Data Collection at the Edge

Randal Glass – Sr. Software Developer
Frank Gasparro - Engineering Group Lead
Greg Holt – Principal Software Developer

 ©2018 OSIsoft, LLC

Who's Who

- **Randy Glass**
- rglass@osisoft.com
- Sr. Software Developer
- OSIsoft, LLC

- **Greg Holt**
- gholt@osisoft.com
- Principal Software Developer
- OSIsoft, LLC

- **Frank Gasparro**
- fgasparro@osisoft.com
- Engineering Group Lead
- OSIsoft, LLC

©2018 OSIsoft, LLC

Edge Data Store



- Persistent Storage
- Self-Healing
- Application Platform

- Cross Platform (Windows, Linux)
- Multiple Data Ingress Options
- Analytics Ready
- Egress to PI Server and OSIsoft Cloud Services
- Developed, Sold, and Supported by OSIsoft

©2018 OSIsoft, LLC

What is the Edge Data Store (Edge DS)?

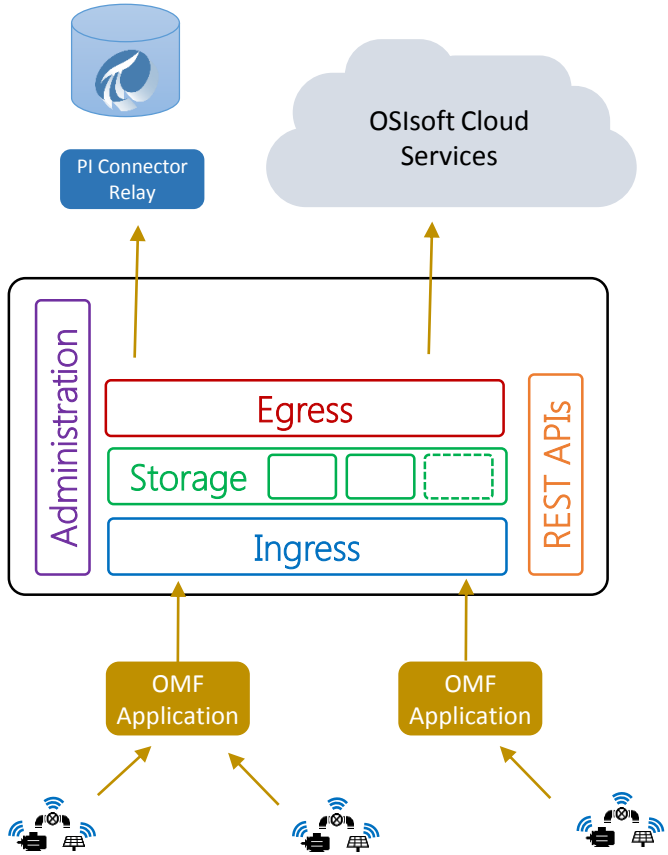
- Edge DS is not an “historian”
- Edge DS is a “sequential” database
  - Anything that can be ordered can be stored
  - Simple or Complex data types can be stored
  - Multiple indexes can be assigned
    - Non-time series
    - Compound
    - Secondary
  - No distinction between future and past data
  - Unbounded Event Size
  - High – precision timestamps (100ns)



