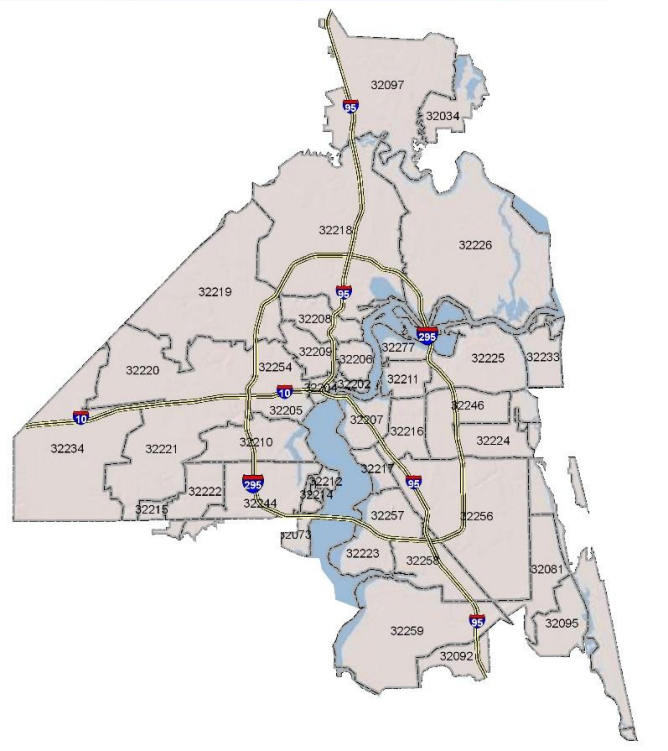
JEA IIOT Proof of Concept Progress with Dianomic FogLAMP



•JEA at a Glance

- Electric, Water and Sewer Utility of Jacksonville, Florida
- 8th largest municipal electric utility in the US
- Service area approximately 900 square miles
- 7000 miles of electric distribution line – 56% underground
- Customers:
  - Electric – 470,000
  - Water – 350,000
  - Sewer – 272,000
- Approximately 1900 employees
- 2017 System Peak:
  - 2827 MW Winter
  - 2776 MW Summer

Pressures on our Business

New or improved technologies have forced changes to business models.

- Available consumer products for energy efficiency are cheaper and improving exponentially.
- Greater push from regulators and political groups for clean energy.
- Stakeholders expect greater operating efficiencies, customer, and community service.

Managers Always Ask the Tough Questions

"Well, the SCADA system has pumping volumes so we know what is going in"

"The GIS and other systems know where the piping is and how it is connected"

How do we find leaks in our water distribution system?

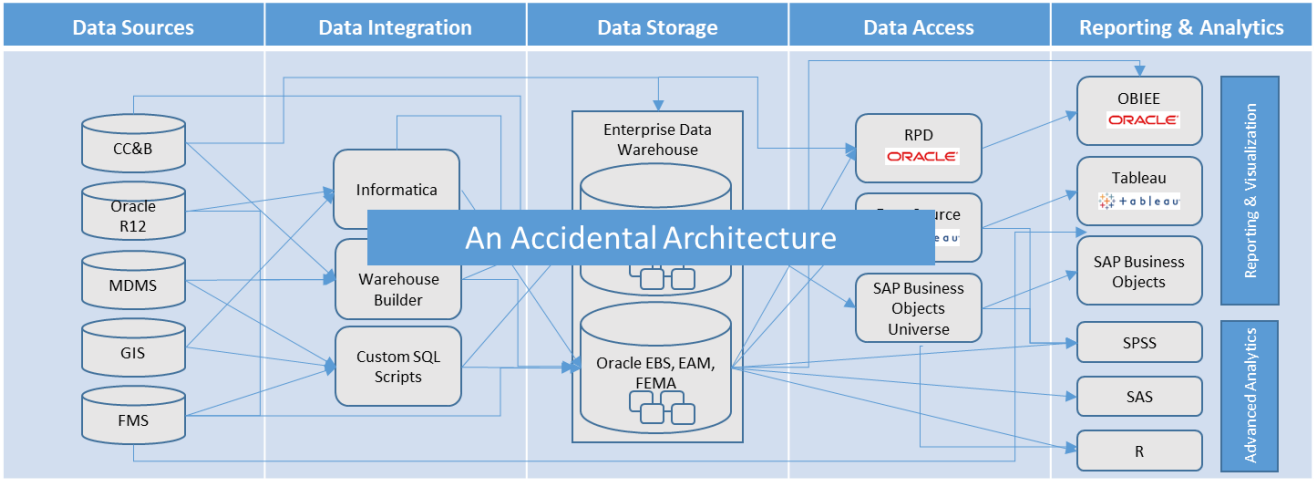
"Customer billing knows how much is going out"

