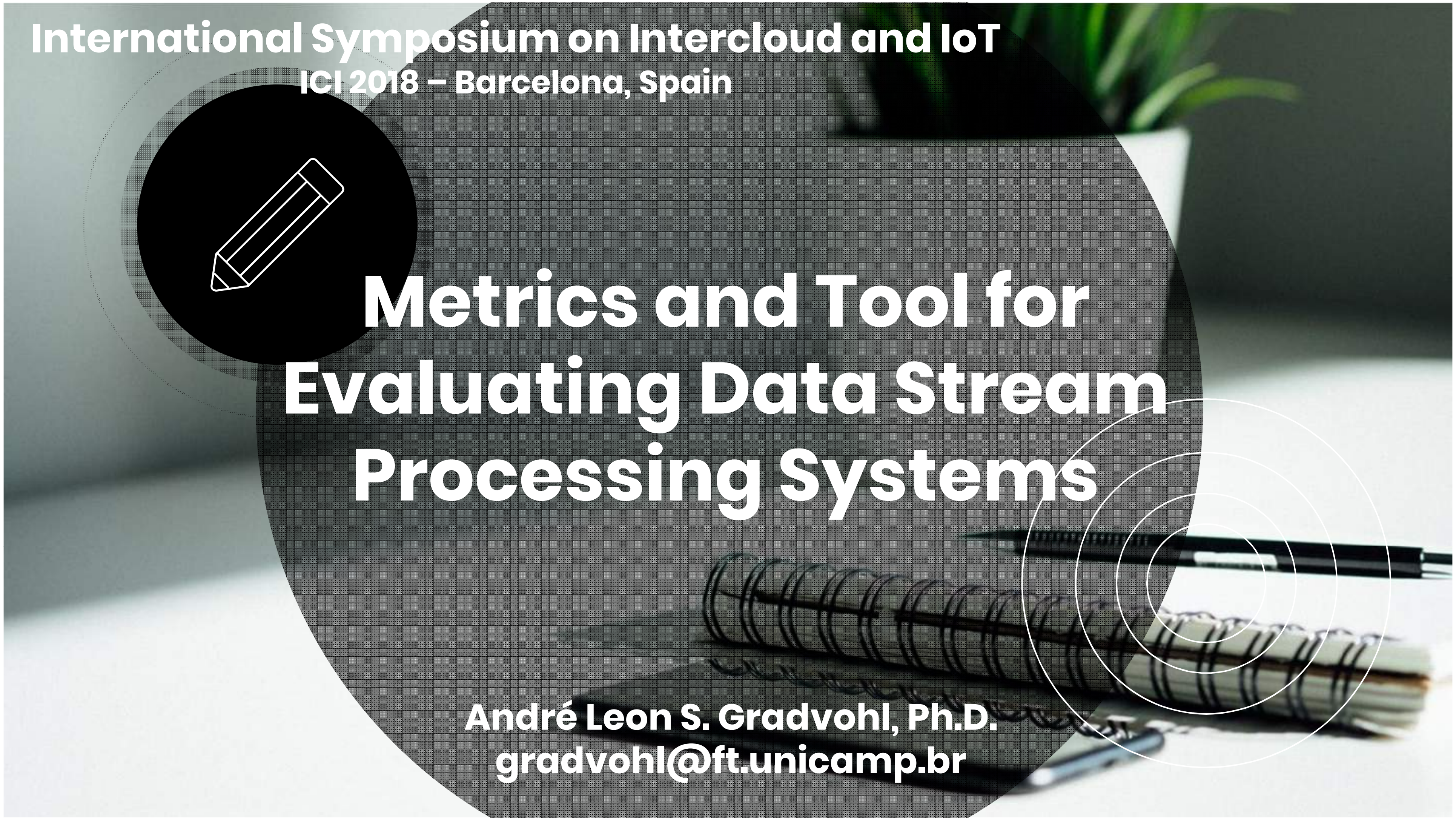


International Symposium on Intercloud and IoT

ICI 2018 – Barcelona, Spain

Metrics and Tool for Evaluating Data Stream Processing Systems

André Leon S. Gradvohl, Ph.D.
gradvohl@ft.unicamp.br

License and Contact Details

- This work is licensed under the Creative Commons Attribution 4.0 International License.
- To view a copy of this license, visit https://choosealicense.com/licenses/cc-by-4.0.

