C3.ai

Predictive Maintenance for a Network of Devices

Product Specification

Table of Contents

 |

1. Introduction

The C3.ai Device Predictive Maintenance (PM) application is designed to help predict failure of devices based on an analysis of large numbers of devices connected to the C3 AI Suite. This application will help operators replace devices before they lead to costly interventions, ensuring continuity of operations. Using the C3.ai Device PM application, operators will systematically reduce the cost associated with device failures.

C3.ai Device PM analyzes the lifecycle of a device based on historical data and recognizes when a device is at risk of failure. By timely assigning risk scores to individual devices, the C3.ai Device PM application allows operators to act before failure occurs.

The C3.ai Device PM application is part of the C3.ai Applications suite. This suite is designed to help building operators and companies optimize the usage of devices in their network. Additional applications include C3.ai Inventory Optimization and C3.ai Energy Management.

C3.ai Device PM supports the acquisition and aggregation of data from different sources. Integrated data elements include:

- Devices (information about device instances)
- PowerGridStatus (information about the power grid status for each building)
- PowerSource (information about power source instances and their locations in buildings)
- DeviceEvent (information about previously known device events like outages)
- DevicePowerSource (information about when devices were in power sources)
- DeviceMeasurement (historical measurement data for each device)
- DeviceWeather (historical weather data for weather stations)

Decision Support Analytics

C3.ai Device PM allows the operator to answer a range of key questions around the status of the devices in their network:

- How many devices are in operation / have failed / are at risk?
- Where are the devices in my network located and what is their status?
- What is the risk distribution of the devices and which devices might fail soon?

Benefits of C3.ai Device PM

C3.ai Device PM generates multiple benefits to operators:

- Avoid operation disruptions by proactively replacing devices that are about to fail
- Manage the inventory of devices better. By knowing when devices will fail, operators will be able to stock up on devices, without keeping too much inventory
- Reduce labor required to service the devices. Maintenance shifts can be planned since device failure can be predicted, reducing the need for ad-hoc interventions

2. Required Functionality & User Interface (UI)

The C3.ai Device PM application must be able to report on the current number of devices, along with their current location, status, and measurement data. Additionally, using this data, the application must continuously and automatically generate predictive risk scores about likelihood of device failure based on the most recent data. Lastly, the application must have an interface that allows an operator to visualize this data both in aggregate and for individual devices.