"We will need to add some additional sensors to close the gaps"

How do we get all of this data and make sense of it?

- There is no catalog of what data is available from where
- Integration with different systems is haphazard and non-standardized
- In short, a whole lot of work seeking out the expert for each system and working out a plan to access data
- I can FTP you a CSV file!

Our Current State – Accidental Architecture



- Primarily traditional RDBMS, flat-files
- Not leveraging all available data sources and types (80% unstructured)

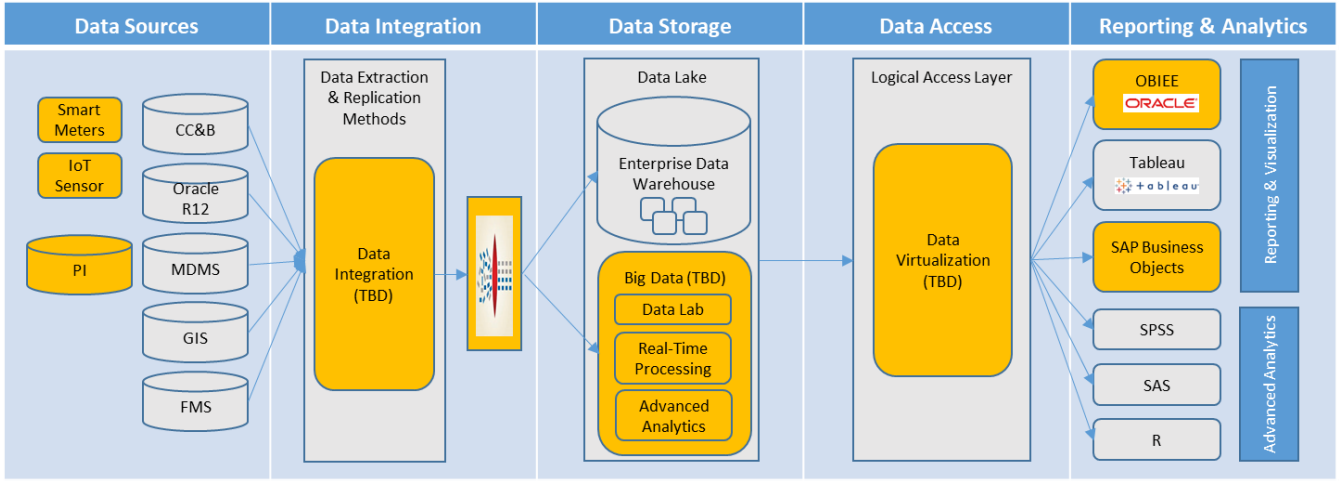
- Limited data integration capabilities
- Additional overhead on source systems to extract data
- Does not support IoT/sensor real-time streaming

- Subject-oriented
- Limited data sets
- Data not in raw-state
- Limited storage capacity
- Not designed to support Big Data analysis

- Logical layer coupled to data source & vendor
- Difficult to navigate for users

- SAP Business Objects unsupported by vendor
- Primarily pre-defined reports provided by BI team