BY

- Contact the author at

gradvohl@computer.org

Agenda

- Introduction
- Metrics for DaSP systems
- Proposed System: B2-4DaSP
- Experiment with B2-4DaSP
- Final Remarks

Introduction

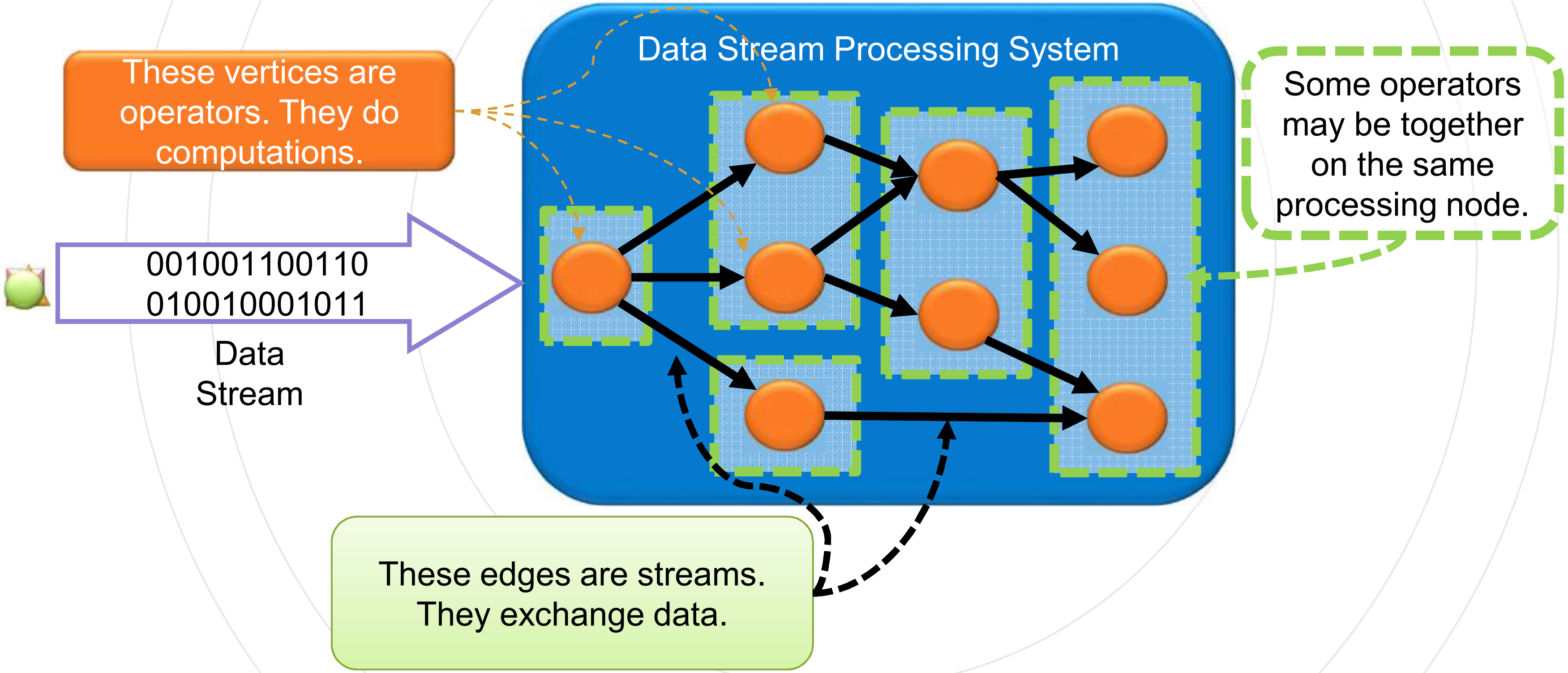
- Data Stream Processing (DaSP) systems are software specialized in processing sequences of data.
- Data streams have high-throughput, low latency, e.g. Twitter posts, sensors data, network cards etc.
- DaSP systems see the input data as transient continuous streams that must be processed "on the fly", with critical requirements on throughput, latency, and memory occupancy.

Introduction

- The choice of the most appropriate system to handle data streams is a challenge today. After all, we cannot foresee the characteristics of the streams that a Data Stream Processing (DaSP) system will handle.
- The main question is which features a data stream processing system must have to meet the solution requirements that we want to implement.
- In this sense, this work contributes to discussions about the metrics for DaSP systems performance analysis and proposes a benchmark analysis tool for DaSP systems considering the proposed metrics.

