Distributed Operator Placement for IoT Data Analytics Across Edge and Cloud Resources

Eduard Gibert Renart, Alexandre da Silva Veith, Daniel Balouek-Thomert, Marcos D. de Assunção, Laurent Lefèvre and Manish Parashar

alexandre.veith@ens-lyon.fr

17th May 2019

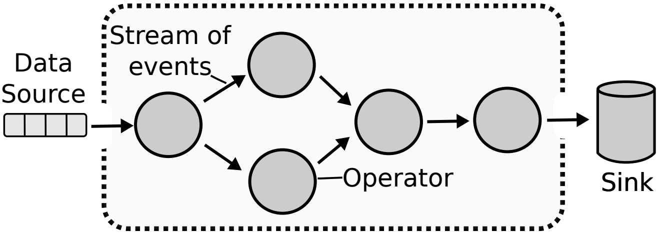
Agenda

IoT Applications

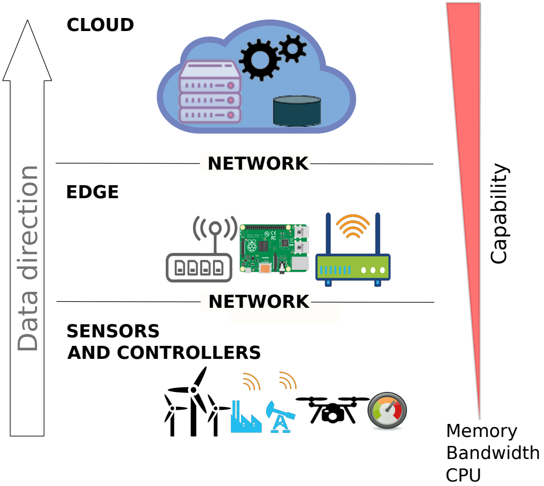
- Generate massive amounts of data
- Data Stream Processing Engines consider IoT applications as dataflows

IoT Applications

- Generate massive amounts of data
- Data Stream Processing Engines consider IoT applications as dataflows

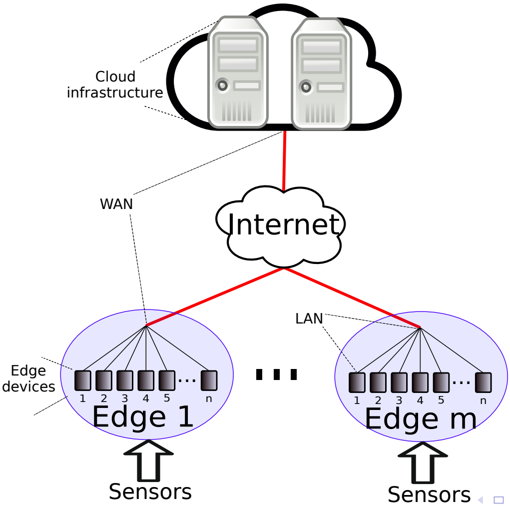


Infrastructure for Deploying Data Stream Analytics

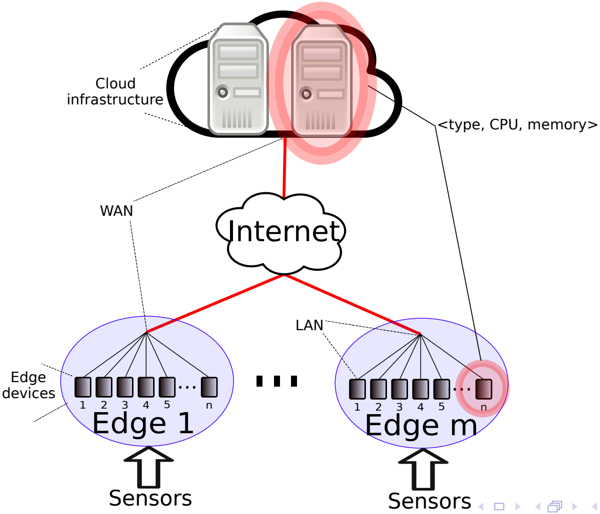


How to **split** an IoT application **dynamically** across edge and cloud resources by considering QoS metrics and respecting environment constraints?