Where We Want to Be



- Provides ability to source data any data source
- **Introduce data from L&G smart meters (Ex: voltage)**
- **PI system upgrade**

- **Supports IoT data streaming**
- Use web services for
- **Introduce data quality rules, data cleansing (Data Governance)**

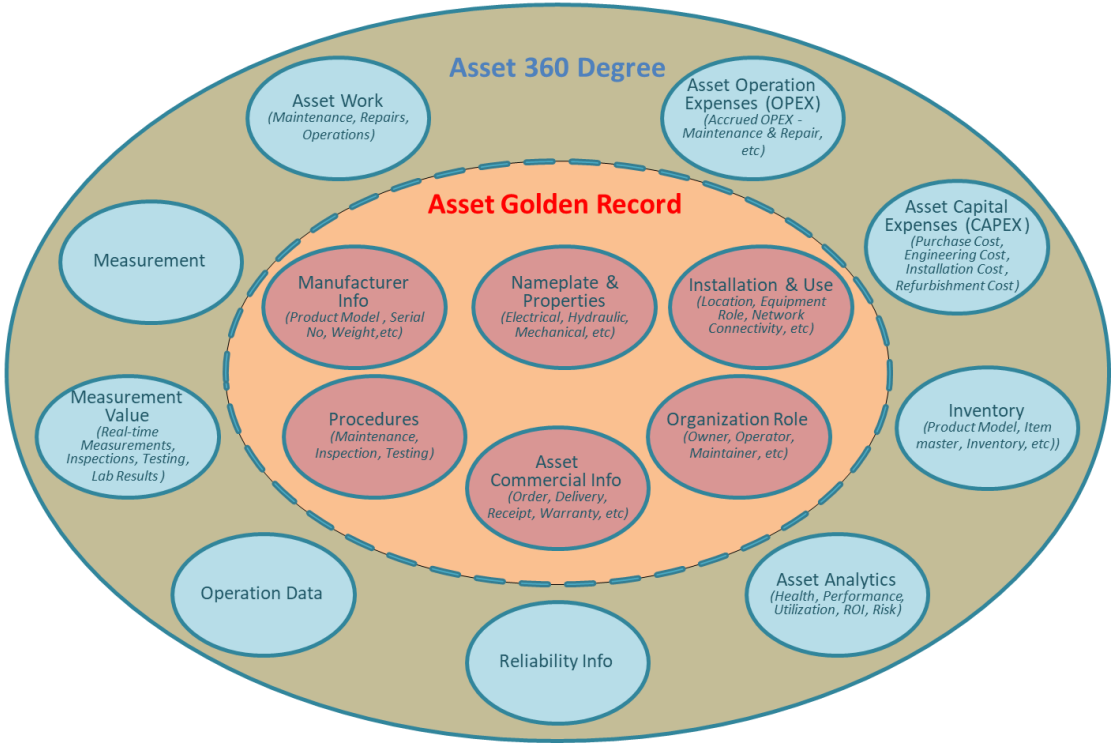
- Serves as central data repository
- Able to store large volumes of historical data
- **Data Lab for testing analytic use cases**

- Data Access layer based on industry standards
- **Based on business context, not technical**
- **Single, consistent view to users**

