Online and Scalable Data Validation in Advanced Metering Infrastructures

Vincenzo Gulisano, Magnus Almgren and Marina Papatriantafilou

Chalmers University
of technology

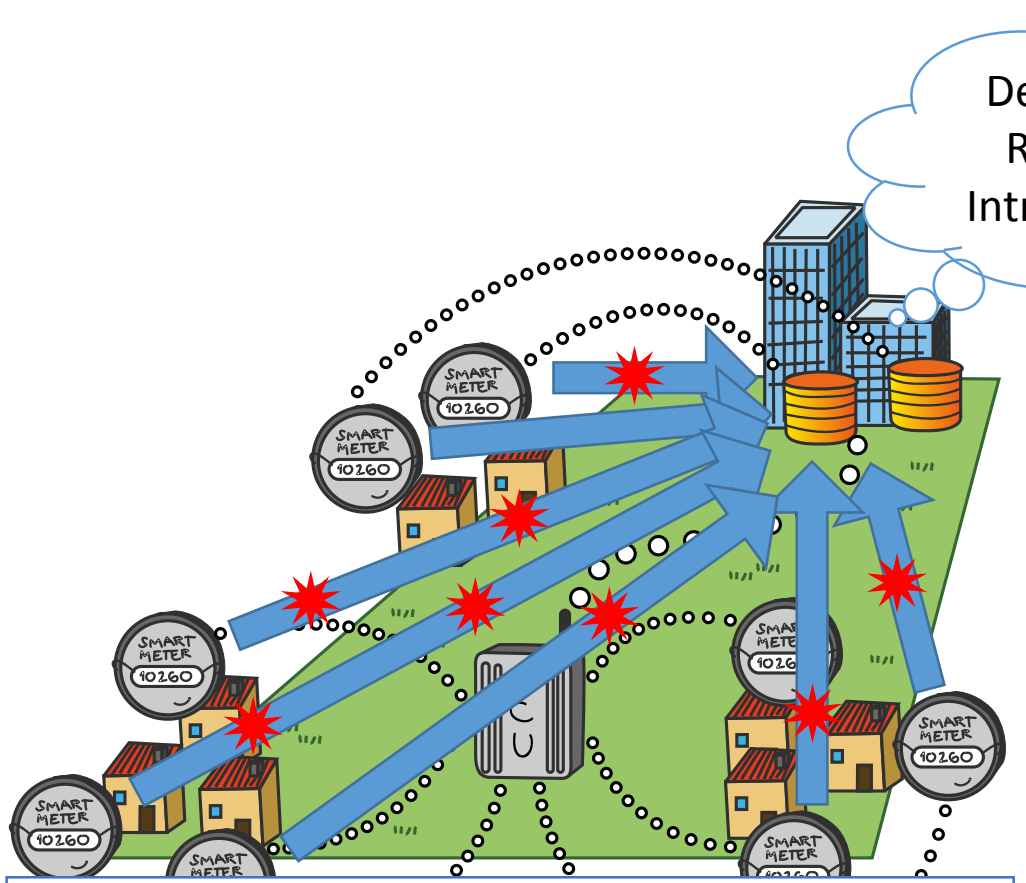
Agenda

1. Problem statement
2. Preliminaries – Data Streaming
3. Streaming-based Data Validation
4. Conclusions

Agenda

1. Problem statement
2. Preliminaries – Data Streaming
3. Streaming-based Data Validation
4. Conclusions

Online and Scalable Data Validation in Advanced Metering Infrastructures



Noisy and Lossy data: bad-calibrated / faulty devices, lossy communication, malicious users, …

How can we validate data given that…

- There is a large volume of *continuous* data demanding for distributed and parallel analysis
- Validation rules depend on installation-specific features such as brands, devices, protocols, …
  - System experts should define installation-specific validation rules?