General architecture for DaSP systems

General architecture for DaSP systems

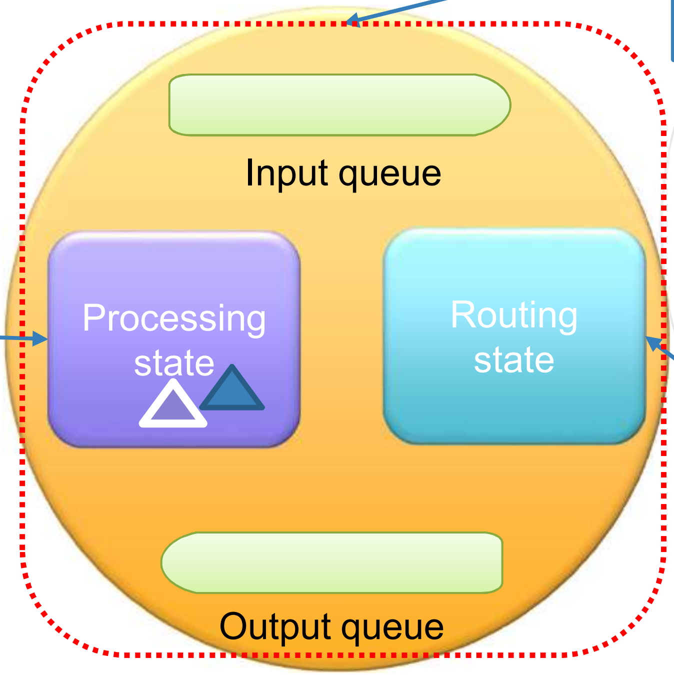


These vertices are operators. They do computations.

Some operators may be together on the same processing node.

These edges are streams. They exchange data.

Operator architecture



All these components are the state of an operator.

Keeps the internal data structures (objects) of the operator

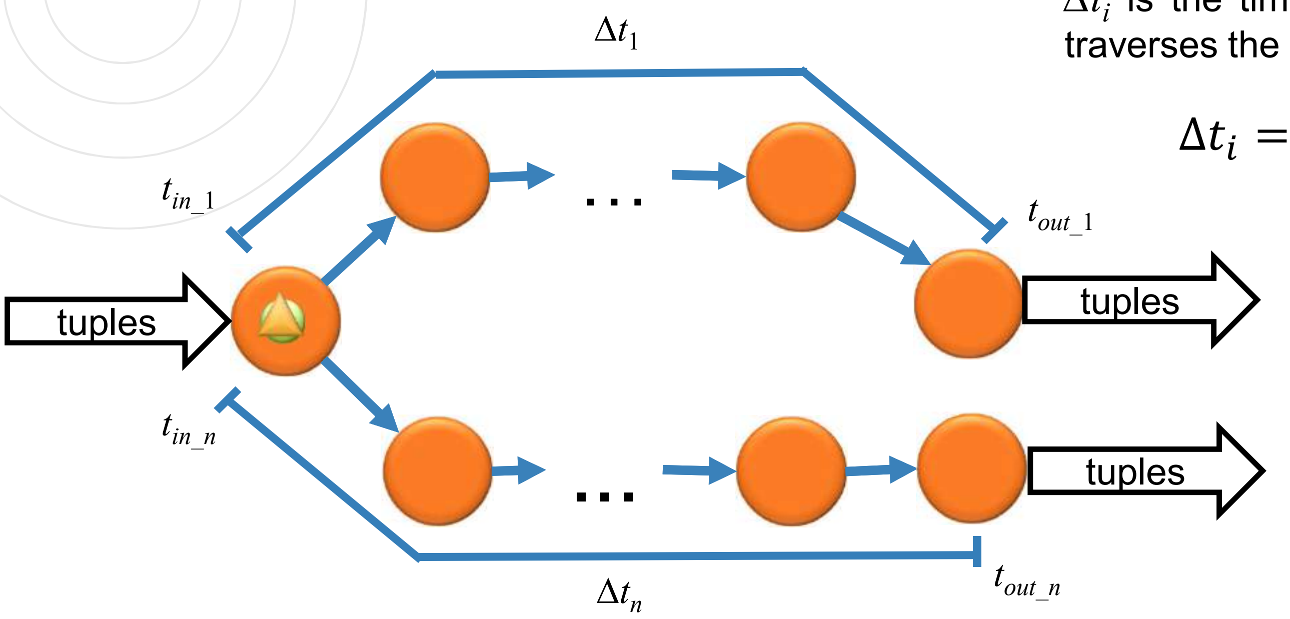
Keeps the information about routes (connections) to other operators