©2018 OSIsoft, LLC

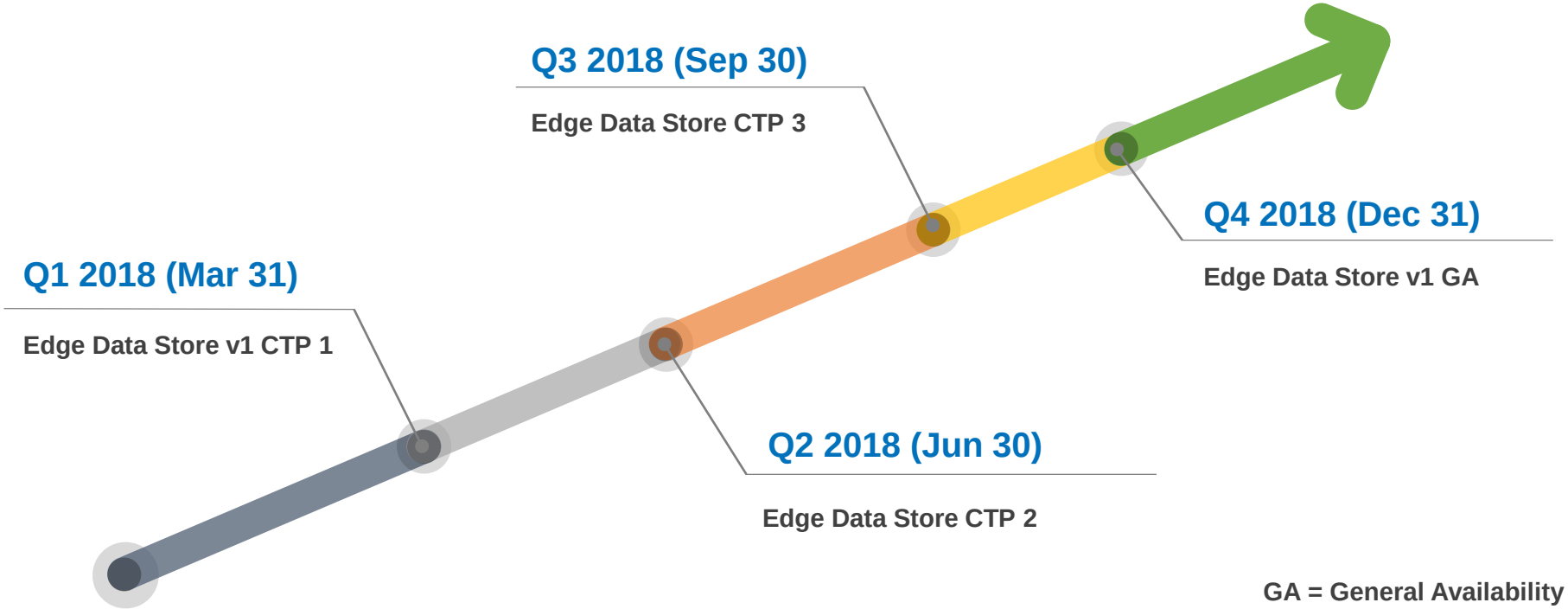
Edge DS – CTP 1 Capabilities

- Administration
  - Basic Settings for storage, Egress
- Ingress
  - data ingestion into Edge via OMF
- Storage
  - data persistence
  - Storage limits and trimming
- Egress
  - Send data to OCS or to PI System via PI Connector Relay
  - Egress Rules via API
- Rest API
  - Use the REST API to perform administrative tasks or build applications
- Capacity/Throughput
  - Up to 2500 Streams
  - Up to 1 Second Data, 2500 Events/sec
  - Expect ~512 MB ram utilization @ this capacity
- Device/OS Requirements
  - 1GB RAM or Greater
  - SSD Storage 32 GB or Greater
  - Multi Core CPU – Intel or ARM
  - Windows 10, 2016
  - Linux – Ubuntu 16.04 (others may work to subject to .NET Core support
  - Linux ARM – Debian 9 (others may work too)
  - Docker Images for Linux



©2018 OSIsoft, LLC

Edge Data Store Development Timeline



GA = General Availability (released to market)
CTP = Customer Technology Preview (not ready for release)

©2018 OSIsoft, LLC

How do I get started?

- Where do we get the Edge DS Bits?
  - CTP 1 is by invitation only
  - https://techsupport.osisoft.com/Downloads/All-Downloads/All-Groups/All-Products/All-Categories/Current-Version/edge/
- Where do I find code samples and libraries
  - https://github.com/osisoft/
- Documents
  - http://osisoft-edge.readthedocs.io/en/latest/EdgeDS_Admin.html

©2018 OSIsoft, LLC

Our sample applications

- FAADemo – Application that publishes real-time aircraft telemetry into Edge
  - Publishes data to Edge via OMF
- FAAMap – Application that visualizes historical routes of aircraft
  - Queries data from Edge via REST endpoint