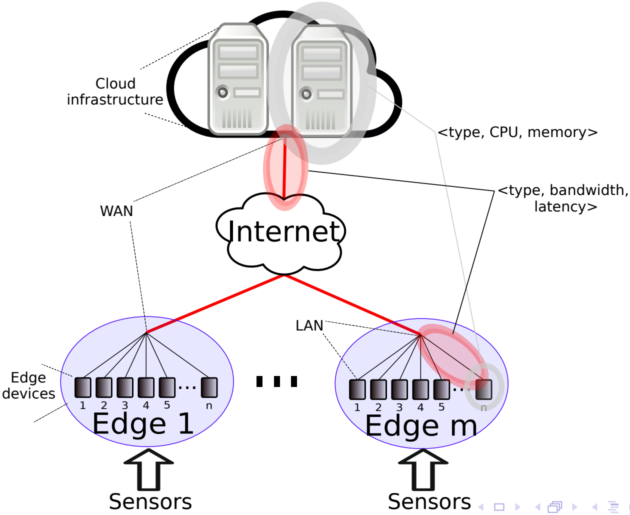
Infrastructure Topology



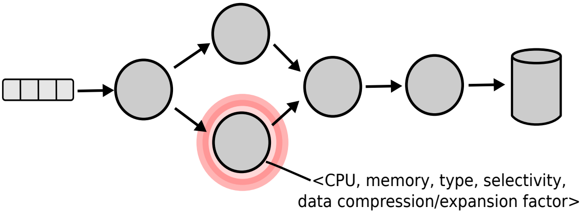
Infrastructure Topology



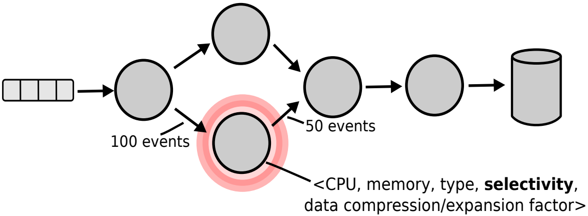
Infrastructure Topology

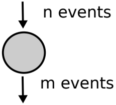


Application Topology



Application Topology

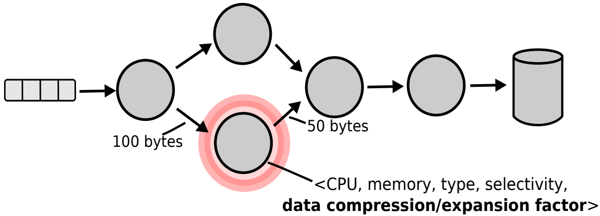


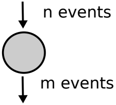


Selectivity

The ratio of number of
input events to output events

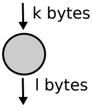
Application Topology





Selectivity

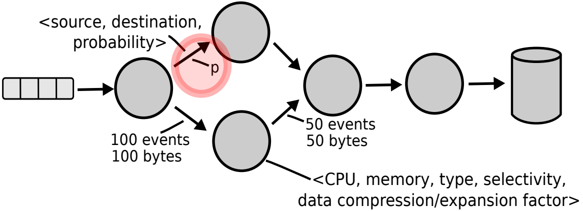
The ratio of number of input events to output events

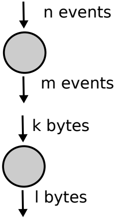


Data compression/ expansion factor

The ratio of the size of input events to the size of output events

Application Topology





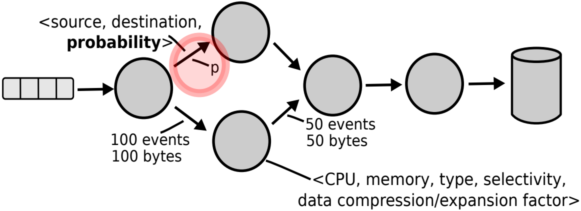
Selectivity

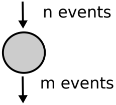
The ratio of number of
input events to output events

Data compression/ expansion factor

The ratio of the size of input events
to the size of output events

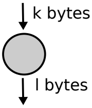
Application Topology





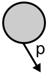
Selectivity

The ratio of number of
input events to output events



Data compression/ expansion factor

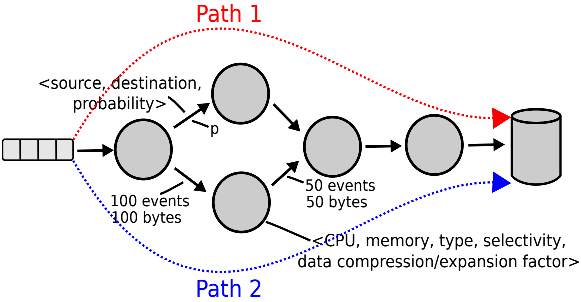
The ratio of the size of input events
to the size of output events