Metrics for DaSP systems

Metrics for DaSP Systems

Metric	How to measure it
Throughput	Measured by the number of processed events (tuples) given a period.
Memory Consumption	Measured by the size of the window, i.e. the number of events considered in a query, a node can handle or the time considered in a window within the nodes of the topology.
Latency	
- System Latency	the time the system takes to process each a tuple or the difference between the time of output a tuple and the time of input of the same tuple.
- Information Latency	the time a system takes output the result of a query, which, in turn, needs many events to process.
- Maximum peak latency	the time delay experienced by the system during a peak load in tuples stream. We measure This metric by calculating the time to process a query when the system is overloaded by tuples.
- Post-peak latency ratio	the time delay experienced by the system after a peak load in tuples.

Metrics for DaSP Systems: System Latency



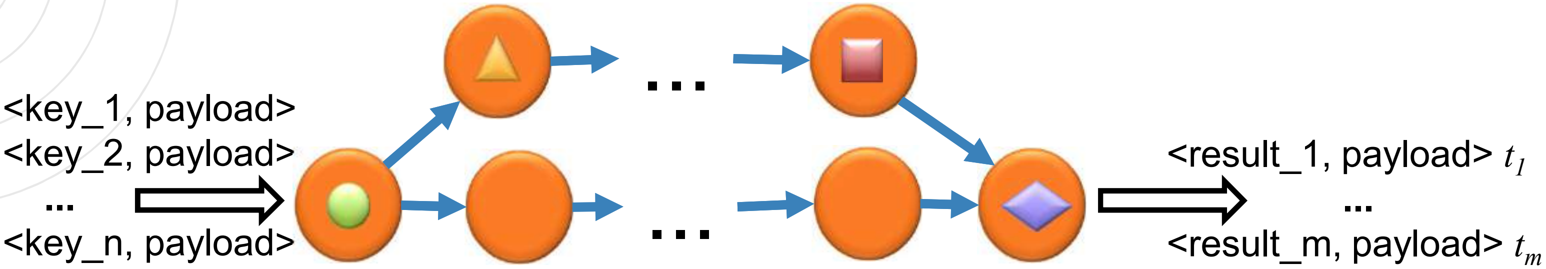
Δt_i is the time that a tuple traverses the i path.

$$\Delta t_i = t_{out\_i} - t_{in\_i}$$

$$sl = \frac{\sum_{i=1}^{n} \Delta t_i}{n}$$

sl is the average time of all paths in the topology.