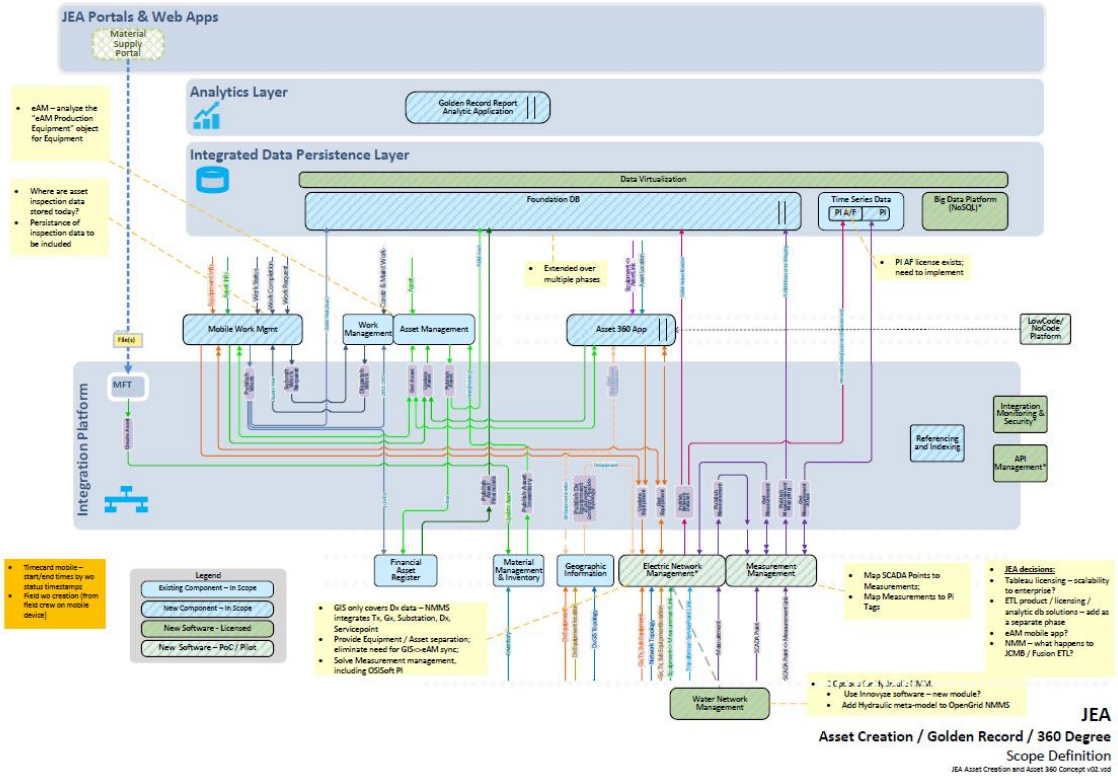
- **Customer chooses technology**
- **Leverage cloud applications for fast time to market**
- Introduce latest advanced analytics technology

FY19 Focus Area

Data From The Asset Management Perspective



JEA/Xtensible Asset 360 Information Model



The Low Altitude View – Dealing with the Edge

Transformer explosion picture

- **Problem** – Expensive equipment failures that could have been prevented if more information available. A single incident can cost over $1 million.
- **Opportunity** – Relatively inexpensive sensors are available to collect information that could prevent failures.
- **Challenge** – Tens, Hundreds, Thousands of these new inexpensive sensors will need to be integrated into the network

Integrating these Edge Devices into the Network

Edge Device

Rich Data Stream of Sensor Information from Edge Devices

"You want to put how many devices on the network?"

"We'll take a few of the tags, but not everything."

Existing Industrial Control Systems

"We don't have enough licensed tags to hold all of that information."

"We have enough trouble managing our network already and this information is not critical to system operation."

"Load is high, we are hands off this week."