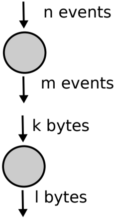


Probability

Probability of an event crosses the stream

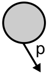
Application Topology





Selectivity

The ratio of number of
input events to output events

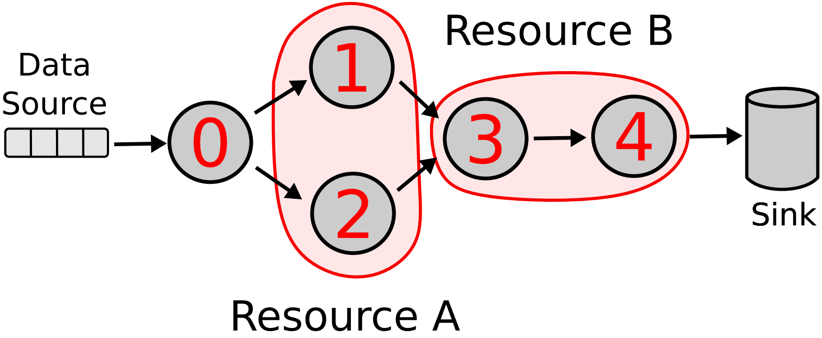


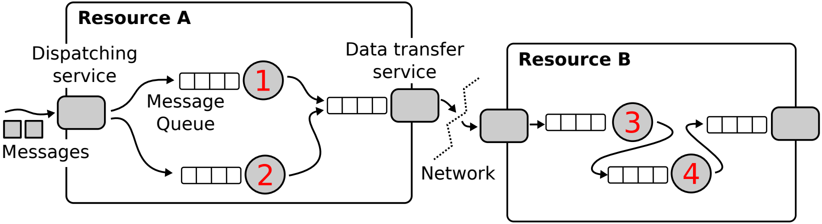
Probability

Probability of an event crosses the stream

Data compression/ expansion factor

The ratio of the size of input events
to the size of output events


Data Source
0
1
2
3
4
Resource B
Resource A
Sink



- Queues for **Computation**(operator) and **Communication**(data transfer service)
- Model is based on **Queueing Theory** - **M/M/1**
- The queues of operators and data transfer service provide:
  - The number of messages stored in the resource
  - The departure rate in each service
  - A manner to estimate the bandwidth, CPU and memory limitations

QoS metrics

Response time

end-to-end latency from the time events are generated to the time they reach the sinks

QoS metrics

Response time
end-to-end latency from the time events are generated to the time they reach the sinks

WAN traffic
accumulates the sizes of the events that cross the WAN network.

QoS metrics

Response time
end-to-end latency from the time events are generated to the time they reach the sinks

WAN traffic
accumulates the sizes of the events that cross the WAN network.

Monetary cost
calculates the monetary cost using the number of messages that reached the cloud from the edge and vice-versa (*i.e.* IoT Hub Based)

Problem Statement

A single aggregate cost using Simple Additive Weighting method [1] and covering the three metrics:

$w_0 \times$ **Response time** $+ w_1 \times$ **WAN traffic** $+ w_2 \times$ **Monetary cost**

where w corresponds to weight assigned to the metric
and the value of each metric is normalized

Each metric is estimated per path

[1] K.P. Yoon, P.K. Yoon, C.L. Hwang, SAGE., and inc Sage Publications. Multiple Attribute Decision Making: An Introduction. Multiple Attribute Decision Making: An Introduction. SAGE Publications, 1995.