Agenda

1. Problem statement
2. Preliminaries – Data Streaming
3. Streaming-based Data Validation
4. Conclusions

Data Streaming - Motivation

- Financial applications, sensor networks monitoring, … require
  - Continuous processing of data streams
  - Real Time fashion

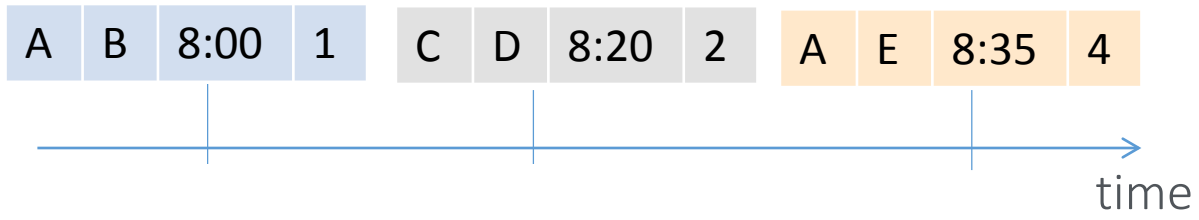
- Store and process is not feasible
  - Financial markets: process millions of messages/second, take fast decisions (< 100 microseconds)

- Data Streaming:
  - In memory
  - Bounded resources
  - Efficient one-pass analysis

Data Streaming - System Model

- Data Stream: unbounded sequence of tuples
  - Example: consumption readings

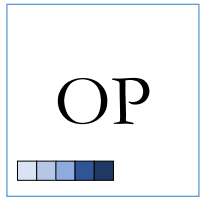
Field	Field
House	text
Device	text
Time (secs)	int
Consumption (kWh)	double



Data Streaming - System Model

- Operators:

