once a batch of historic data is submitted.

Users come to the platform to view real-time data that Aquicore's on-site devices are sending. But since things happen in the real world with on-site devices, there might be gaps of data. There might be periods where the customer wants to upload historical data to compare their new data with a baseline. Given the major need for such historical data uploading, Aquicore built a historic data background process that will work as customers upload the historical data or when Aquicore's devices send them historical data from going offline after a period of time.

When the historical data comes in, this initial data sample is submitted to InfluxDB as well as being sent to Redis. Then the platform is also listening to know if there's any batch of historic data that it needs to aggregate. If it finds that, it will manually execute the continuous query, again, for that window of time that the historic data was submitted for. It's definitely something that they've always had to do and something they've iterated up to scale because their customers never want gaps in their data.



5- Write performance

Problem:	Under heavy load. they experienced slow write performance causing memory and CPU issues with their application.
Solution:	Initial implementation used synchronous method in the Java client and switching to recommended asynchronous approach improved performance.

The initial implementation uses the Java-Influx client. Aquicore didn't use the recommended approach, but as soon as they switched to using the recommended approach of asynchronous writing and batch processing, then all their write problems to InfluxDB Cloud disappeared. The cause for error was that they were trying to manage synchronous writes. For them, this serves as a real-world example of where the robustness of the platform is based on the people and decisions that every developer makes on a daily basis.

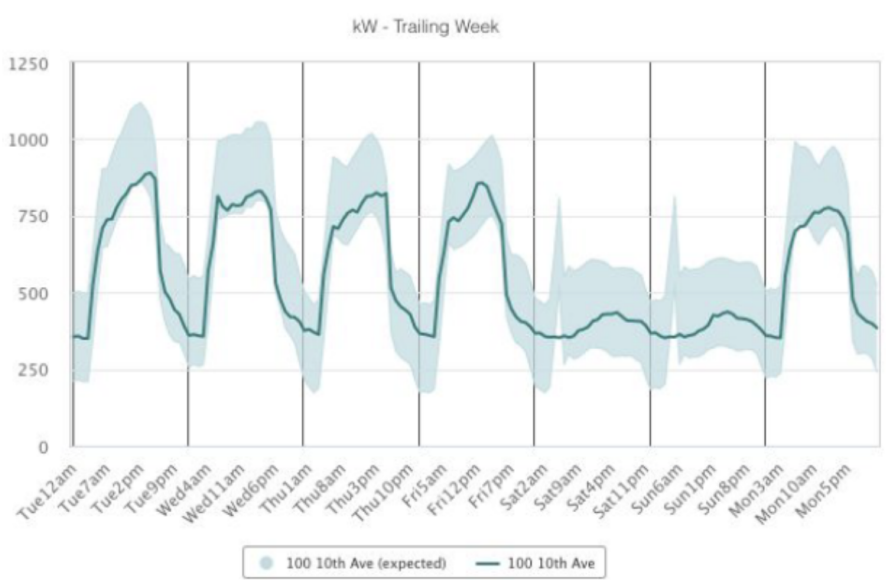
This is a recommended approach to write data points into InfluxDB. The influxdb-java client stores writes into an internal buffer and flushes them asynchronously to InfluxDB at a fixed flush interval to achieve good performance on both client and server side. This requires influxdb-java 2.7 or newer.

```
InfluxDB influxDB = InfluxDBFactory.connect("http://172.17.0.2:8086", "root", "root");
String dbName = "aTimeSeries";
influxDB.query(new Query("CREATE DATABASE " + dbName));
influxDB.setDatabase(dbName);
String rpName = "aRetentionPolicy";
influxDB.query(new Query("CREATE RETENTION POLICY " + rpName + " ON " + dbName + " DURATION 30h REPLICATIO
influxDB.setRetentionPolicy(rpName);

influxDB.enableBatch(BatchOptions.DEFAULTS);

influxDB.write(Point.measurement("cpu")
    .time(System.currentTimeMillis(), TimeUnit.MILLISECONDS)
    .addField("idle", 90L)
```

6- Predictive analytics

Problem:	Feature request to build predictive capabilities for time series data factoring in building characteristics like weather and occupancy
Solution:	They attempted to use Kapacitor but were unable to join multiple data sources together, and ended up implementing a Java application.