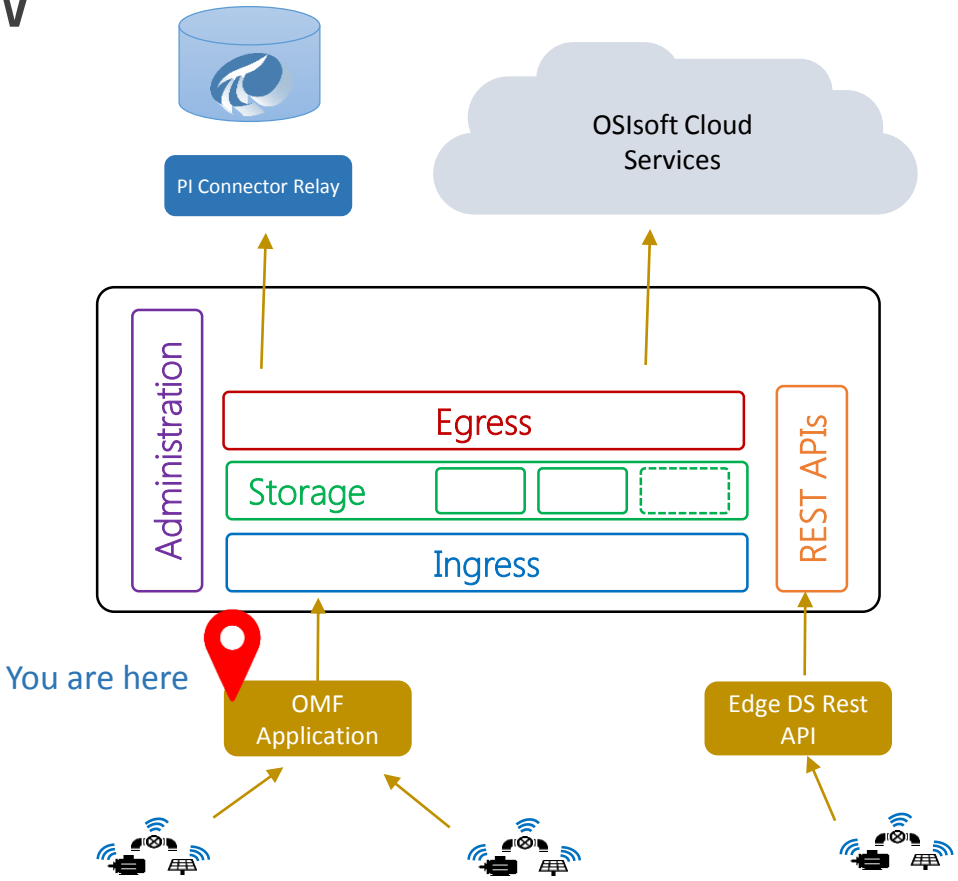
©2018 OSIsoft, LLC

Edge Data Store data ingress with OMF

Randy Glass

 ©2018 OSIsoft, LLC

Edge DS Overview



©2018 OSIsoft, LLC

Getting data into Edge Data Store

- Provide the developer with good options for data ingress:
  - OMF Ingress – common standard across OSIsoft
  - REST Ingress – similar surface area as OCS
  - REST API still used for Queries, Management
- These options protect your investment in app development
- Both Ingress options are straightforward, approachable with different languages
- The programming model is similar in either case…

©2018 OSIsoft, LLC

Ingress - Code Pattern

1. Create Tenant, Namespace (if desired)
   - Use REST API
2. Create Type
3. Create Streams (Containers)
4. Send data associated with stream

©2018 OSIsoft, LLC

Ingress – Code Pattern

- Objects are passed as JSON Strings via HTTPS requests
- Helpful to use library for JSON generation – e.g. in .Net
  - Newtonsoft Json.NET
  - OMF Ingress – OSIsoft.Parse.OMFGeneration
  - REST API – OSIsoft Qi Client Library
- App must provide apropos HTTP header values
  - Ingress always expects "producertoken"
  - OMF – message type, action, etc. expected

©2018 OSIsoft, LLC

Ingress

```
public class AircraftInFlight
{
    public int Altitude { get; set; }
    public string Dest { get; set; }
    public int Heading { get; set; }
    public DateTime TimeId { get; set; }
   // …
}
```