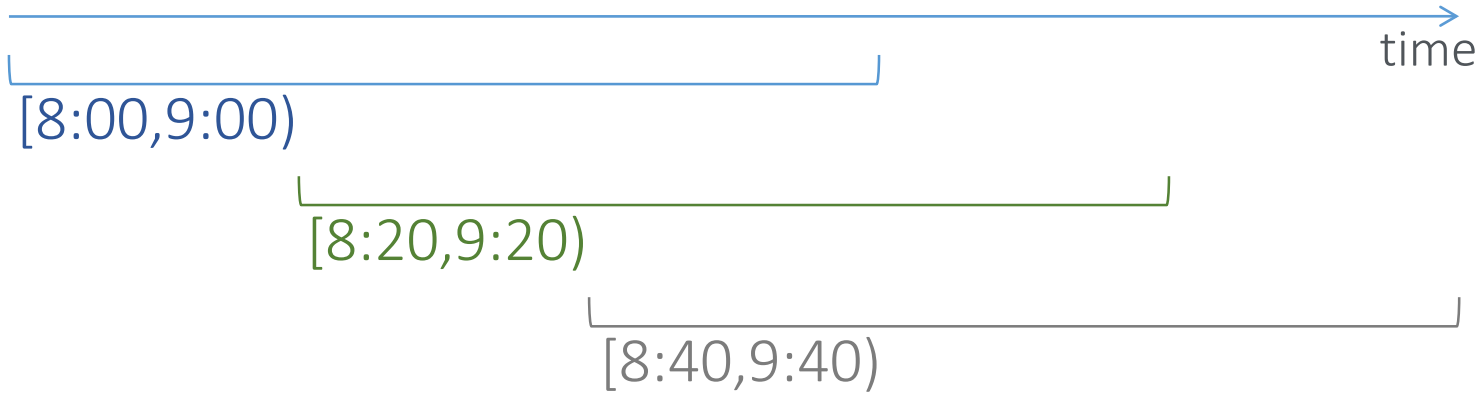
Stateless
1 input tuple
1 output tuple



Stateful
1+ input tuple(s)
1 output tuple

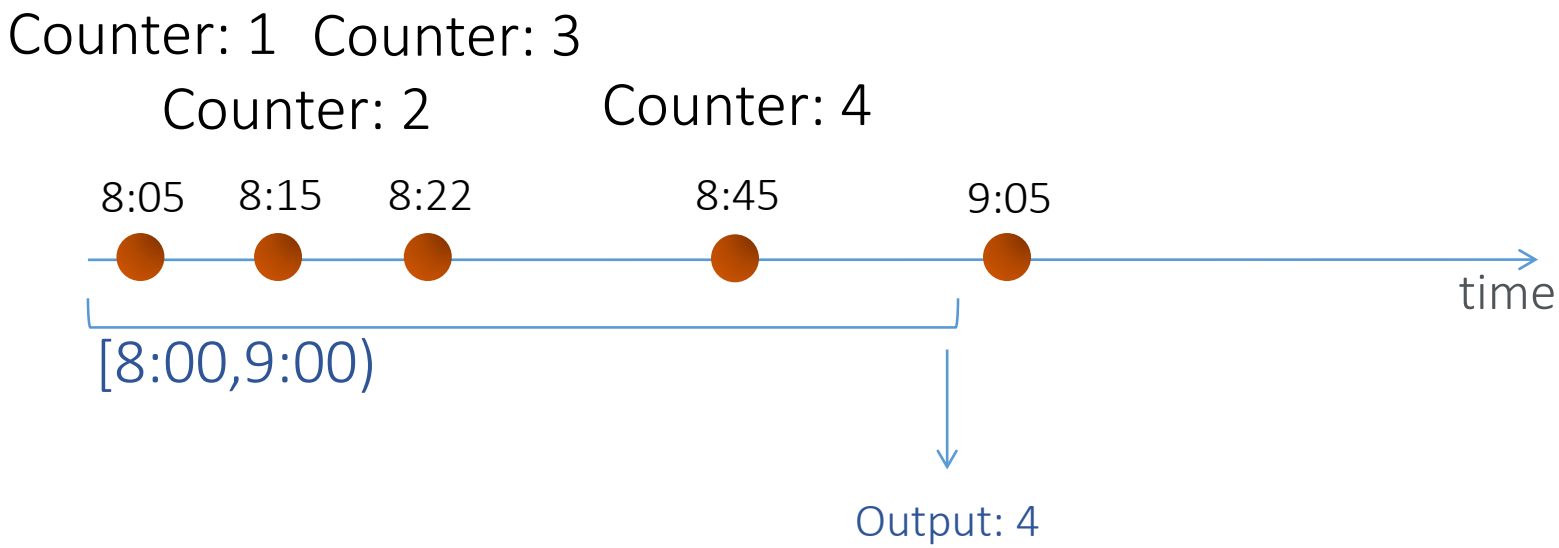
Data Streaming - System Model

- Infinite sequence of tuples / bounded memory
  → windows
- Example: 1 hour windows



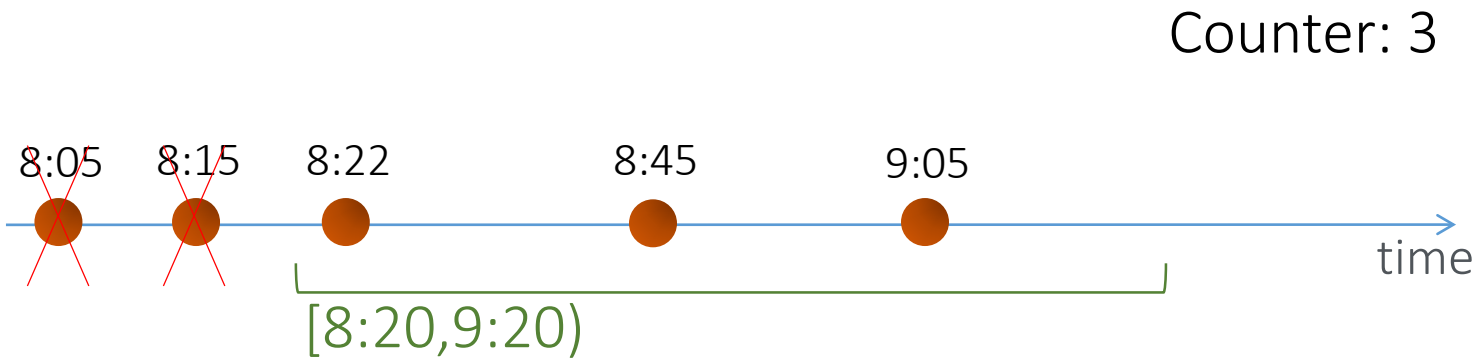
Data Streaming - System Model

- Infinite sequence of tuples / bounded memory
  → windows
- Example: 1 hour windows



