INTERNET OF THINGS DEVICES AND STREAMS

SPARQL-TO-SQL ON

EUGENE SIOW
THANASSIS TIROPANIS
WENDY HALL
Trekking up 913m
Mt. Rokko
Current State of the Internet of Things

“The Internet of Things is currently beset by product silos.”
W3C Web of Things Interest Group

- Sensor Data from Multiple Sources
- Product & Data Silos
- Dependency on the Cloud
- Limited Interoperability of Apps
LINKED DATA ON LIGHTWEIGHT COMPUTERS
INTRODUCING LINKED DATA FOR INTEROPERABILITY

URI AND ONTOLOGIES
ESTABLISH COMMON DATA STRUCTURES & REFERENCES

ENABLES RICH METADATA
WHAT, WHERE, WHEN, HOW OF DATA

INTEGRATES WITH LINKED OPEN DATA
CYBER, SOCIAL, PHYSICAL LOD ON WEB

Linked data on distributed lightweight computers

Tiered levels of applications between the "Ground" and "Cloud" for meteorological data:

- Weather data
- Data stream
- Meteorological station on lightweight computer
- Environmental sensors
- State inclement weather planning application
- Irrigation application
- Query on rainfall
- Broadcast queries
- SRBench: National weather and disaster monitoring
CHALLENGES

PERFORMANCE
STORES DON’T SCALE & PERFORM WELL ON WEB YET

RESOURCE CONSTRAINED DEVICES
~500MHZ CPU, 512MB RAM, SD CARD

HISTORICAL AND STREAMING DATA
NEED STORAGE AND QUERY EFFICIENCY ON BOTH
THE SHAPE OF IOT TIME-SERIES DATA

FLAT

{ timestamp : 1467673132,
  temperature : 32.0,
  wind_speed : 10.5,
  pressure : 1016
}

COMPlex

{ timestamp : 1467673132,
  temperature : { 
    max : 22.0,
    min : 15.0,
    current : 17.0,
    error : {
      percentage : 5.0
    }
  }
}