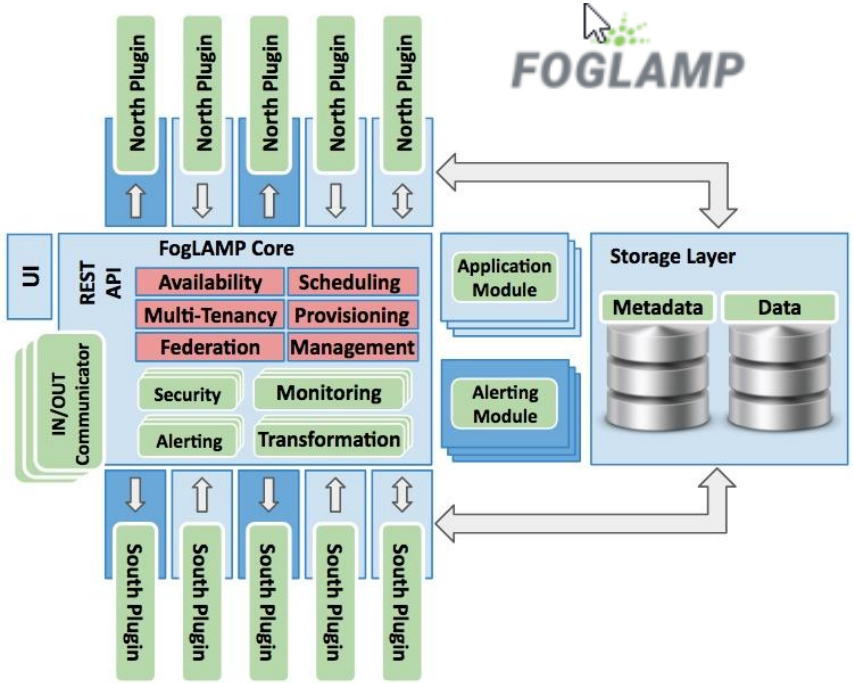
In Comes the Fog

- Build a separate network to handle these Edge devices.
- Non-critical network separate from SCADA so low possibility of disruption.
- Separate network of instrumentation does not require same level of security as primary control system.
- **Common platform** to be used everywhere while SCADA, DCS, and PLC systems are different.
- Optimized to collect data – filter and process, collect events.

Where can we find such a platform?

Enter Dianomic FogLAMP

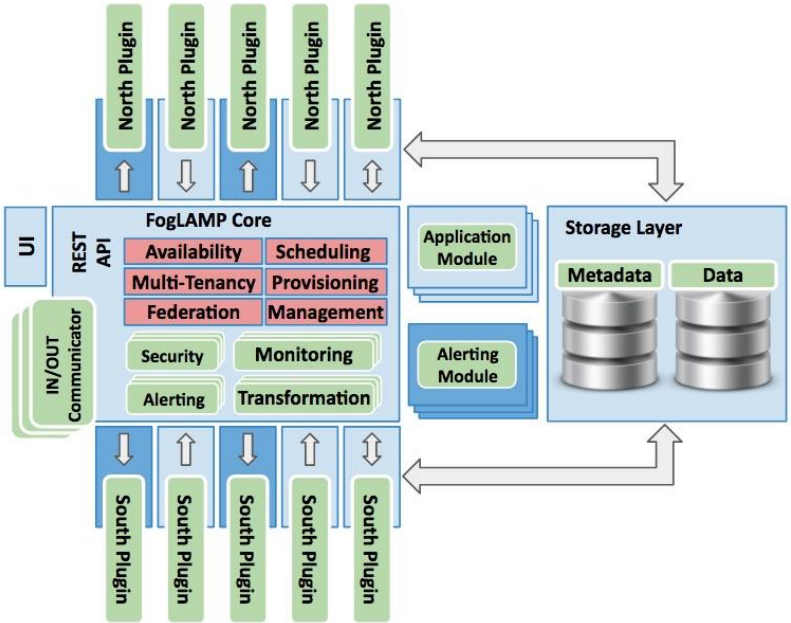
- Linux-Based Microservice architecture
- Open Source
- Offers All of the Basic Services Needed and More



Open Platform is Crucial

- Many of the business units are pursuing point solutions from vendors who already have a data collection infrastructure they support, often cloud-based.
- Since FogLAMP is relatively small and open, it is not unreasonable to ask vendor to add it so that we can plug into a standardized infrastructure.
- Business unit can still use vendor solution while data is plugged into the Enterprise system.

Think of This Part as A PI Interface that Runs on a Small Linux System



North Plugins to forward data upstream. Built in Plugins for PI Server and OSIsoft Cloud Services using PI Relay. Upstream system may also be another FogLAMP node.