Data Streaming - System Model

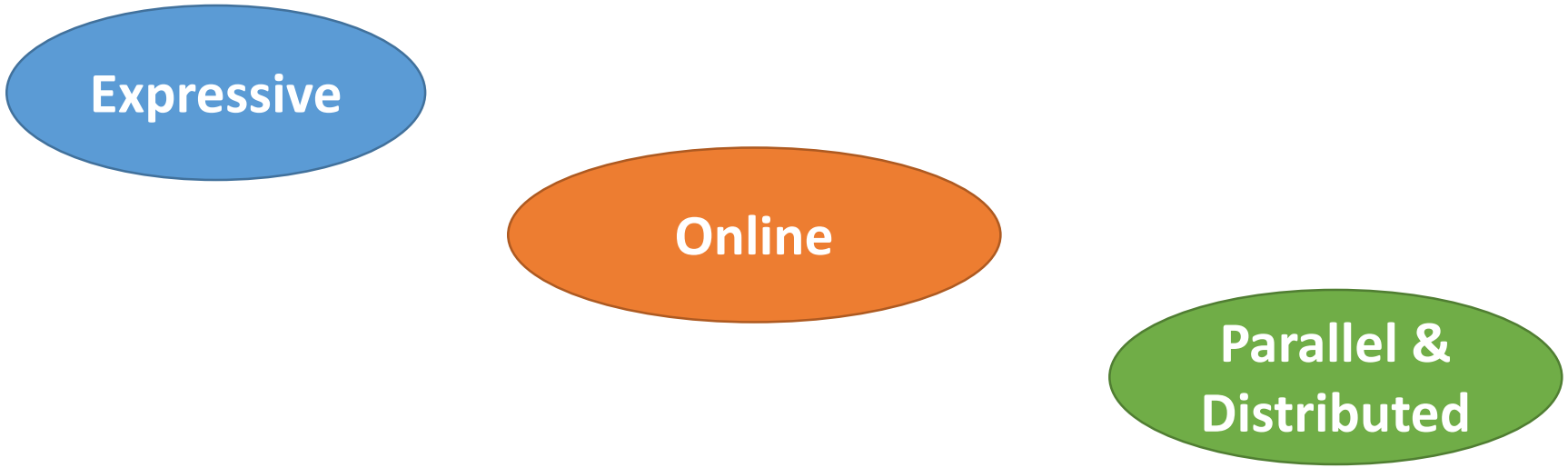
- Infinite sequence of tuples / bounded memory
  → windows
- Example: 1 hour windows