Metrics for DaSP Systems: Information Latency

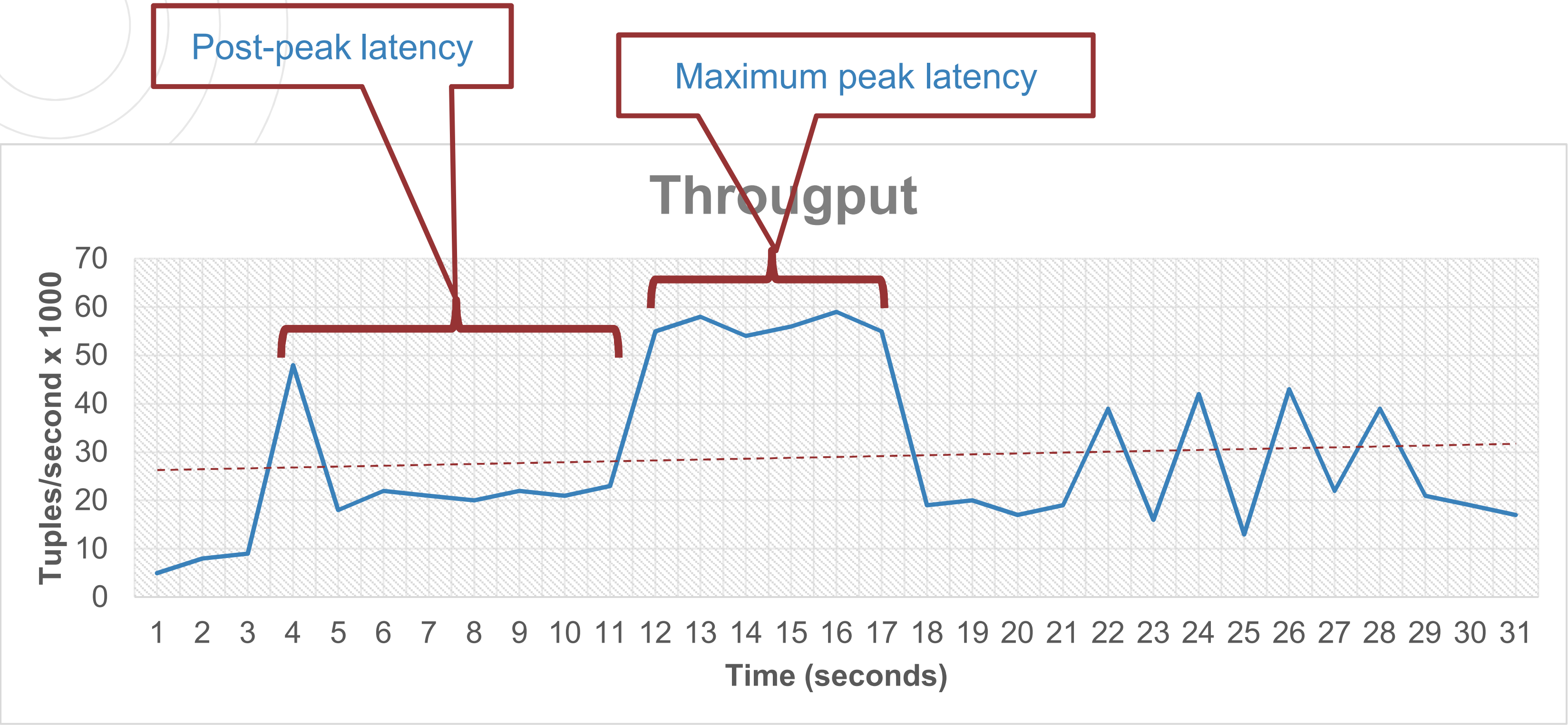


t_i is the time to produce a tuple with a result.

$$il = \frac{\sum_{i=1}^{m-1}(t_{i+1} - t_i)}{m}$$

il is the average time of the results that the system produces.