Storage Layer is like PI Buffer Subsystem to temporarily store data until it is forwarded upstream

South Plugins to communicate with Edge Devices. Our initial POC uses Modbus

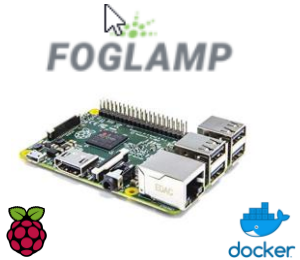
We Gave it a Quick Try – *Very Cool*

Edge Device

Ethernet Protocols:

- Modbus
- DNP3
- IEC 61850

Many data items available



- Developed Dockerfile to “Dockerize” FogLAMP
- Developed South Plugin to collect data from sensor
- Configured North HTTP Plugin to forward data upstream

FOGLAMP

Ubuntu Linux 18.04 VM

- Configured South HTTP Plugin to collect inbound
- Configured PI Server Plugin to send data to PI Relay
- Automatically configures PI tags and populates AF

Where to Run FogLAMP – Successes so far

- Raspberry PI – both natively and in Docker container
- Ubuntu Server and Workstation – both natively and in Docker
- Amazon AWS Docker container

Where to Run FogLAMP – Cisco IOx

- JEA already had plans to upgrade substation networks using new Cisco IE-9000 switches
- These switches support the Cisco IOx Edge platform allowing us to deploy Docker images, but having issues with Ubuntu-based images
- So far no success, but IOx can also run a Linux VM, so we will be trying that.

Where to Run FogLAMP - Schweitzer

- JEA already has SEL industrial computers at substations. Can we run FogLAMP there?
- The SEL computers run Security Enhanced (SE) Linux.
- We are going to get a demo unit to try.
- Goal will be to deploy Docker.

“Dockerizing” FogLAMP

- Docker packages everything up in an easily deployable container
- Ideally we should be able to deploy the same image to anything supporting Docker
- “Dockerization” of FogLAMP is pretty straightforward

Docker Example – (Very condensed to fit slide)

```
FROM ubuntu:18.04

RUN apt-get update && apt-get install -y \
apt-utils \
G++ \
automake \
Sqlite3

WORKDIR /foglamp
RUN git clone https://github.com/foglamp/FogLAMP.git /foglamp \
&& git checkout v1.5.1 \
&& make \
&& make install

VOLUME /usr/local/foglamp/data
EXPOSE 8081 1995

CMD ["bash", "/usr/local/foglamp/foglamp.sh"]
```

Some Paradigm Shifts

- PI interfaces typically connect to a configurable control device, so tags depend on program running
- IOT devices typically offer a set collection of information that does not change unless firmware is upgraded
- South Plugin is developed for specific device
- Configuration is simplified, for our case, just give it the IP address of the device.

Data Structure for IOT Device

OSIsoft Cloud Services (OCS) supports these types of structure natively

```
{
        asset: “Robinwood:Transformer1”,
        timestamp: “2019-03-21 13:55:14 +0000”,
        key: “1a4910c8-cff2-4451-b5d0-75134737”
        readings : {
                LTC_TANK_TEMP: “48”,
                PTC_TOP_TEMP: “49.3”
                }
}
```