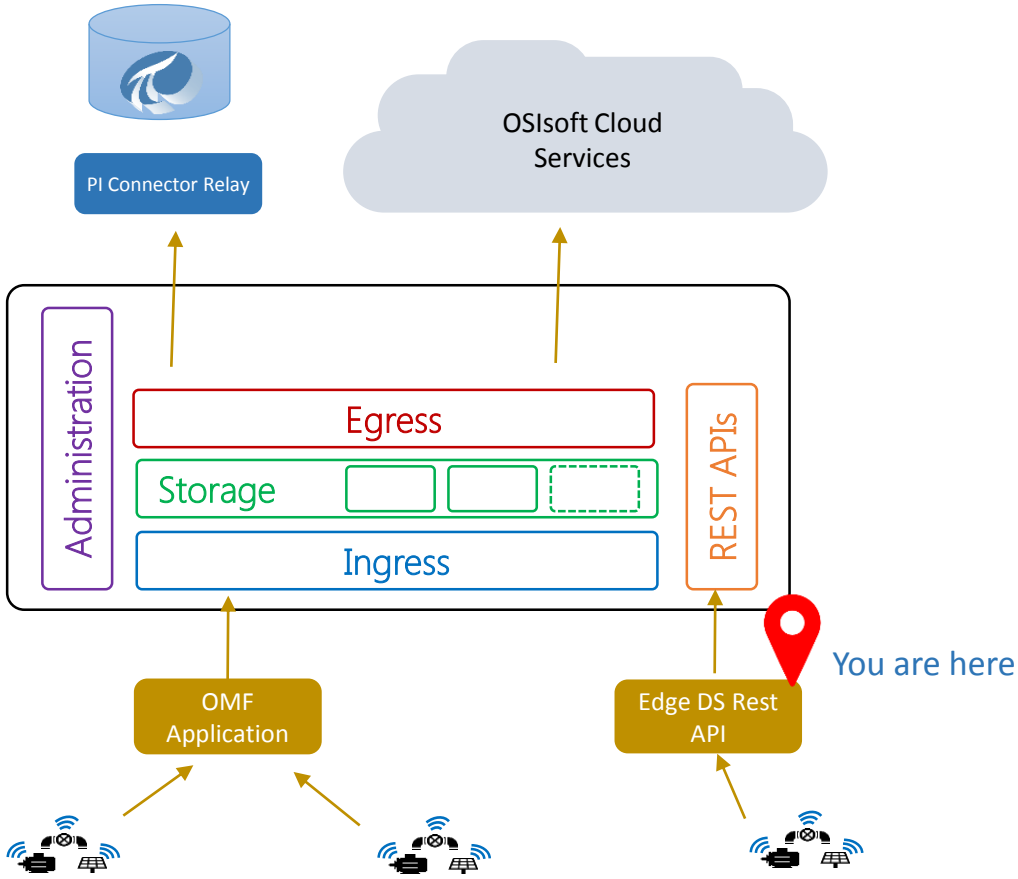

©2018 OSIsoft, LLC

Edge Data Store Rest API

Greg Holt

 ©2018 OSIsoft, LLC

Edge DS Overview



©2018 OSIsoft, LLC

What are we going to cover?

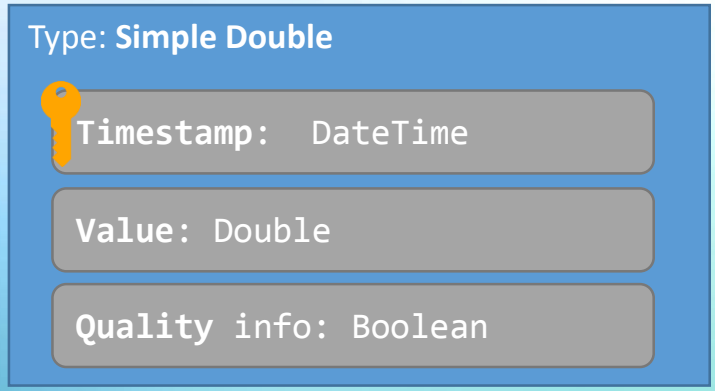
- Performing Queries against Streams
- Edge Data Store Specific Methods
- Resiliency
- Sending data to other OSIsoft systems
  - What exactly is Egressed?
  - Configuring EgressTarget
  - Configuring EgressRules