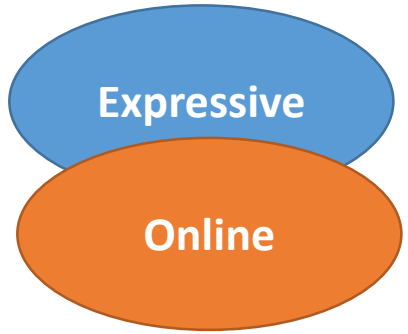
Agenda

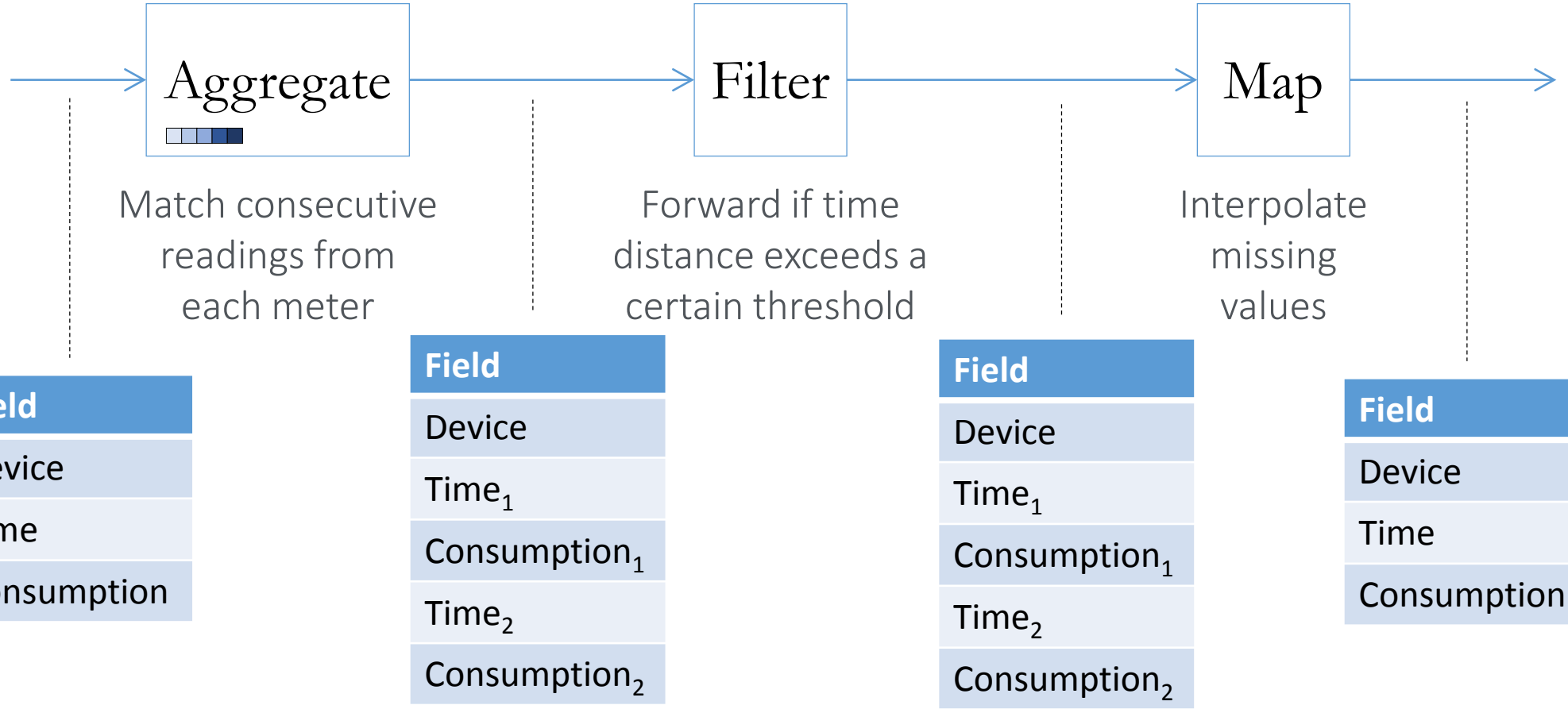
1. Problem statement
2. Preliminaries – Data Streaming
3. Streaming-based Data Validation
4. Conclusions

Why Streaming-based data Validation?