Metrics for DaSP Systems: Memory, Latency, Throughput



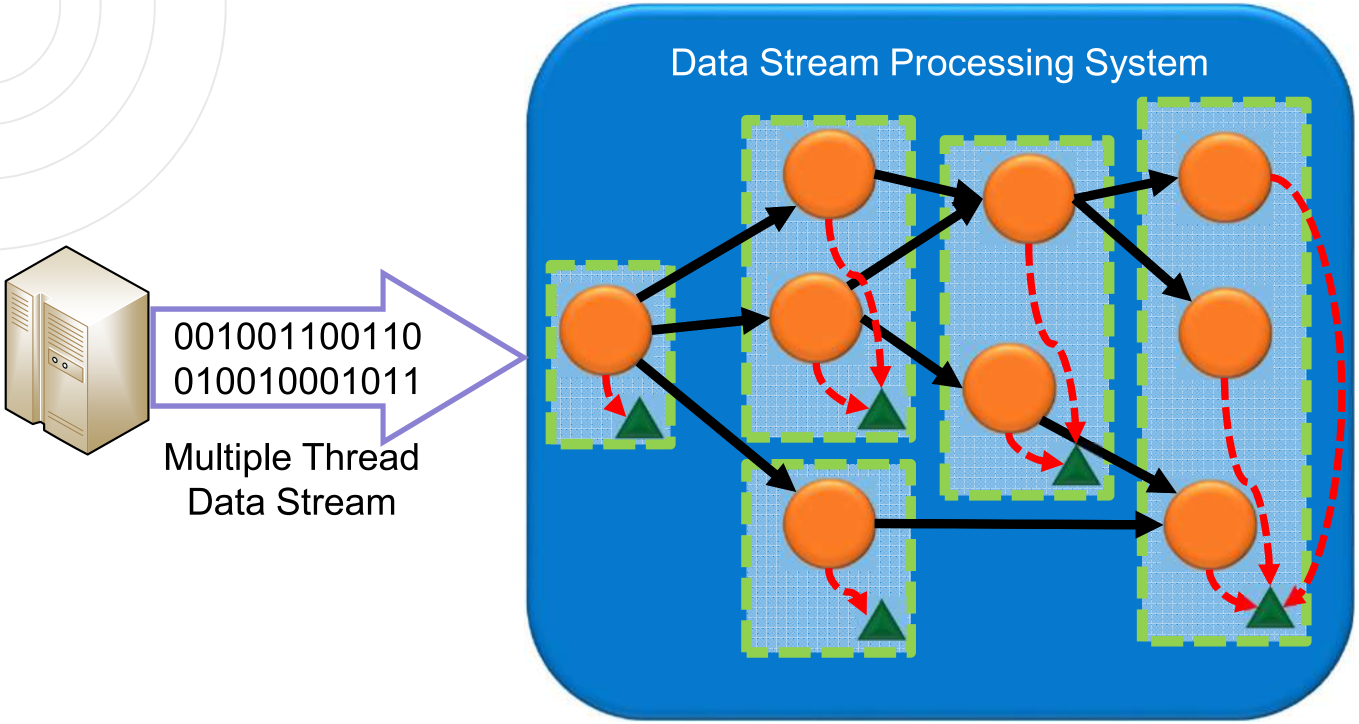
Post-peak latency

Maximum peak latency

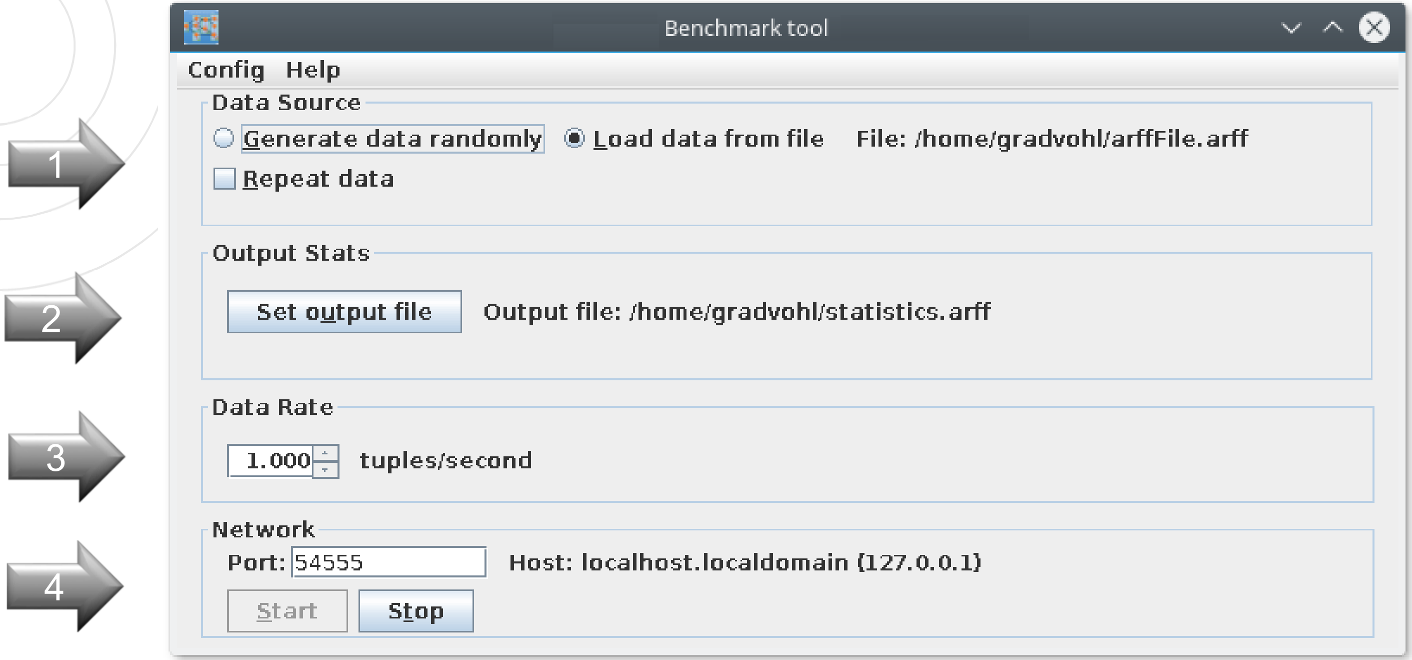
Proposed System B2-4DaSP

How to measure metrics for DaSP systems:

Benchmark tool for DaSP systems (B2-4DaSP)



