Digital Twins in IoT

Using flux across measurements in real life

David G. Simmons, Senior Developer Evangelist

davidgs@influxdata.com

@davidgsIoT

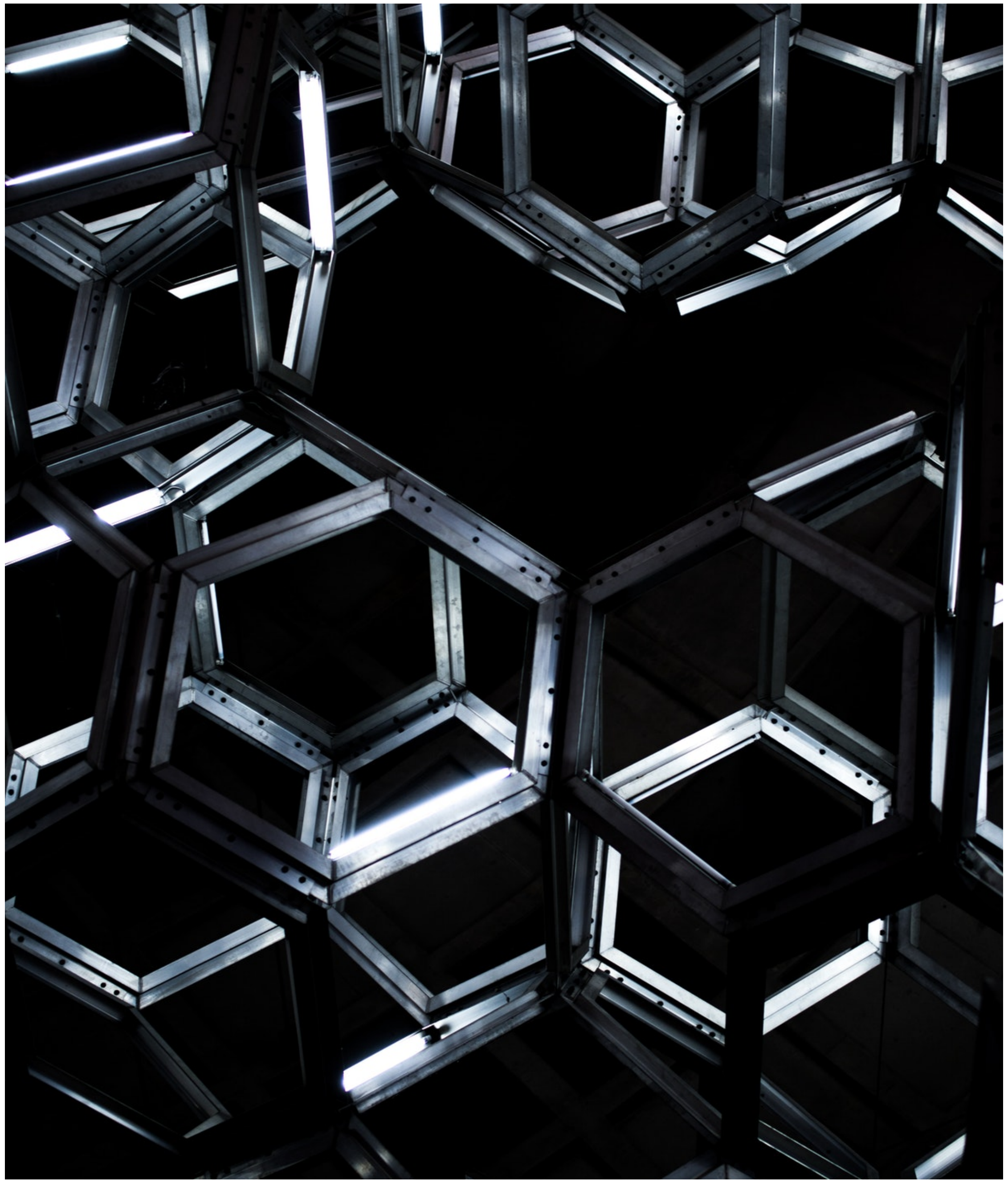
What is a Digital Twin?

- "A digital twin is a real time digital replica of a physical device”
- “Using a digital copy of the physical system to perform real-time optimization”

@davidgsIoT

Why Digital Twins?

- Huge time savings
- Massive cost savings
- Opportunity for innovation and change



What's Required?

- **Lots** of data
    - High speed ingestion of data
    - Efficient storage of data
    - Highly efficient data query
    - Data visualization tools

Manipulating Digital Twin Data

- Ability to do complex calculations on incoming data
- Ability to correlate data across data streams
- Ability to do complex calculations across various incoming data streams

@davidgsIoT

Let’s build a Digital Twin!