Problem Statement

A single aggregate cost using Simple Additive Weighting method [1] and covering the three metrics:

$$w_0 \times \textbf{Response time} + w_1 \times \textbf{WAN traffic} + w_2 \times \textbf{Monetary cost}$$

where w corresponds to weight assigned to the metric
and the value of each metric is normalized

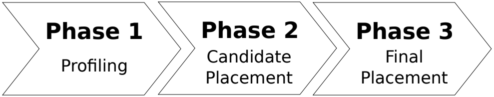
Each metric is estimated per path

Objective:

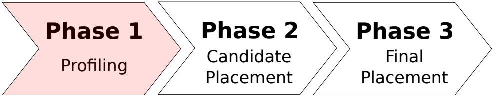
$$min \sum_{p_i \in \text{paths}} \text{Aggregate cost of } p_i$$

[1] K.P. Yoon, P.K. Yoon, C.L. Hwang, SAGE., and inc Sage Publications. Multiple Attribute Decision Making: An Introduction. Multiple Attribute Decision Making: An Introduction. SAGE Publications, 1995.

Aggregate Cost Aware Strategy



Aggregate Cost Aware Strategy



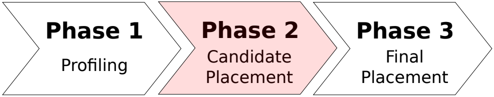
R-Pulsar [2], an edge-based runtime for stream processing

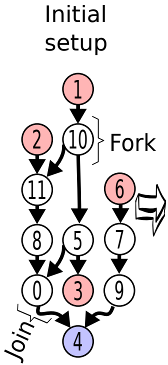
- "All-in-one" lightweight framework for efficient and real-time data-driven stream processing
- Content- and location-based programming abstraction for **user-defined composition and optimization templates**
- A rule-based programming abstraction for **programmable reactive behaviors**

R-Pulsar establishes the selectivity, data compression/expansion factor, as well as, the CPU and memory requirements

[2] http://rpulsar.org

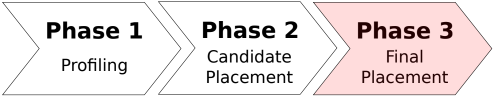
Aggregate Cost Aware Strategy





- The user-predefined locations of sinks and sources are used to identify patterns in the dataflow
- Split points (i.e., forks and joins)
- The destination of messages is used to create two sets of candidate placements (cloud and edge)
- Operators are organized into an operator deployment sequence

Aggregate Cost Aware Strategy

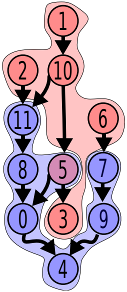


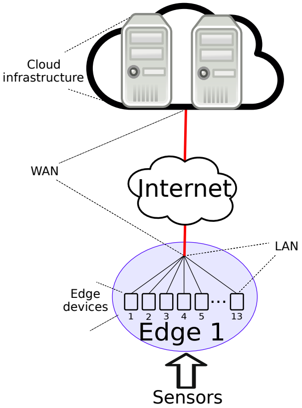
The Aggregate Cost is calculated as a rate

For each operator in the operator deployment sequence

- Computation and communication estimation for all resources
- Evaluate memory, CPU, and bandwidth constraints
- Resource with lowest aggregate cost (response time + WAN traffic + monetary cost) is elected to host the operator

Final placement





- Real testbed platform;
- Edge: One site with 5 Raspberry PIs 3 (4x ARM Cortex A53 1.2GHz, 1GB of RAM and 10/100 Ethernet) and 8 Raspberry PIs 2 (4x ARM Cortex A7 900MHz, 1GB of RAM and 100 Ethernet);
- Cloud: Chameleon Cloud [a] with 5 instances of type m1.medium (2 CPU and 4GB RAM);
- LAN [b]: Latency 0.523ms and bandwidth equal to 15Mbps;
- WAN [b]: Latency: 66.75ms and bandwidth equal to 87Mbps.

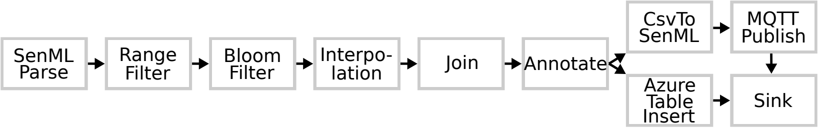
[a] Chameleon Cloud. https://chameleoncloud.org/

[b] Hu, W., Gao, Y., Ha, K., Wang, J., Amos, B., Chen, Z., Pillai, P., Satyanarayanan, M.: *Quantifying the impact of edge computing on mobile applications.* APSys '16, ACM, New York, NY, USA (2016)

Comparison

- **Cloud** which deploys all operators in the cloud, apart from operators provided in the initial placement;
- **Taneja et. al. (LB)** [2] which iterates a vector containing the operators, gets the middle host of the computational vector and evaluates the constraints;
- **Random** which is the average of 15 different dataflow deployments between the edge and the cloud resources.