Benchmark tool for DaSP systems (B2-4DaSP): Control Panel



Data Source

○ Generate data randomly ● Load data from file File: /home/gradvohl/arffFile.arff

□ Repeat data

Output Stats

Set output file Output file: /home/gradvohl/statistics.arff

Data Rate

1.000 tuples/second

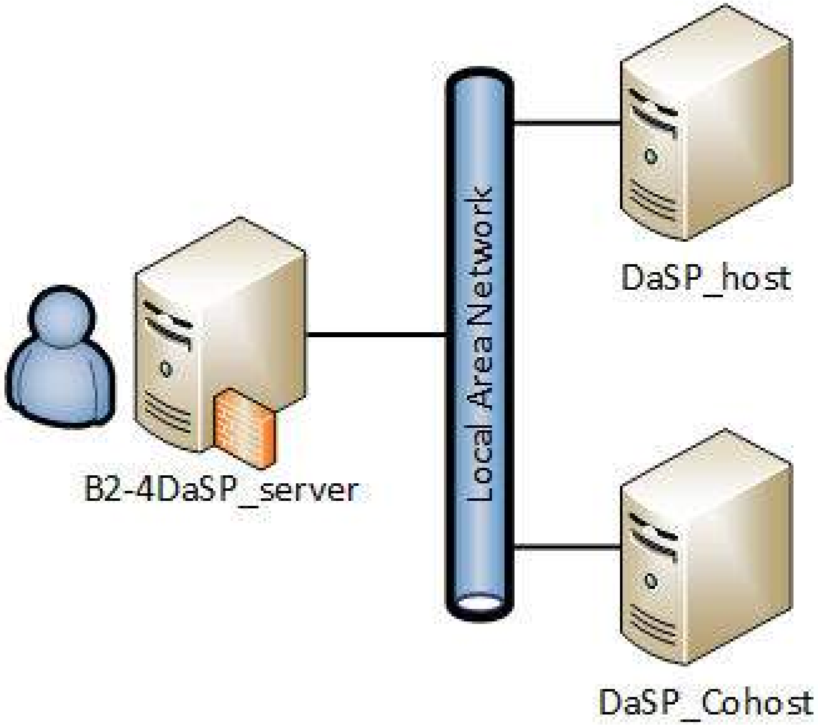
Network

Port: 54555 Host: localhost.localdomain (127.0.0.1)