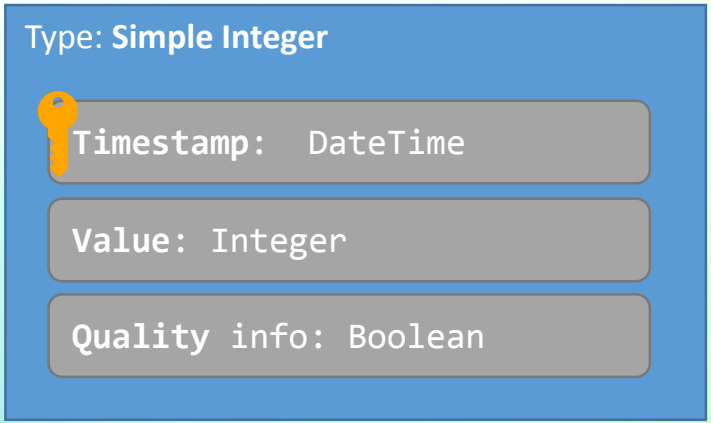
©2018 OSIsoft, LLC

But 1st a little about Edge and OCS Types and Streams

©2018 OSIsoft, LLC

Data Structure: Types – Simple Cases





©2018 OSIsoft, LLC

Data Structure: Types – Complex Cases

Type: **Platform pressure**

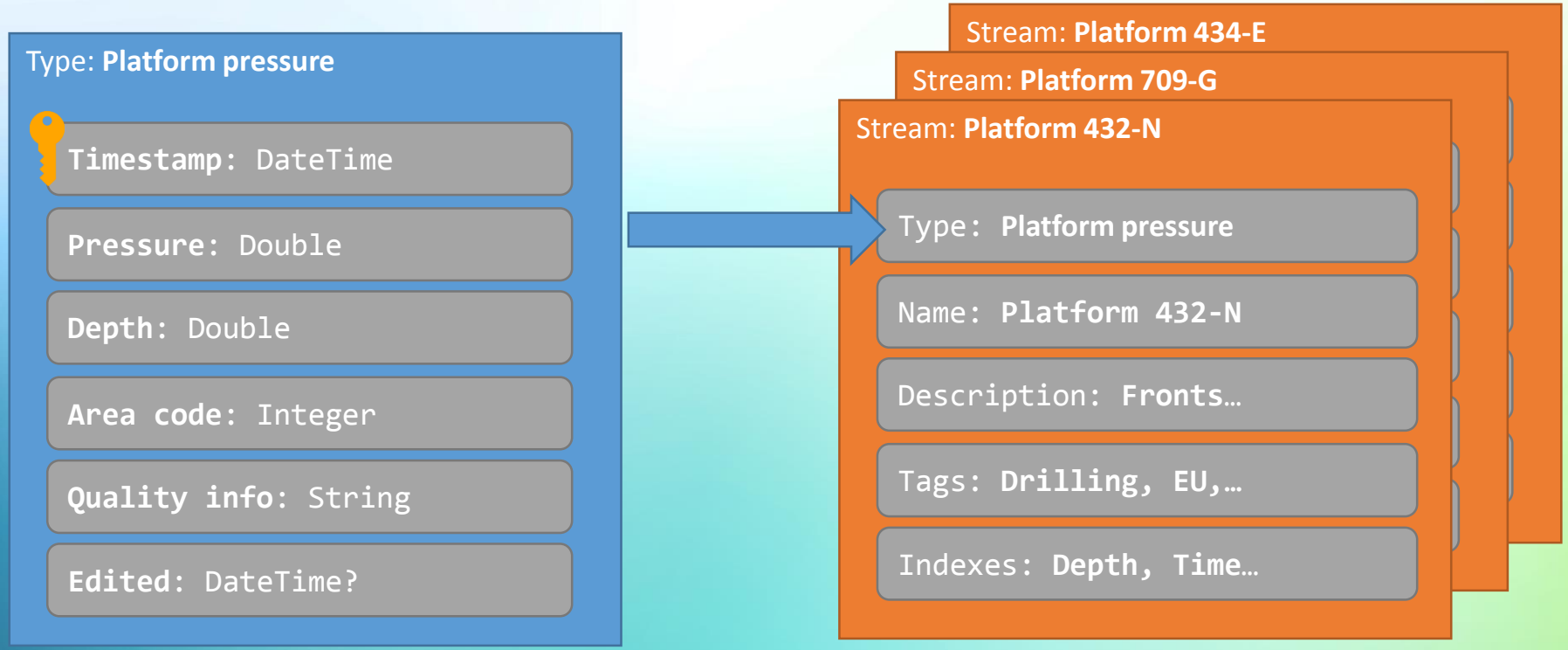
- **Timestamp**: DateTime
- **Pressure**: Double
- **Depth**: Double
- **Area code**: Integer
- **Quality info**: String
- **Edited**: DateTime?

Type: **Batch measurement**

- **Lot number**: Integer
- **pH**: Single
- **Color**: String
- **Weight**: Double
- **Quality info**: String
- Tested: DateTime?

©2018 OSIsoft, LLC

Data Storage: Streams (instances of a Type)



©2018 OSIsoft, LLC

Data Ingress: Streaming data

Stream: Platform 432-N

- Type: **Platform pressure**
- Name: **Platform 432-N**
- Description: **Fronts…**
- Tags: **Drilling, EU,…**
- Indexes: **Depth, Time…**



New event

- Stream: **Platform 432-N**
- Timestamp: **24-Oct-2017…**
- Pressure: **134.0254**
- Depth: **-256.12**
- Area code: **254**
- Quality info: **Quest…**
- Edited: **null**

©2018 OSIsoft, LLC

On with the show!

©2018 OSIsoft, LLC

Merci

谢谢

Danke

Спасибо

Gracias

Thank You

감사합니다

ありがとう

Grazie

Obrigado

Optional: Click to add a takeaway you wish the audience to leave with.

©2018 OSIsoft, LLC