[2] Taneja, M., Davy, A. "*Resource aware placement of iot application modules in fog-cloud computing paradigm*". IM - May 2017

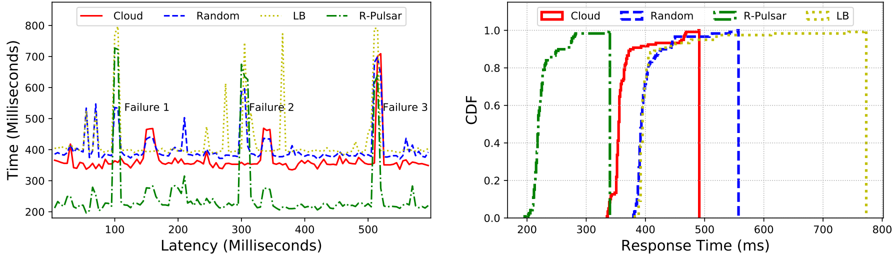


The experiments were conducted using Sense Your City [3] which consists of :

- Data is transmitted at each minute;
- Sensors are spread in 7 cities across 3 continents;
- An average of 12 sensor per city;
- Data content includes metadata on the sensor ID, geolocation, and five timestamped observations (outdoor temperature, humidity, ambient light, dust, and air quality).

[3] http://map.datacanvas.org

End-to-end tuple latency

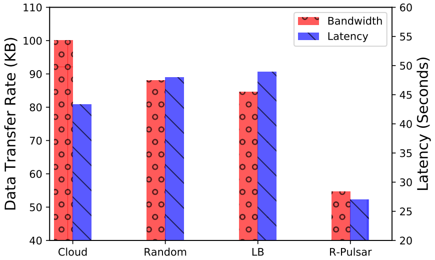


First failure makes 38% of edge devices unavailable (100ms). **Second failure** affects 62% of edge devices unavailable (300ms). And **third failure** affects 50% of cloud instances (505ms).

31% faster than the traditional cloud setup, and **38%** faster than Random and LB approaches

Without failures 38% better than cloud and 44% of reduction compared to Random and LB

Data transfer rate

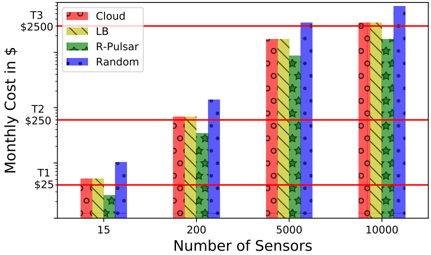


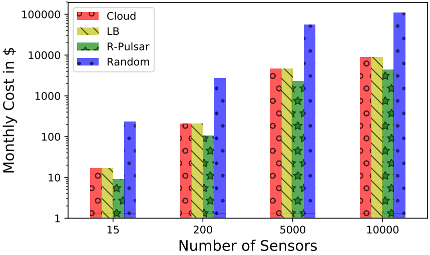
35% and 45% less data transferred compared to LB, and cloud and random, respectively

	Cloud	LB	Random
Latency	38%	45%	44%
Data Transfer	46%	36%	38%

Second experiment considers a weight of 50% for the data transfer and 50% for the latency

Messaging cost





Microsoft Azure IoT Hub [4] and **Amazon IoT Core** [5]

- **90%** of savings when considering Microsoft Azure IoT Hub
- **50%** (cloud and LB) and **97%** (random) of savings for AWS IoT Core

[4] https://azure.microsoft.com/en-us/pricing/details/iot-hub/
[5] https://aws.amazon.com/iot-core/pricing/

Conclusions and Future Work

Summary

- A model and the IoT operator placement problem formalization
- A strategy to improve the multi criteria objective
- An implementation atop of R-Pulsar software stack
- A performance evaluation using a real testbed platform

Conclusions

- The key behaviors (forks and joins) of the dataflows directed us to our strategy
- Our strategy using the dataflow aspects allow us to be 45% better in response time

Future Work

- An extension for including energy consumption as a metric

Questions?