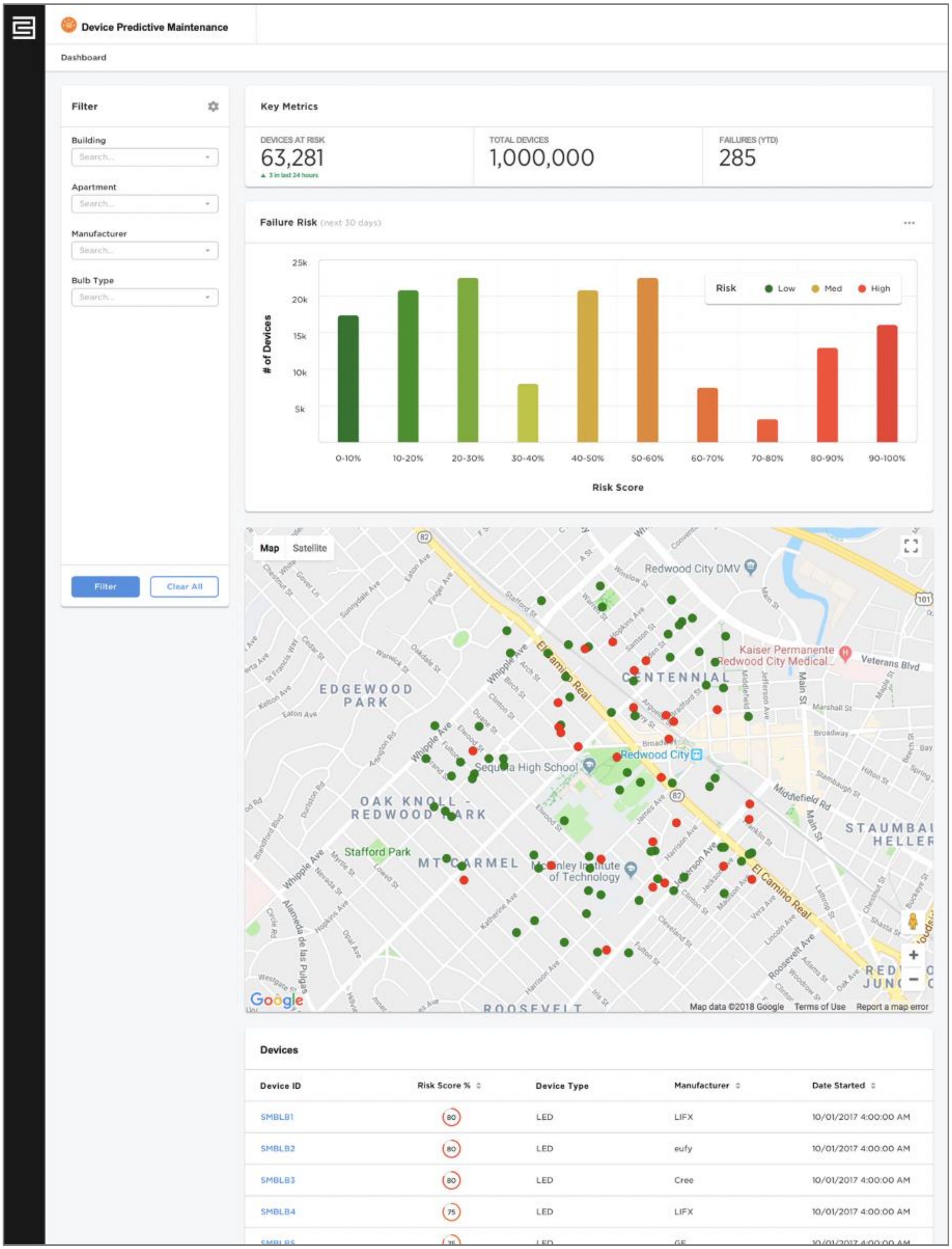
Device Dashboard

The Device Dashboard home page provides a rich, at-a-glance summary of a range of valuable information around the status of the device network (Figure 1). Furthermore, the operator can filter this dashboard based on the building, manufacturer or device type that they want to look at. By selecting a device in the details list at the bottom of the dashboard, the operator is directed to a detailed analysis page for that specific device.

Across this page, operators can view components with the following functionality:

1. Status of the device network (devices at risk, total devices and failures
2. Real-time risk distribution, showing risk of device failure in the 30 days
3. Location of devices in the network and their status
4. Detailed list of devices (showing various product details and the risk scores)
5. Dashboard filter based on building, manufacturer and device type

Figure 1. Dashboard View



Overview

Across the top of the page, operators can view components that show the:

- Total number of devices at risk
- Total number of devices
- YTD device failures

Below is a description of the components and their functionality

Component	**Functionality**
Devices at risk	Total number of devices currently operating above 50% risk score
Devices switched on	Total number of devices currently switched on
Device failures	YTD device failures (status is on but no output)

Risk Distribution

This element shows the risk distribution of the devices as a histogram, based on the risk scores assigned through our machine learning algorithms. This histogram displays the rolled up current risk score of all the devices, binned into 10 risk score categories.

Component	**Functionality**
Risk distribution	Aggregate risk score from individual devices binned in 10 different risk levels (so bins 0-10%, 10-20% and up to 100%)

Device Status Map

Operators can view the geospatial locations of each of the devices. A green dot on the map indicates a healthy device. A red dot indicates that the light device is at risk (i.e. the risk score is > 50%). Clicking on an item on the map takes operators to the detailed analysis page. Geospatial data are available on the Device Type (latlong of the device) as well as the Building Type (latlong of the building itself).