When the platform was first launched, Aquicore didn't have any predictive analytics. They received a feature request to add analytics since users want to improve their operations and know if they're doing well. The darker line in the below graph represents the real-time usage of energy in the building.



That predictive envelope is what they predict the building should use based on historical data as well as external factors to the building (such as weather and occupancy changes). Weather and building context information are stored in a different dataset, into the Java application, which can include these external data sources.

6- Predictive analytics

Problem:	Diagnosing production performance issues where the cause appears to be writing from or reading to InfluxDB.
Solution:	Originally, InfluxDB Cloud did not provide robust monitoring capabilities. Enhancements to Grafana have made this much better but would be better if it integrated with their APM tools.

As a platform that's existed for a few years, they've had various go-rounds of different application monitoring tools. They always try to utilize the best third-party services to stick to their own business needs and what they need to do for their customers. When they had first launched their platform, InfluxDB Cloud monitoring capabilities were still under development and have significantly improved since that time.

Results

> *The name of the game is all about how fast can we build high-quality software in production. I don't want to be worried about building my own time series database. I don't want to even be worried about managing my own time series database... By using InfluxDB Cloud, we've got that whole responsibility lifted off our shoulders.*

In terms of outcomes, InfluxDB Cloud has enhanced Aquicore's competitive advantage. It freed them to focus on their core business of on-demand real-time and predictive analytics across devices. By querying InfluxDB to populate the graphs on their mobile and web apps, they can show customers exactly what's going on.

To determine what other data sources to bring into the mix to generate predictions, Aquicore draws on the knowledge of its in-house building systems experts. They work with them to understand how equipment runs, what impacts the building and how seasonal variabilities and HVAC systems impact building operations. Keeping up to date with that information and knowing how to factor it in is critical for Aquicore.

The Aquicore platform's user base is benefiting from having real-time data at their fingertips, Aquicore platform delivers the following advantages for:

- **Building engineers**: Aquicore's live recommendations and real-time monitoring keep buildings and operations running smoothly.
- **Property managers**: Aquicore helps property managers stick to their operating budgets and keeps their team aligned around performance goals.
- **Asset managers and limited partners**: With configurable performance targets, budgets, and executive reporting, Aquicore gives asset managers and limited partners the complete view of their portfolio.

The platform's user engagement, which is tracked by Aquicore, currently exceeds 70% MAU (monthly active users). This high figure further indicates that the building engineers, property managers, and other users are extracting value and insights from this data. High usage has also resulted in a growing list of feature requests requiring Aquicore to consider collecting an even wider variety of data. They recently implemented a humidity sensor and a temperature sensor in a data center. In real estate, more management companies are renting out data center space and want to monitor their indoor temperature and humidity to ensure optimal operations. Aquicore continues to work with customers to solve emerging needs.

Aquicore's advice about building a SaaS is to be mindful to not reinvent the wheel and to always align your architecture with the business value you're trying to deliver because this is how you'll always make the right decisions along the way and not over-engineer your solution. By maintaining focus on its core business and leveraging InfluxDB Cloud as its platform's managed time series database, Aquicore is living up to its promise of "changing the way commercial real estate does business."

About InfluxData

InfluxData is the creator of InfluxDB, the leading time series platform. We empower developers and organizations, such as Cisco, IBM, Lego, Siemens, and Tesla, to build transformative IoT, analytics and monitoring applications. Our technology is purpose-built to handle the massive volumes of time-stamped data produced by sensors, applications and computer infrastructure. Easy to start and scale, InfluxDB gives developers time to focus on the features and functionalities that give their apps a competitive edge. InfluxData is headquartered in San Francisco, with a workforce distributed throughout the U.S. and across Europe. For more information, visit influxdata.com and follow us @InfluxDB.