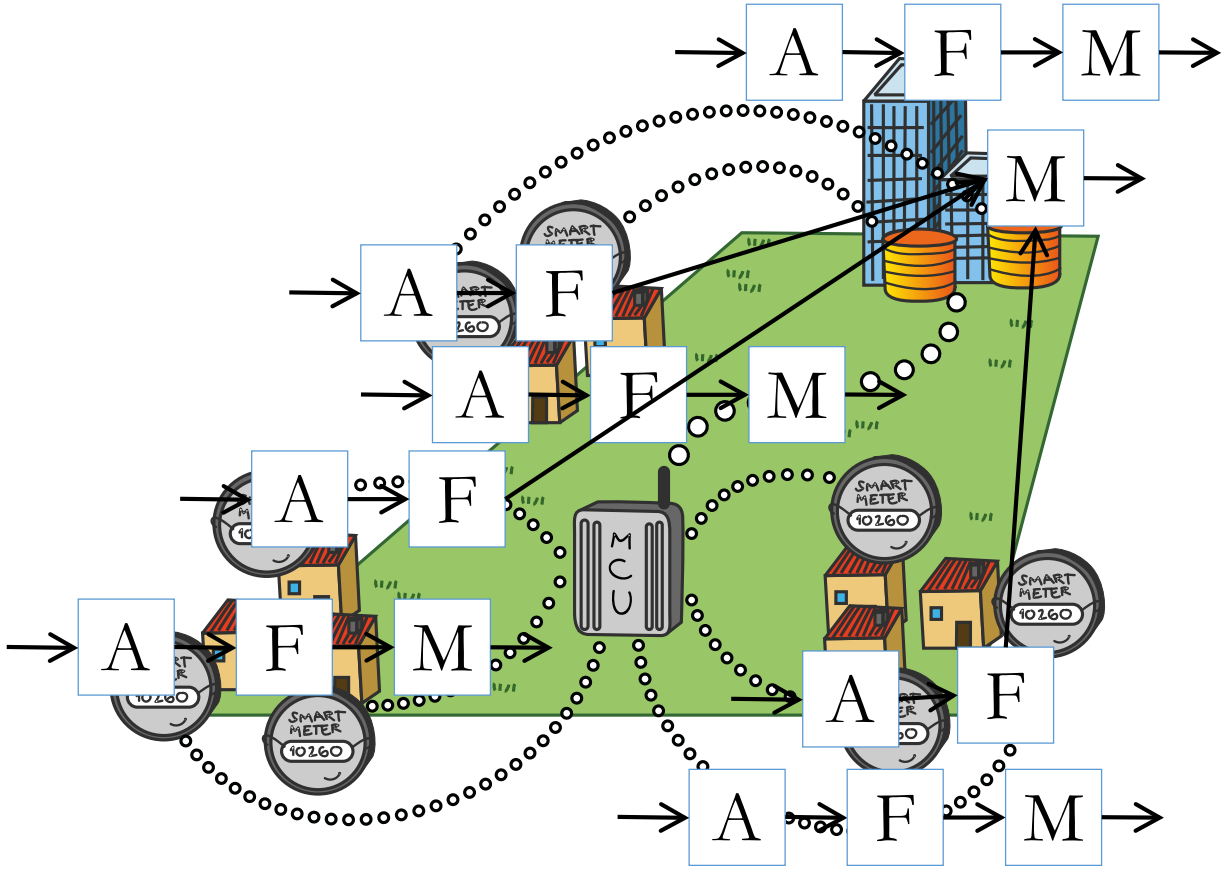


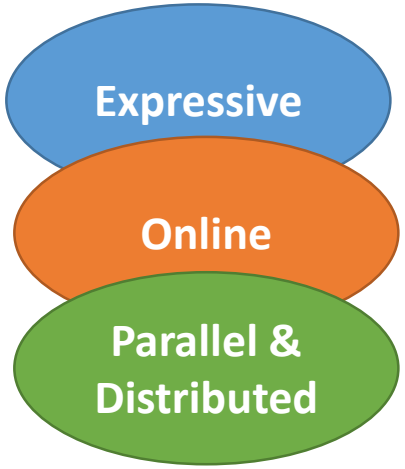
Sample Streaming-based Data Validation: Interpolate missing consumption values