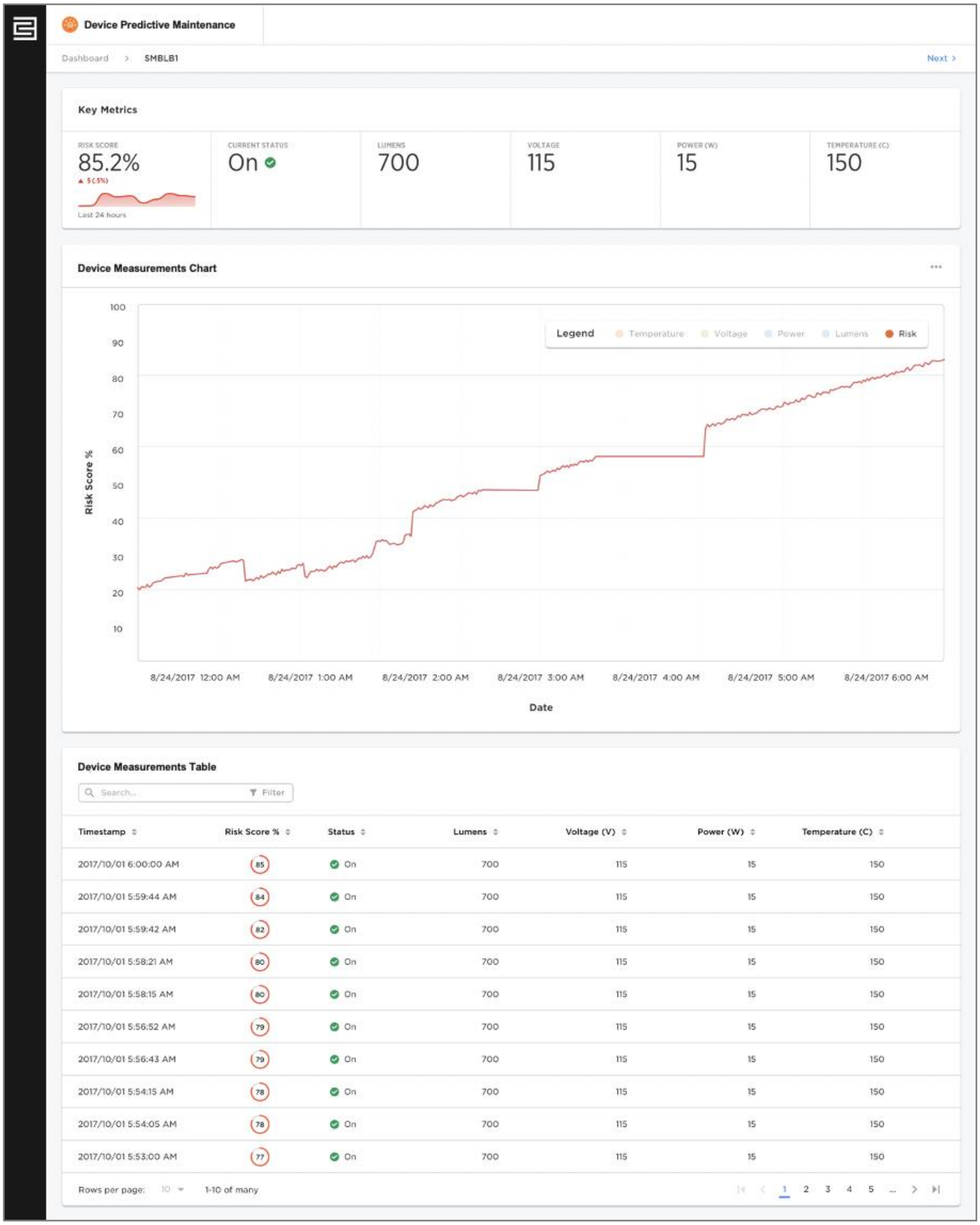
Component	Functionality
Device map	Device at risk indicator (green or red) and latitude and longitude information on all the devices

Device Details

Operators can see a list view of devices at highest risk, with an opportunity to intervene before these devices fail (being alerted to this beforehand). Details shown are device ID, device type, manufacturer, start date, age (days), risk score, and building. Clicking on this list takes operators to the detailed analysis page.

Component	Functionality
Device details	• List of devices above a 50% risk score with product details (ID, device type, manufacturer, start date), location (building) and risk score • Number of days the smart device has been active • Current risk score of individual devices

Figure 2. Device Analysis



Device Analysis

The Device Analysis screen as seen in Figure 2 enables operators to select criteria to analyze the measurements related to their devices over time. Operators are guided to this page by selecting a specific device on the dashboard page. The top of this page gives an overview of the status of a device. Below the device overview, operators can select a measurement variable they want to graph (temperature, voltage, power, output and risk score) and look up specific data in a list.

Device Summary

Across the top of the page, operators can view components that show the:

- Risk score of the selected device
- Status of the selected device
- Current values of output, voltage, power, temperature

Several time series are used to feed the components described above. The following table summarizes the time series involved:

Component	**Functionality**
Risk score	Current risk score of devices
Status	Status of the device
Output	Current output of device
Voltage	Current voltage of device
Power	Current power of device
Temperature	Current temperature of device

Device Chart

This element allows the operator to create a chart of a device's measurement values and risk score. In the filter the operator can select the device ID, the period they want to look at, the data frequency and the analytic they want to view (temperature, voltage, power, output and risk score):

Component	**Functionality**
Device chart	• Output of device at different times • Voltage of device at different times • Power of device at different times • Temperature of device at different times • Risk score of device at different times

Device Details

In this table, operators can see details on the selected device and view the measurement data at different time stamps. Below is an overview of the time series involved in generating this table:

Component	**Functionality**
Device details	• Output of device at different times • Voltage of device at different times • Power of device at different times • Temperature of device at different times • Risk score of device at different times • Status of device at different times

 |

3. Time Series and Metrics

C3.ai Device PM leverages several configurable metrics that analyze the normalized device data to produce timeseries data using the specified inputs and configurations. C3.ai metrics allow for massive data processing power with very little code using the C3.ai Expression Engine. For example, device data may arrive erratically every 10-58 minutes but can easily be normalized and standardized for reporting at any requested interval, such as every 15 minutes. Below is a list of metrics that are run in C3.ai Device PM, with details around the inputs, outputs, and underlying algorithmic logic. Detailed descriptions are in the Appendix.