WIDTH

6+

5

4

2,3

1

99.5% FLAT SCHEMATA

0.5% COMPLEX SCHEMATA

20K UNIQUE DEVICES
dweet.io

1

2

3

4
EFFICIENT QUERIES WITH TIME-SERIES DATA

THING produces TEMPERATURE OBS
produces

TEMPERATURE OBS has value 13.0
has value CELCIUS

TEMPERATURE OBS has value 2016-01-01 06:00:00

THING produces HUMIDITY OBS
produces
time 2016-01-01 06:00:00

HUMIDITY OBS has value 93.0
has value PERCENT

HUMIDITY OBS has value 2016-01-01 06:00:00

THING produces WIND SPEED OBS
produces

WIND SPEED OBS has value 10.5
has value MPH

WIND SPEED OBS has value 2016-01-01 06:00:00

THING has value LOCATION
<table>
<thead>
<tr>
<th>Sensor Metadata</th>
<th>Observation Metadata</th>
<th>Observation Data</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>THING</strong></td>
<td><strong>TEMPERATURE OBS</strong></td>
<td><strong>has value</strong></td>
</tr>
<tr>
<td><strong>produces</strong></td>
<td></td>
<td>13.0</td>
</tr>
<tr>
<td><strong>THING</strong></td>
<td><strong>HUMIDITY OBS</strong></td>
<td><strong>has value</strong></td>
</tr>
<tr>
<td><strong>produces</strong></td>
<td></td>
<td>93.0</td>
</tr>
<tr>
<td><strong>THING</strong></td>
<td><strong>WIND SPEED OBS</strong></td>
<td><strong>has value</strong></td>
</tr>
<tr>
<td><strong>produces</strong></td>
<td></td>
<td>10.5</td>
</tr>
<tr>
<td><strong>THING</strong></td>
<td><strong>LOCATION</strong></td>
<td></td>
</tr>
<tr>
<td><strong>located</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>TEMPERATURE OBS</strong></td>
<td></td>
<td><strong>time</strong></td>
</tr>
<tr>
<td><strong>unit</strong></td>
<td></td>
<td>2016-01-01 06:00:00</td>
</tr>
<tr>
<td><strong>TEMPERATURE OBS</strong></td>
<td></td>
<td><strong>unit</strong></td>
</tr>
<tr>
<td><strong>HUMIDITY OBS</strong></td>
<td></td>
<td>CELCIUS</td>
</tr>
<tr>
<td><strong>TIME SERIES DATA</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>HUMIDITY OBS</strong></td>
<td></td>
<td><strong>time</strong></td>
</tr>
<tr>
<td><strong>unit</strong></td>
<td></td>
<td>2016-01-01 06:00:00</td>
</tr>
<tr>
<td><strong>HUMIDITY OBS</strong></td>
<td></td>
<td><strong>unit</strong></td>
</tr>
<tr>
<td><strong>WIND SPEED OBS</strong></td>
<td></td>
<td>PERCENT</td>
</tr>
<tr>
<td><strong>TIME SERIES DATA</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>WIND SPEED OBS</strong></td>
<td></td>
<td><strong>time</strong></td>
</tr>
<tr>
<td><strong>unit</strong></td>
<td></td>
<td>2016-01-01 06:00:00</td>
</tr>
<tr>
<td><strong>WIND SPEED OBS</strong></td>
<td></td>
<td><strong>unit</strong></td>
</tr>
<tr>
<td><strong>THING</strong></td>
<td><strong>LOCATION</strong></td>
<td></td>
</tr>
<tr>
<td><strong>produces</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>THING</strong></td>
<td><strong>TEMPERATURE OBS</strong></td>
<td><strong>has value</strong></td>
</tr>
<tr>
<td><strong>produces</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>THING</strong></td>
<td><strong>HUMIDITY OBS</strong></td>
<td><strong>has value</strong></td>
</tr>
<tr>
<td><strong>produces</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>THING</strong></td>
<td><strong>WIND SPEED OBS</strong></td>
<td><strong>has value</strong></td>
</tr>
<tr>
<td><strong>produces</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>THING</strong></td>
<td><strong>LOCATION</strong></td>
<td><strong>located</strong></td>
</tr>
<tr>
<td><strong>produces</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>THING</strong></td>
<td><strong>TEMPERATURE OBS</strong></td>
<td><strong>has value</strong></td>
</tr>
<tr>
<td><strong>produces</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>THING</strong></td>
<td><strong>HUMIDITY OBS</strong></td>
<td><strong>has value</strong></td>
</tr>
<tr>
<td><strong>produces</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>THING</strong></td>
<td><strong>WIND SPEED OBS</strong></td>
<td><strong>has value</strong></td>
</tr>
<tr>
<td><strong>produces</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
OUR APPROACH

EFFICIENT QUERIES WITH TIME-SERIES DATA

THING located LOCATION

TEMPERATURE OBS produces unit CELCIUS

HUMIDITY OBS unit PERCENT

WIND SPEED OBS MPH

<table>
<thead>
<tr>
<th>TIME</th>
<th>TEMPERATURE</th>
<th>HUMIDITY</th>
<th>WIND SPEED</th>
</tr>
</thead>
<tbody>
<tr>
<td>2016-01-01 06:00:00</td>
<td>13.0</td>
<td>93.0</td>
<td>10.5</td>
</tr>
</tbody>
</table>
Efficient queries with time-series data

Our approach

Observation metadata

Sensor metadata

Thing located location

Temperature obs

Humidity obs unit

Wind speed obs

CELIUS

PERCENT

MPH

Observation data

Time | Temperature | Humidity | Wind speed
---|-------------|----------|-------------
2016-01-01 06:00:00 | 13.0 | 93.0 | 10.5

Temperature: CELCIUS

Humidity: PERCENT

Wind speed: MPH

Location: somewhere

Produces: something
DESIGNING OUR ENGINE

THING

produces

TEMPERATURE OBS

HUMIDITY OBS

WIND SPEED OBS

located

LOCATION

has value

CELCIUS

unit

PERCENT

MPH

has value

TABLE 1.TEMPERATURE

TABLE 1.HUMIDITY

TABLE 1.WINDSPEED

TABLE 1

TIME | TEMPERATURE | HUMIDITY | WINDSPEED
--- | --- | --- | ---
2016-01-01 06:00:00 | 13.0 | 93.0 | 10.5
DESIGNING OUR ENGINE

THING located in LOCATION

TEMPERATURE OBS produces CELCIUS

HUMIDITY OBS has unit PERCENT

WIND SPEED OBS has unit MPH

TABLE 1.TEMPERATURE has value

TABLE 1.HUMIDITY has value

TABLE 1.WINDSPEED has value

TABLE 1

<table>
<thead>
<tr>
<th>TIME</th>
<th>TEMPERATURE</th>
<th>HUMIDITY</th>
<th>WINDSPEED</th>
</tr>
</thead>
<tbody>
<tr>
<td>2016-01-01 06:00:00</td>
<td>13.0</td>
<td>93.0</td>
<td>10.5</td>
</tr>
</tbody>
</table>
DESIGNING OUR ENGINE

SELECT MAX(?TEMPERATURE) { ?OBS a TEMPERATURE OBS; ?OBS has value ?TEMPERATURE; ?OBS has unit ?UOM. }

\[
\pi(\gamma(\max(?TEMPERATURE)))
\]

?OBS a TEMPERATURE OBS; ?OBS has value ?TEMPERATURE; ?OBS has unit ?UOM.