Sample Streaming-based Data Validation: Interpolate missing consumption values





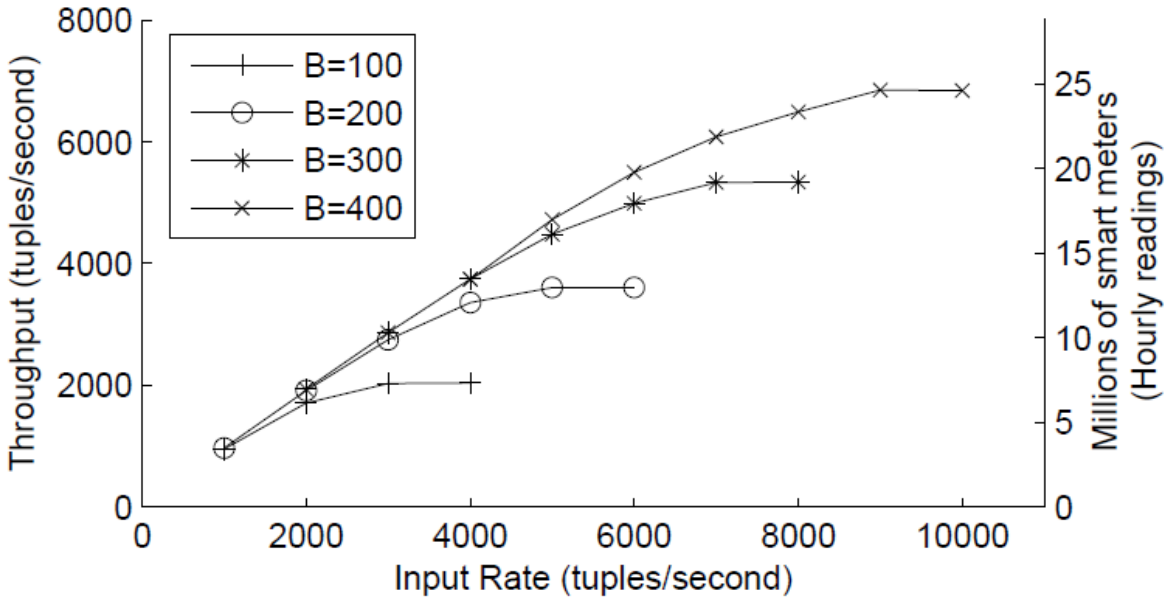
Evaluation - Setup

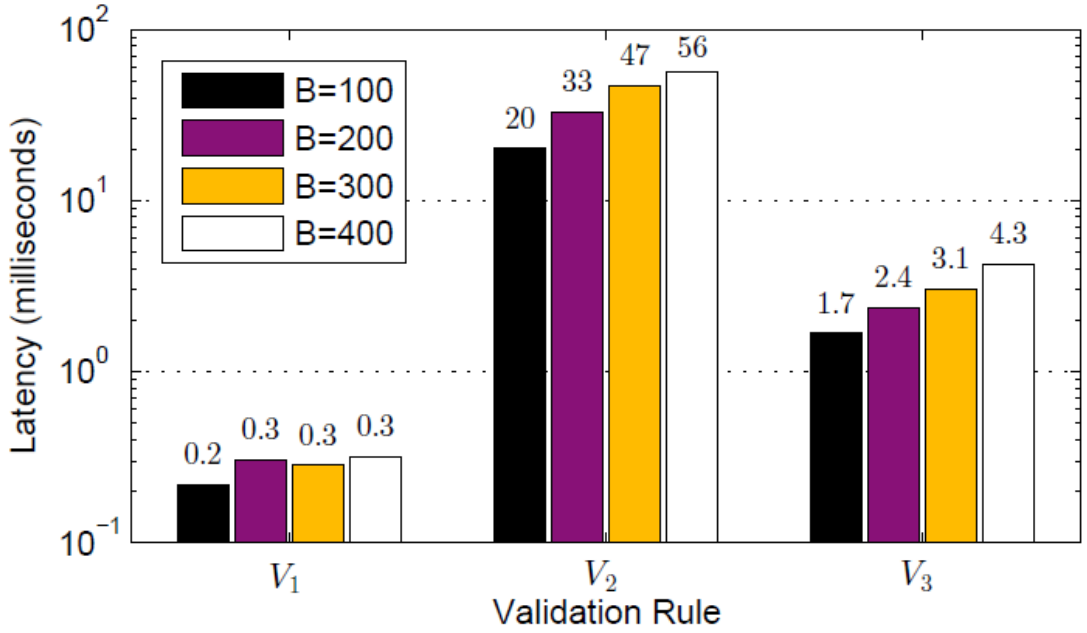
- From a real-world AMI, data extracted from 50 meters
- Data covers 13 months (May 2012 – June 2013)

- Implemented on top of Storm, a widely-used SPE (e.g., used in Twitter)

- Evaluated in terms of throughput (tuples/second) and processing latency (milliseconds)

Evaluation


Throughput (tuples/second)
Millions of smart meters (Hourly readings)
Input Rate (tuples/second)
B=100
B=200
B=300
B=400
0
2000
4000
6000
8000
10000
0
5
10
15
20
25


Latency (milliseconds)
Validation Rule
B=100
B=200
B=300
B=400
10^{2}
10^{1}
10^{0}
10^{-1}
0.2
0.3
0.3
0.3
20
33
47
56
1.7
2.4
3.1
4.3
V_1
V_2
V_3

Agenda

1. Problem statement
2. Preliminaries – Data Streaming
3. Streaming-based Data Validation
4. Conclusions

Conclusions

- Streaming-based Data Validation
  - Expressive / Distributed-Parallel / Online
  - Implemented on top of the Storm SPE
  - Evaluated with real-world AMI data