Name	Description
AverageOutput_Device	Average output over time of the device
AveragePower_Device	Average power over time of the device
AverageTemperature_Device	Average voltage over time of the device
AverageVoltage_Device	Average voltage over time of the device
PowerGridStatus_Building	Status (1 or 0) indicating ON or OFF of the power grid over time
PowerGridStatus_Device	Status (1 or 0) indicating ON or OFF of the power grid over time
Status_Device	Status (1 or 0) indicating ON or OFF of the device over time
DurationOnInHours	The total amount of time a device is switched on up to the interval
HasEverFailed	Indicates (1 or 0) if the device has failed during evaluated interval
IsDefective	Indicator (1 or 0) if the device died
SwitchCount	The number of times a device is switched on or off beginning at the 'start' given
SwitchCountPreviousWeek	The total number of times a device is switched on or off up in the previous week of the interval
WillFailNextMonth	Indicator (1 or 0) if the device died within the next 30 days

4. Machine Learning

C3.ai Device PM creates risk scores for devices in an operator's network by using machine learning algorithms. Jupyter Notebooks, integrated directly into the C3 AI Suite, allow users to work with Python, C3.ai types and functions, and real-time data to produce sophisticated models that can operate on the existing data, create new data, and be re-integrated into the application to continue automatically updating and making predictions on the latest available data.

The application first normalizes device measurement data and then trains a classification model using logistic regression. This training model uses the following time series as regression input variables:

Input Variables	Description
SwitchCountWeek	Total number of times a device is switched on or off, in the previous week of the interval
DurationOnInHours	Total amount of time a device is switched on, up to the interval

These time series are then regressed against the dependent variable WillFailNextMonth, which is a categorical variable indicating (1 or 0) if the device died within the next 30 days. Finally, using this regression, a probability of failure in the next 30 days (the risk score) is calculated.

Using Jupyter Notebooks, operators can segregate sections of code into discrete cells that allow for data manipulation on the side before re-integrating the results into the main programmatic sequence. Using native C3 AI Suite functionality, they can explore and manipulate data already in the platform directly.

 |

Figure 3. Jupyter Notebook View

File Edit View Insert Cell Kernel Widgets Help | Trusted | Python [conda env:c3-deeplearning]

Markdown

Let's get a little bit of data from **SMBLB1** using `.get()` to render in a grid.

In [11]:

```
ts = emr.get("SMBLB1", {}).get('data')
ts
```

Out[11]:

Time	SwitchCountPreviousWeek	HasEverFailed	WillFailNextMonth	DurationOnInHours
2011-01-01 00:00:00	0.0	0.0	0.0	3.0
2011-01-02 00:00:00	4.0	0.0	0.0	13.0
2011-01-03 00:00:00	8.0	0.0	0.0	21.0
2011-01-04 00:00:00	16.0	0.0	0.0	33.0
2011-01-05 00:00:00	20.0	0.0	0.0	43.0
2011-01-06 00:00:00	24.0	0.0	0.0	49.0
2011-01-07 00:00:00	30.0	0.0	0.0	52.0
2011-01-08 00:00:00	34.0	0.0	0.0	56.0
2011-01-09 00:00:00	34.0	0.0	0.0	64.0
2011-01-10 00:00:00	38.0	0.0	0.0	74.0
...	...	...	...	...

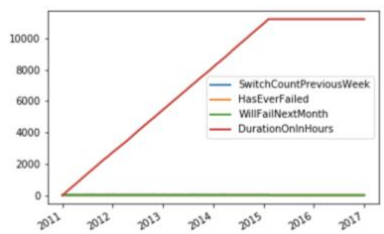
Chart View

pyc3.tsutils is a library written to operate on timeseries. Convenient static and interactive plotting functions are available in this library.

Static Plotting

In [12]:

```
c3Utils.plot(ts)
```



TimeSeries objects can be sliced 2 ways:

- by signal (i.e. Metrics)
- by time

5. Analytics

C3.ai Device PM runs several analytics that generate operator alerts when certain thresholds are passed. This alerts help the operator track important events related to the device's in the network. Below is a list of the analytics that are run in C3.ai Device PM, with details around the inputs, outputs and the underlying algorithmic logic.