BGP

2016-01-01 06:00:00 13.0 93.0
DESIGNING OUR ENGINE

SELECT MAX(?TEMPERATURE) 
{ 
?OBS a TEMPERATURE OBS
?OBS has value ?TEMPERATURE
?OBS has unit ?UOM

}\n
\[ \pi \{(\text{MAX} (?\text{TEMPERATURE}))\} \]

?OBS a TEMPERATURE OBS
?OBS has value ?TEMPERATURE
?OBS has unit ?UOM

\[ \text{TEMPERATURE OBS} \]
\text{CELCIUS} has value

\[ \text{TABLE 1.TEMPERATURE} \]
\text{TEMPERATURE}
13.0

\[ \text{MAX} ( \text{?TEMPERATURE} ) \]
DESIGNING OUR ENGINE

\[
\text{SELECT } \max(\text{TEMPERATURE}) \{ \\
\quad \text{OBS a TEMPERATURE OBS} \\
\quad \text{OBS has value TEMPERATURE} \\
\quad \text{OBS has unit UOM} \\
\}\]

\[
\pi(\max(\text{TEMPERATURE}))
\]

\[
\text{MAX(TEMPERATURE)}
\]

\[
\text{NODE_TEMP} \quad \text{TABLE1.TEMPERATURE} \quad \text{CELCIUS}
\]

\[
\text{OBS} \quad \text{TEMPERATURE} \quad \text{UOM}
\]
SELECT MAX(?TEMPERATURE) {
  ?OBS a TEMPERATURE OBS
  ?OBS has value ?TEMPERATURE
  ?OBS has unit ?UOM
}

SQL SELECT MAX(TEMPERATURE) FROM TABLE1
LOW VISIBILITY STATIONS

SELECT ?SENSOR {
  { ?OBS a VISIBILITY OBS
  ?OBS has value ?VISIBILITY
  ?OBS has unit ?UOM
  FILTER(?VISIBILITY,<10 )
  UNION
  { ?OBS a RAINFALL OBS
  ?OBS has value ?RAINFALL
  ?OBS has unit ?UOM
  FILTER(?RAINFALL,>30 )
  }
  }
}

UNION

FILTER(?SNOWFALL, TRUE)

{ ?OBS a SNOWFALL OBS
  ?OBS has value ?SNOWFALL
  ?OBS has unit ?UOM
}

UNION

FILTER(?VISIBILITY,<10 )

{ ?OBS a VISIBILITY OBS
  ?OBS has value ?VISIBILITY
  ?OBS has unit ?UOM
}

UNION

FILTER(?RAINFALL,>30 )

{ ?OBS a RAINFALL OBS
  ?OBS has value ?RAINFALL
  ?OBS has unit ?UOM
}

UNION

FILTER(?SNOWFALL, TRUE)

{ ?OBS a SNOWFALL OBS
  ?OBS has value ?SNOWFALL
  ?OBS has unit ?UOM
}

BGP
LOW VISIBILITY STATIONS EXAMPLE

SELECT ?SENSOR {
  { ?OBS a VISIBILITY OBS
    ?OBS has value ?VISIBILITY
    ?OBS has unit ?UOM
    FILTER( ?VISIBILITY, <10 ) UNION
    UNION
    { ?OBS a RAINFALL OBS
      ?OBS has value ?RAINFALL
      ?OBS has unit ?UOM
      FILTER( ?RAINFALL, >30 ) } }
}

UNION

FILTER( SNOWFALL, TRUE )
FROM TABLE1
UNION
FILTER( VISIBILITY, <10 )
FROM TABLE1
FILTER( RAINFALL, >30 )
FROM TABLE1
LOW VISIBILITY STATIONS

```
SELECT ?SENSOR {
  { ?OBS a VISIBILITY OBS
  ?OBS has value ?VISIBILITY
  ?OBS has unit ?UOM
  FILTER(?VISIBILITY < 10) UNION
  { ?OBS a RAINFALL OBS
  ?OBS has value ?RAINFALL
  ?OBS has unit ?UOM
  FILTER(?RAINFALL > 30 )
  
  UNION
  { ?OBS a SNOWFALL OBS
  ?OBS has value ?SNOWFALL
  ?OBS has unit ?UOM
  FILTER(?SNOWFALL, TRUE)
  
  FROM TABLE1 WHERE RAINFALL > 30
  SELECT SENSOR FROM TABLE1 WHERE VISIBILITY < 10
  UNION
  SELECT SENSOR FROM TABLE1 WHERE RAINFALL > 30
  }
}
```
LOW VISIBILITY STATIONS EXAMPLE