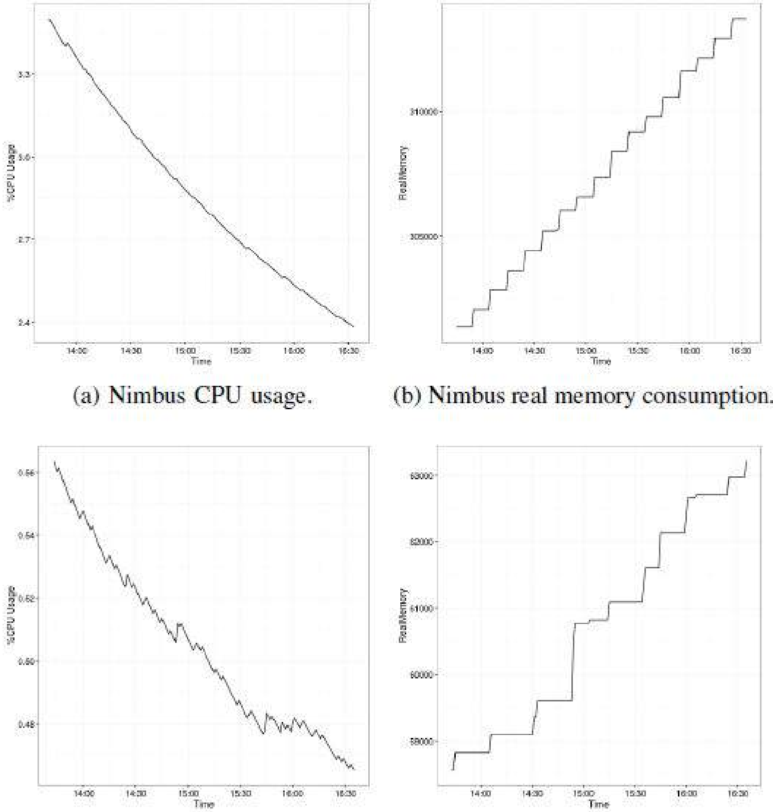
Start Stop

Experiment with B2–4DaSP

Experimental setup



Testing Apache Storm with B2-4DaSP – Management Components



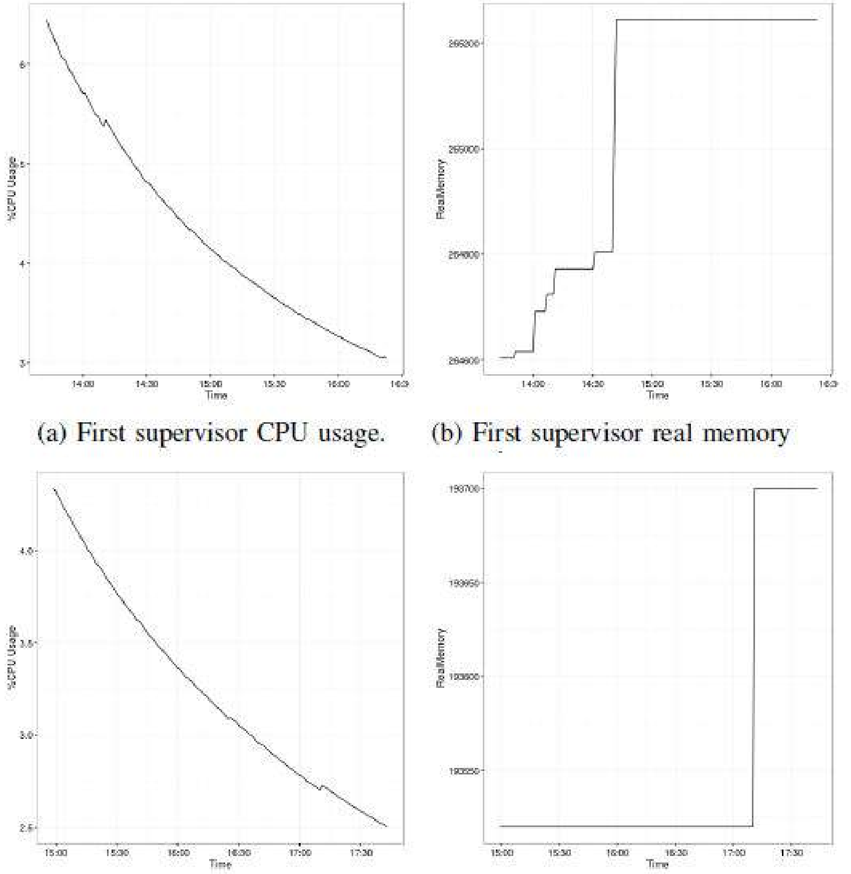
(a) Nimbus CPU usage.

(b) Nimbus real memory consumption.

(c) Zookeeper CPU usage.

(d) Zookeeper real memory consumption.

Testing Apache Storm with B2-4DaSP – Operators



(a) First supervisor CPU usage.

(b) First supervisor real memory

(c) Second supervisor CPU usage.

(d) Second supervisor real memory

Final remarks

Final remarks

- We observed that there are some aspects that require improvements for the construction of more robust DaSP systems.
  - For instance, among the DaSP systems analyzed, none of them implements mechanisms to adapt the computational resources demand to the data stream throughput.
  - Each component of the system may request more resources when the throughput increases.
  - However, when the throughput declines, resources remain allocated, even when they are not necessary.

Final remarks

- Another issue that we observed throughout this study was the presence of cluster managers in some DaSP systems.
  - They are important for the management of the operators within the topology, as they allow the fast verification of operators' status and, depending on the situation, operator's replacement.
- However, the cluster managers are a single point of failures (SPOF) and, if we want to increase the fault tolerance of the DaSP system as a whole, we need to implement mechanisms that allow its monitoring and self-regulation.
- Concerning the metrics approached in this work, we found that the memory and CPU consumptions bring much information about the behavior of a DaSP system depending on the streams it receives.
  - This information can be useful in defining what resources the system can request and what capacity it can support. To collect this information we built the B2-4DaSP system, which proved very useful for this task.

More info about the author

IEEE Collabratec
https://ieee-collabratec.ieee.org/app/p/AndreGradvohl

LinkedIn/in/andregradvohl

https://orcid.org/0000-0002-6520-9740

/agradvohl

gradvohl@ft.unicamp.br

This presentation is available in PDF with the following DOI:
https://doi.org/10.5281/zenodo.1292767