Name	Defective Alert
Description	Alerts when device has become defective (device status is on but no output is generated)
Inputs	1. Data a. Output measurements b. Status (history) 2. Time Grain: Hourly
Outputs	Write event with event code DEFECTIVE to device event type, supplying timestamp of defective alert
Algorithm	1. For all devices, identify for devices the value of the defective metric (output = 0 while status = 1) has changed from 0 to 1 2. For devices where defective metric is 1, make an event with event code DEFECTIVE, supplying timestamp of this event

Name	Overheat Alert
Description	Alerts when device surpasses a temperature threshold and is overheating
Inputs	1. Data a. Temperature measurements 2. Time Grain: Hourly
Outputs	Write event with event code OVERHEAT to device event type, supplying timestamp of overheat alert as well as temperature value
Algorithm	1. For all devices, identify which ones have surpassed the 95 degrees Fahrenheit threshold 2. For devices that have surpassed the 95 degrees Fahrenheit threshold, make an event with event code OVERHEAT, supplying timestamp of this event as well as temperature value at time of event

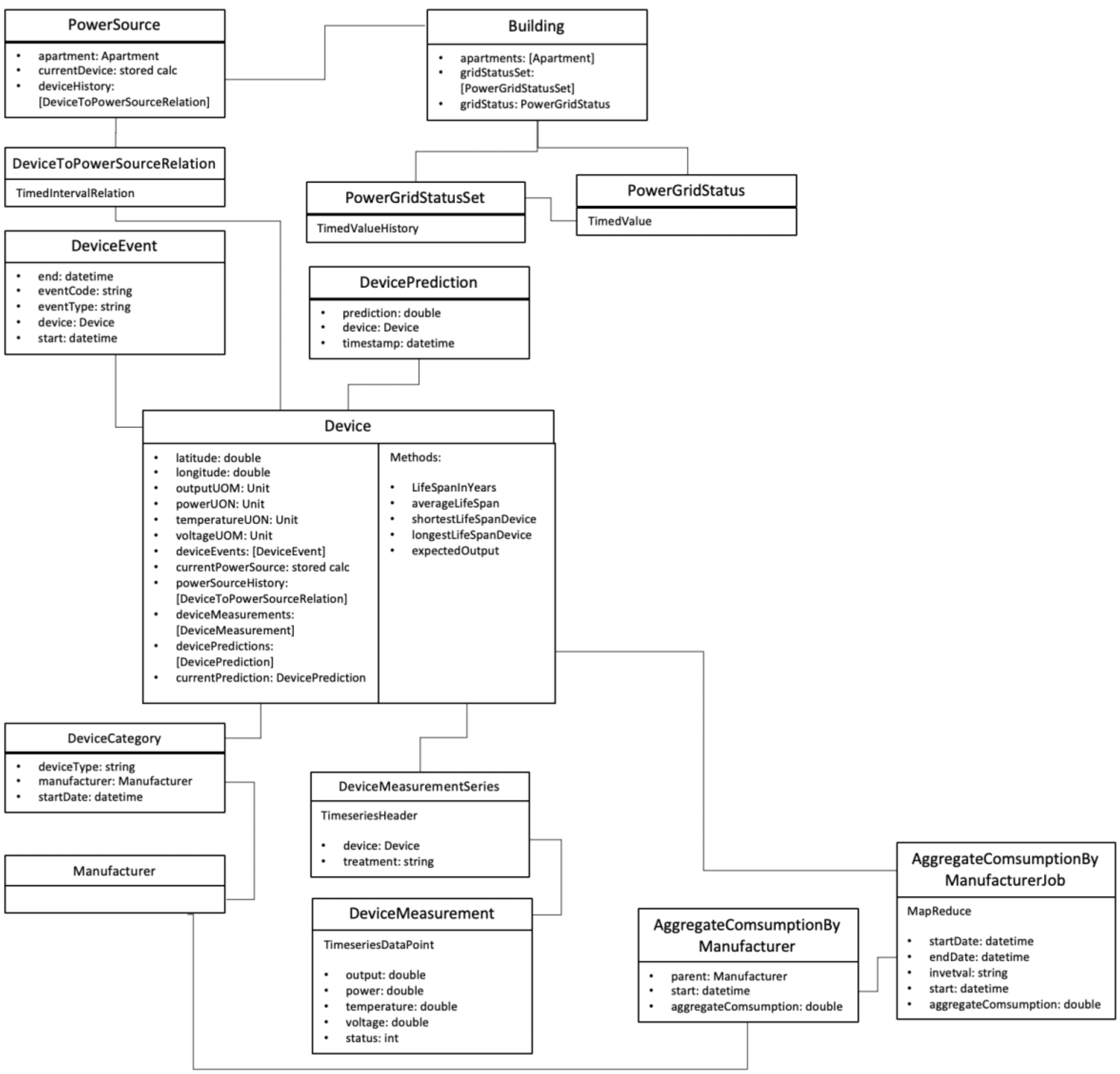
Name	Long Life Alert
Description	Alerts when device surpasses expected lifetime threshold of 10,500 hours
Inputs	1. Data a. Status (history) 2. Time Grain: Hourly
Outputs	Write event with event code LONGLIFE to device event type when device surpasses lifetime threshold
Algorithm	1. For all devices, calculate the duration that a device has been on over the course of its lifetime 2. Identify which devices have surpassed the 10,500-hour threshold 3. For the devices that have surpassed the 10,500-hour threshold make an event with event code LONGLIFE, supplying timestamp at which this threshold was reached