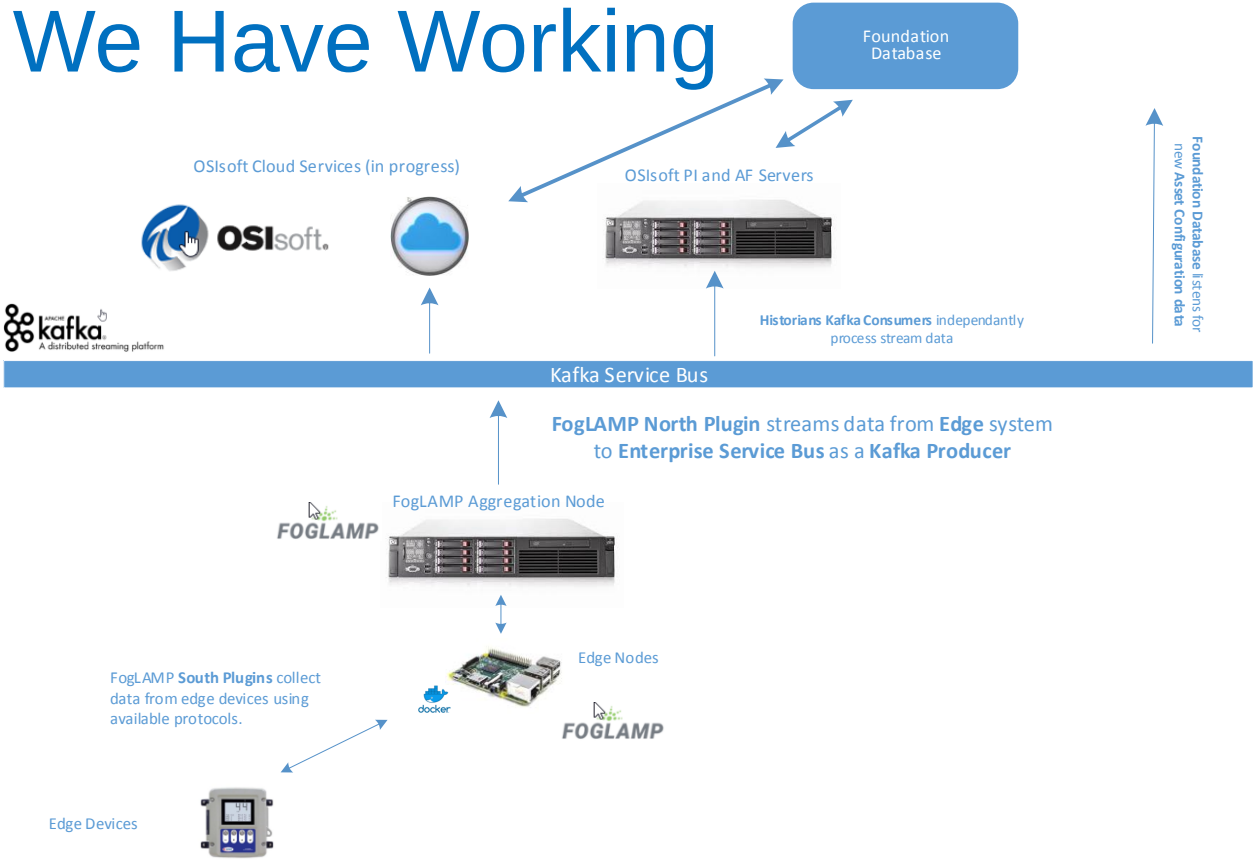
Don’t Let the Programming Scare You

```
from pymodbus.client.sync import ModbusTcpClient as ModbusClient
from pymodbus.transaction import ModbusRtuFramer as ModbusFramer
from pymodbus.constants import Endian
from pymodbus.payload import BinaryPayloadDecoder

# read LTC Tank Temperature
try:
        ltc_tank_temp_read =
        modbus_client.read_input_registers(LTC_TANK_TEMP_REGISTER,2,unit=1)
        ltc_tank_temp_raw = ltc_tank_temp_read.registers[0]
        ltc_tank_temp = convert_to_scaled_signedint(val,1000)
except Exception as ex:
ltc_tank_temp = f'error: {ex}'

readings = {
'ltc_tank_temp': ltc_tank_temp,
'top_oil_temp': top_oil_temp
}
```

Don’t Let the Programming Scare You

- South Plugins can be developed in C++ or Python
- Python has readily available libraries for Modbus and DNP3
- IEC61850 has a C++ library available
- After you have a template it is plug and chug.

What We Have Working



Upcoming

- Implement another Kafka topic for alerts and build FogLAMP notification plugin
- Deployment to substation for field testing
- Development of new plugins for additional devices
- More complex use cases

Questions?

Please wait for the **microphone**

State your **name & company**

Please remember