- Let’s model this room!
- Environmental monitoring only
- Data we will collect:
  - CO_2 Concentration
  - Temperature
  - Humidity
  - Pressure
  - Particulate matter

Adjusting Values

- The CO_2 sensor — all CO_2 sensors — are affected by atmospheric pressure and temperature
- Simple physics

Table 1: CO_2 Measurement Change With Temperature					
Temp. in °F	CO_2 Measured in PPM	Temp. in °F	CO_2 Measured in PPM	Temp. in °F	CO_2 Measured in PPM
32	1092	60	1033	85	985
35	1085	65	1023	90	976
40	1074	70	1013	95	968
45	1063	75	1004	100	959
50	1053	77	**1000**	105	950
55	1043	80	994	110	942

Table 2: CO_2 Measurement Change with Altitude and Barometric Pressure		
Altitude in Feet	Barometric Pressure in inches Hg	CO_2 Measured in PPM
-1000	31.02	1037
0	29.92	**1000**
1000	28.85	964
2000	27.82	930
3000	26.82	896
4000	25.84	864
5000	24.9	832
6000	23.98	801
7000	23.09	772
8000	22.23	743
9000	21.39	715
10000	20.58	688

Compensating

ppm CO2 corrected = ppm CO2 measured * ((Tmeasured*Pref) / (Pmeasured*Tref))

- Pmeasured = Current pressure, in the same units as reference pressure (not corrected to sea level)
- Tref = reference temperature, usually 25°C, 77°F, converted to absolute (298.15 for °C, 536.67 for °F)
- Tmeasured = Current absolute temperature, °C + 273.15, °F +459.67
- Pref = reference Barometric Pressure, usually sea level, 29.92 in Hg, 760 mm Hg, 1013.207 hPa or 14.6959 psi

Calculating in Flux

```
Tref =298.15
Pref =1013.25
CO2meas = from(bucket: “telegraf/autogen")
  |> range($range)
  |> filter(fn: (r) => r._measurement == "k30_reader" and (r._field == "co2"))
  |> aggregateWindow(every: 30s, fn: mean)
  |> keep(columns: ["_value", "_time"])
```

ppm CO2 corrected = ppm CO2 measured * ((Tmeasured*Pref) / (Pmeasured*Tref))

@davidgsIoT

Calculating in Flux

```
Tmeas =from(bucket: "telegraf/autogen")
    |>range($range)
    |>filter(fn: (r)=>r._measurement =="environment"and(r._field =="temp_c"))
    |>aggregateWindow(every: 30s, fn: mean)
    |>keep(columns: [“_value", "_time"])
```

ppm CO2 corrected = ppm CO2 measured * ((Tmeasured*Pref) / (Pmeasured*Tref))

Calculating in Flux

```
Pmeas = from(bucket: "telegraf/autogen")
    |> range($range)
    |> filter(fn: (r) => r._measurement == "environment" and (r._field ==
"pressure"))
    |> aggregateWindow(every: 30s, fn: mean)
    |> keep(columns: ["_value", "_time"])
```

ppm CO_2 corrected = ppm CO_2 measured * ((Tmeasured*Pref) / (Pmeasured*Tref))

Join all the Tables!

```
first_join = join(tables: {CO2meas: CO2meas, Tmeas: Tmeas}, on: ["_time"])

second_join = join(tables: {first_join: first_join, Pmeas: Pmeas}, on: ["_time"])
    |>map(fn: (r) => ({_time: r._time, _Pmeas: r._value,
_CO2meas:r._value_CO2meas, _Tmeas:r._value_Tmeas}))

final = second_join
    |>map(fn: (r) => ({Pmeas: r._Pmeas, CO2meas:r._CO2meas,
Tmeas:r._Tmeas, Pref: Pref, Tref: Tref, _time: r._time,}))
```

@davidgsIoT

Do the Calculating

```
CO2corr = final
	|> map(fn: (r) => ({"_time": r._time, "CO2Adjust": r.CO2meas * (((r.Tmeas + 273.15) * r.Pref) / (r.Pmeas * r.Tref)), "_value": r.CO2meas * (((r.Tmeas + 273.15) * r.Pref) / (r.Pmeas * r.Tref))}))
```

ppm CO2 corrected = ppm CO2 measured * ((Tmeasured*Pref) / (pmeasured*Tref))

@davidgsIoT

Demo Time!

Questions?