 |

6. Data Model and Data Sources

Below (Figure 4) is an overview of the device data model. The data model starts off with the Device type that holds the basic characteristics of the device. This type is extended to the Device type. DeviceMeasurement is linked to Device through DeviceMeasurementSeries and holds measurement data on all devices. Furthermore, the DeviceStatus type indicates the status of every device and the Events type holds the events acting on Device. Every Device is connected to a Power Source through a timed interval relation (DevicePowerSource) and is placed in a building. Finally, we predict device failures and store the predictions on the DevicePrediction type.

Figure 4. Device Data Model *(Detailed Data Model in Appendix)*



 |

Data sources provide the data to be leveraged by the application. Below is a list of data sources and the fields expected from each.

Data Source Name	**Expected Fields**
Devices	Manufacturer, DeviceType, SN, StartDate, Latitude, Longitude
PowerGridStatus	TS, Building, Status
PowerSource	Power source, Building
DeviceEvent	id, device, eventCode, eventType, start, end
DevicePowerSource	SN, power source, start, end,
DeviceMeasurement	TS, SN, Status, Output, Voltage, Temp
DeviceWeather	WeatherStation, WSLat, WSLong, TS, Temperature, Precipitation, WeatherEvent

External Data Sources

C3.ai Device PM has pre-built integrations to various external sources of information to support predictive maintenance and monitoring activities. These extraprise sources provide supplemental information to help property managers keep their device network healthy.

Some sources include:

- **WeatherAware:** Weather data is retrieved from an external weather data provider and is stored on the observation type associated with a weather station that covers a specific geographic region

7. **Functional Testing**

Functional testing is to be performed on all the metrics and functions populating the dashboard by writing tests with predefined inputs and expected outputs. The same holds for the analytics that are being triggered.

Furthermore, aggregation and filtering functionality needs to be tested by uploading a data set of 10 devices with different manufacturers and device types, with the devices spread across at least 2 buildings. This will allow for verification of filter functionality on the building, manufacturer and device type filters.

Next to that, at least one failed device and one device that is at risk should be part of the data set. This will allow for testing of the overview part of the dashboard, as well as the "devices at risk" histogram. Adding the failed device to the test data will also allow for testing of the status map. The device list can be cross-checked against the imported data sample.

Finally, streaming data functionality needs to be tested by uploading a new data set and evaluating whether the dashboard updates the values correctly (additional devices as well as new device data).