SPARQL

```sparql
SELECT ?SENSOR {
  {
    ?OBS a VISIBILITY OBS
    ?OBS has value ?VISIBILITY
    FILTER(?VISIBILITY < 10)
  } UNION 
  {
    ?OBS a RAINFALL OBS
    ?OBS has value ?RAINFALL
    FILTER(?RAINFALL > 30)
  }
}
```

SQL

```sql
SELECT SENSOR FROM TABLE1
WHERE RAINFALL > 30
UNION
SELECT SENSOR FROM TABLE1
WHERE SNOWFALL = TRUE
UNION
SELECT SENSOR FROM TABLE1
WHERE VISIBILITY < 10
```
S2SML MAPPING

WRITTEN IN RDF
Reflects the RDF it represents directly

R2RML COMPATIBLE
Can be translated to and from R2RML

SUPPORTS COLLAPSING NODES
Efficient queries with IoT time-series data

IRI IRI\textsubscript{MAP} BNODE FNODE

IRI

IRI IRI\textsubscript{MAP} BNODE FNODE

LITERAL LITERAL\textsubscript{MAP}
S2SML MAPPING

FAUX NODE

HUMIDITY OBS

hasValue

LITERAL MAP

TABLE1.COL1

<http://knoesis.wright.edu/ssw/obs/{readings.uuid}>

if (HUMIDITY OBS is projected):
    if(readings.uuid !exists):
        create_col(readings.uuid)
    return IRImap(readings.uuid)
RDF STREAM PROCESSING

1. **Register Query**
   - **Stream Window**
   - SPARQL query specifying stream window size

2. **Translate Query**
   - **sparql2stream**
   - Same engine and mappings but translates to EPL instead of SQL

3. **Stream Data**
   - **Stream Sockets**
   - Supports multiple platforms and streams with ZeroMQ

4. **Receive Push Results**
   - **Real-time analytics**
   - PIOTRe & Smart Home Freeboard demo
   - github.com/eugenesiow/iotwo
SELECT MAX(?TEMPERATURE) FROM NAMED STREAM OBSSTREAM RANGE 1H {
  ?OBS a OBS;
  ?OBS has value ?TEMPERATURE;
  ?OBS has unit ?UOM
}

EPL Event Processing Language
SELECT MAX(TEMPERATURE) FROM WEATHER.WIN:TIME(1 HOUR)
EVALUATION WITH BENCHMARKS

**SRBENCH**
- ~20,000 Stations
- 100 – 300k triples
- Wind, Rainfall, etc.
- 10 SRBench Queries

**SMART HOME BENCH**
- 3 months, 1 home
- ~30k triples
- Motion, energy, environment
- 4 Analytics Queries


GraphDB (OWLIM)
Ontop
Our Approach (S2S)
Morph
TDB

Raspberry Pi 2 Model B+
1GB RAM, 900MHz Quad Core ARM Cortex A7, Class 10 SD Cards
STORAGE SIZE

OUR APPROACH (S2S)

TDB

GRAPHDB

30K SMART HOME

100K NEVADA BLIZZARD

300K HURRICANE IKE
Get the rainfall observed in a particular hour from all stations

Q01 with an optional clause on unit of measure
03 Detect if a hurricane has been observed

Join between wind observation and temperature observation subtrees time-consuming in low resource environment (Raspberry Pi)

04 Get the average wind speed at the stations where the air temperature is >32

05 Detect if a station is observing a blizzard
06 Get the stations with extremely low visibility

07 Detect stations that are recently broken

08 Get the daily minimal and maximal air temperature observed by the sensor at a given location
Get the daily average wind force and direction observed by the sensor at a given location.

Get the locations where a heavy snowfall has been observed.

Join between wind force and wind direction observation subtrees is time-consuming in low resource environment (Raspberry Pi).

Our Approach (s2s) is shown to be faster on all queries in the Distributed Meteorological System with SRBench.
SMART HOME RESULTS

01. Temperature aggregated by hour on a specified day

02. Minimum and maximum temperature each day for a particular month
Involves motion and meter data (much larger set), with space-time aggregations and joins between motion and meter tables/subgraphs.

Diagnose unattended appliances consuming energy with no motion in room

Energy Usage Per Room By Day

Involves meter data (larger set), with space-time aggregations.

Our Approach (s2s) is shown, once again, to be faster on all queries for Smart Home Analytics.
STREAM PROCESSING SCALABILITY

**VELOCITY**
>99% <1ms latency increasing from 1 to 1000 rows/ms

**VOLUME**
33.5 million rows, projected ~2.5 billion triples!
PERSONAL IOT REPOSITORY

“It's a long road, it's a long and narrow way. If I can't work up to you, you'll surely have to work down to me someday.”

_Narrow Way_ by Bob Dylan