Table 1. Performance, Scalability and Reliability

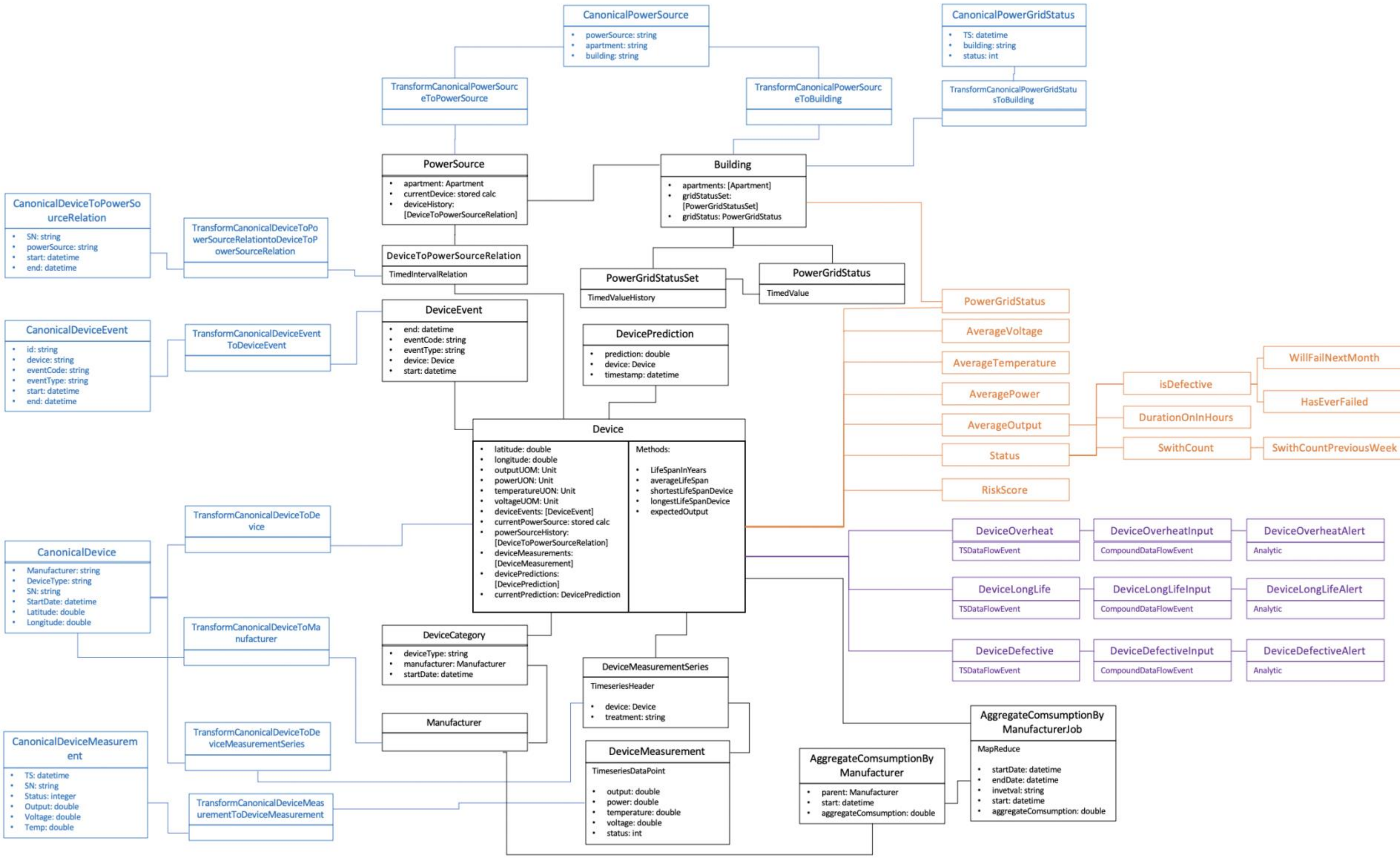
Page / Action	Performance	Scalability	Reliability
Open / Refresh Device Dashboard	< 1 second	0 ramping to 100 users per environment (10k devices per user)	100 users at 95% uptime
Populate Overview and List Components	< 2 seconds	0 ramping to 100 users per environment (10k devices per user)	100 users at 95% uptime
Populate Histogram and Status Map	< 4 seconds	0 ramping to 100 users per environment (10k devices per user)	100 users at 95% uptime
Filter Device Dashboard	< 2 seconds	0 ramping to 100 users per environment (10k devices per user)	100 users at 95% uptime
Open / Refresh Device Analysis Page	< 1 seconds	0 ramping to 100 users per environment (10k devices per user)	100 users at 95% uptime
Populate Device Analysis Page	< 2 seconds	0 ramping to 100 users per environment (10k devices per user)	100 users at 95% uptime
Change Measurement in Device Analysis Graph	< 2 seconds	0 ramping to 100 users per environment (10k devices per user)	100 users at 95% uptime

8. Appendix

 |

Comprehensive Data Model

Types | Canonicals/Transforms | Metrics | Analytics



C3.ai

Detailed Metrics Descriptions

Name	AverageOutput_Device
Description	Average output over time of the device
Inputs	Normalized historical device output measurements
Outputs	A timeseries for the devices, reporting intervals, and time range specified, containing averaged output data
Algorithm	For given devices, average the output measurement data for the time period provided and aggregate/disaggregate as needed for the requested reporting intervals

Name	AveragePower_Device
Description	Average power over time of the device
Inputs	Normalized historical device power measurements
Outputs	A timeseries for the devices, reporting intervals, and time range specified, containing averaged power data
Algorithm	For given devices, average the power measurement data for the time period provided and aggregate/disaggregate as needed for the requested reporting intervals

Name	AverageTemperature_Device
Description	Average temperature over time of the device
Inputs	Normalized historical device temperature measurements
Outputs	A timeseries for the devices, reporting intervals, and time range specified, containing averaged temperature data
Algorithm	For given devices, average the temperature measurement data for the time period provided and aggregate/disaggregate as needed for the requested reporting intervals

Name	AverageVoltage_Device
Description	Average voltage over time of the device
Inputs	Normalized historical device voltage measurements
Outputs	A timeseries for the devices, reporting intervals, and time range specified, containing averaged voltage data
Algorithm	For given devices, average the voltage measurement data for the time period provided and aggregate/disaggregate as needed for the requested reporting intervals

Name	PowerGridStatus_Building
Description	Status (1 or 0) indicating ON or OFF of the power grid over time
Inputs	Historical building power grid measurements
Outputs	A timeseries for the buildings, reporting intervals, and time range specified, containing normalized (0 or 1) power grid status data
Algorithm	For given buildings, normalize & interpolate the power grid status data for the time period provided and aggregate/disaggregate as needed for the requested reporting intervals. If there is no data for that time, maintain the last reported power grid status value.

Name	PowerGridStatus_Device
Description	Status (1 or 0) indicating ON or OFF of the power grid over time
Inputs	Historical building power grid measurements
Outputs	A timeseries for the devices, reporting intervals, and time range specified, containing normalized (0 or 1) power grid status data
Algorithm	For given devices, normalize & interpolate the power grid status data for the time period provided and aggregate/disaggregate as needed for the requested reporting intervals. If there is no data for that time, maintain the last reported power grid status value.

Name	Status_Device
Description	Status (1 or 0) indicating ON or OFF of the device over time
Inputs	Historical device status measurements
Outputs	A timeseries for the devices, reporting intervals, and time range specified, containing normalized (0 or 1) device status data
Algorithm	For given devices, normalize the status data to the maximum value for the time period provided and aggregate/disaggregate as needed for the requested reporting intervals.

Name	DurationOnInHours
Description	The total amount of time a device is switched on up to the interval
Inputs	A normalized timeseries of device status (0 or 1)
Outputs	A timeseries for the devices, reporting intervals, and time range specified, containing the total number of hours the device's status was on (1)
Algorithm	For given devices, cumulatively sum the number of hours the device was on for the time period provided and aggregate/disaggregate as needed for the requested reporting intervals.

Name	HasEverFailed
Description	Indicates (1 or 0) if the device has failed during evaluated interval
Inputs	A normalized timeseries of device status (0 or 1), a normalized timeseries of device output measurements, and a normalized timeseries of the power grid status (0 or 1)
Outputs	A timeseries for the devices, reporting intervals, and time range specified, containing 0 or 1
Algorithm	For given devices, evaluate if the device has ever failed (been defective) for the time period provided and aggregate/disaggregate as needed for the requested reporting intervals.

Name	IsDefective
Description	Indicator (1 or 0) if the device died
Inputs	A normalized timeseries of device status (0 or 1), a normalized timeseries of device output measurements, and a normalized timeseries of the power grid status (0 or 1)
Outputs	A timeseries for the devices, reporting intervals, and time range specified, containing 0 or 1
Algorithm	For given devices, evaluate if the output were ever 0, power grid status was 1, and the device status was 1 for the time period provided to 1 if true and 0 if false, and aggregate/disaggregate as needed for the requested reporting intervals.

Name	SwitchCount
Description	The number of times a device is switched on or off beginning at the 'start' given
Inputs	A normalized timeseries of device status (0 or 1)
Outputs	A timeseries for the devices, reporting intervals, and time range specified, containing the number of times the device status changed during the time period
Algorithm	For given devices, sum the number of times the device status changed for the time period and aggregate/disaggregate as needed for the requested reporting intervals.

Name	SwitchCountPreviousWeek
Description	The total number of times a device is switched on or off up in the previous week of the interval
Inputs	A normalized timeseries of device status (0 or 1)
Outputs	A timeseries for the devices, reporting intervals, and time range specified, containing the number of times the device status changed during the 7 days prior to the time period
Algorithm	For given devices, sum the number of times the device status changed for the previous seven days, moving with the time period, and aggregate/disaggregate as needed for the requested reporting intervals. (ie: On the 7th, report the switch count from the 1st to the 7th. On the 8th, report from the 2nd to the 8th)

Name	WillFailNextMonth
Description	Indicator (1 or 0) if the device died within the next 30 days
Inputs	A normalized timeseries of device status (0 or 1), a normalized timeseries of device output measurements, and a normalized timeseries of the power grid status (0 or 1)
Outputs	A timeseries for the devices, reporting intervals, and time range specified, containing 0 or 1
Algorithm	For given devices, evaluate if the device will be considered defective within the next 30 days from the time period and aggregate/disaggregate as needed for the requested reporting intervals. (ie: On the 7th, report if the device will be considered defective at any point up to the 7th of the next month. On the 8th, report up to the 8th of the next month)

