BIG DATA STREAM MINING WITH **APACHE SAMOA**

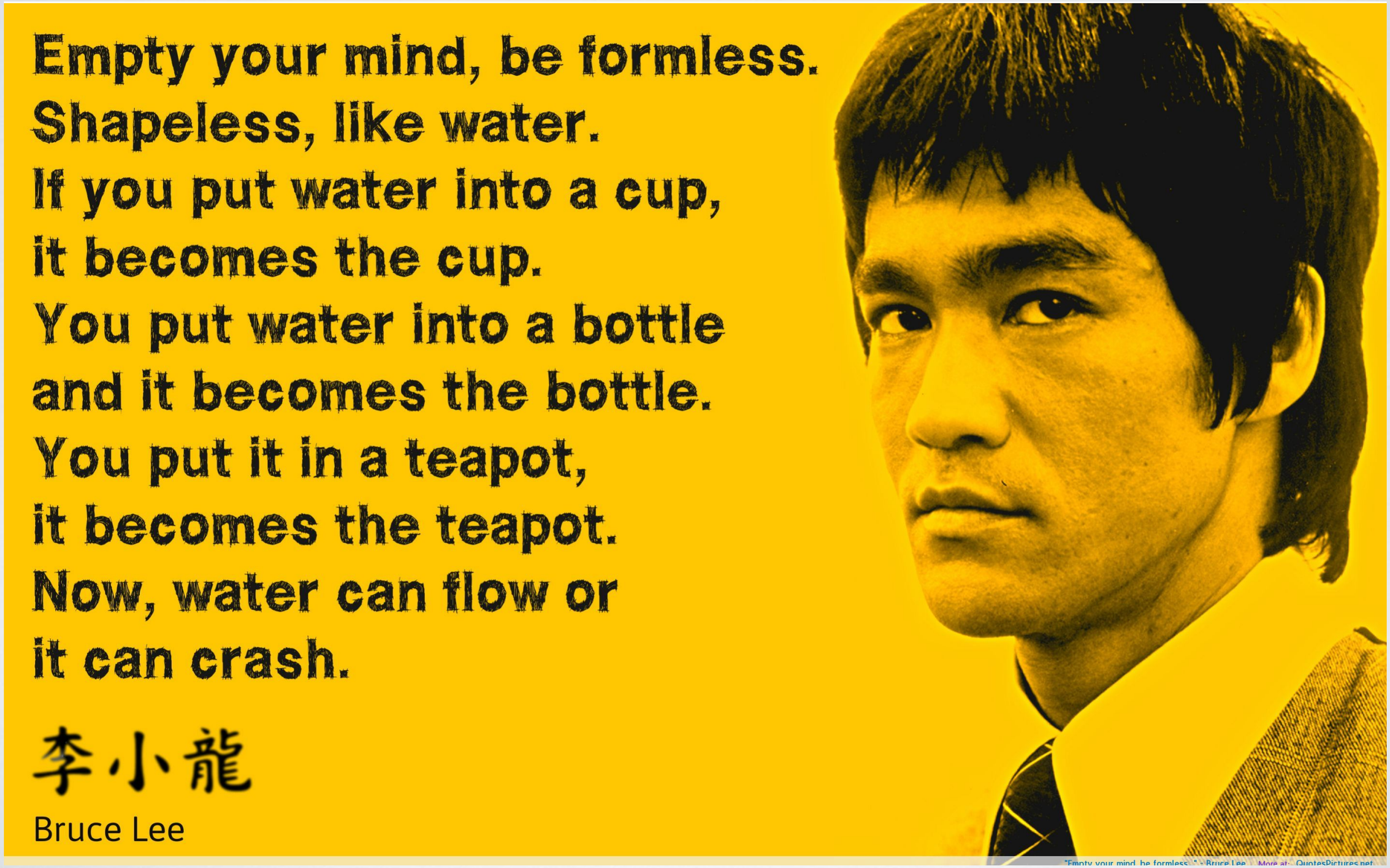
Albert Bifet @abifet
DBWeb Team
INFRES Department

10 November 2015

MOTIVATION

REAL TIME ANALYTICS

REAL TIME ANALYTICS

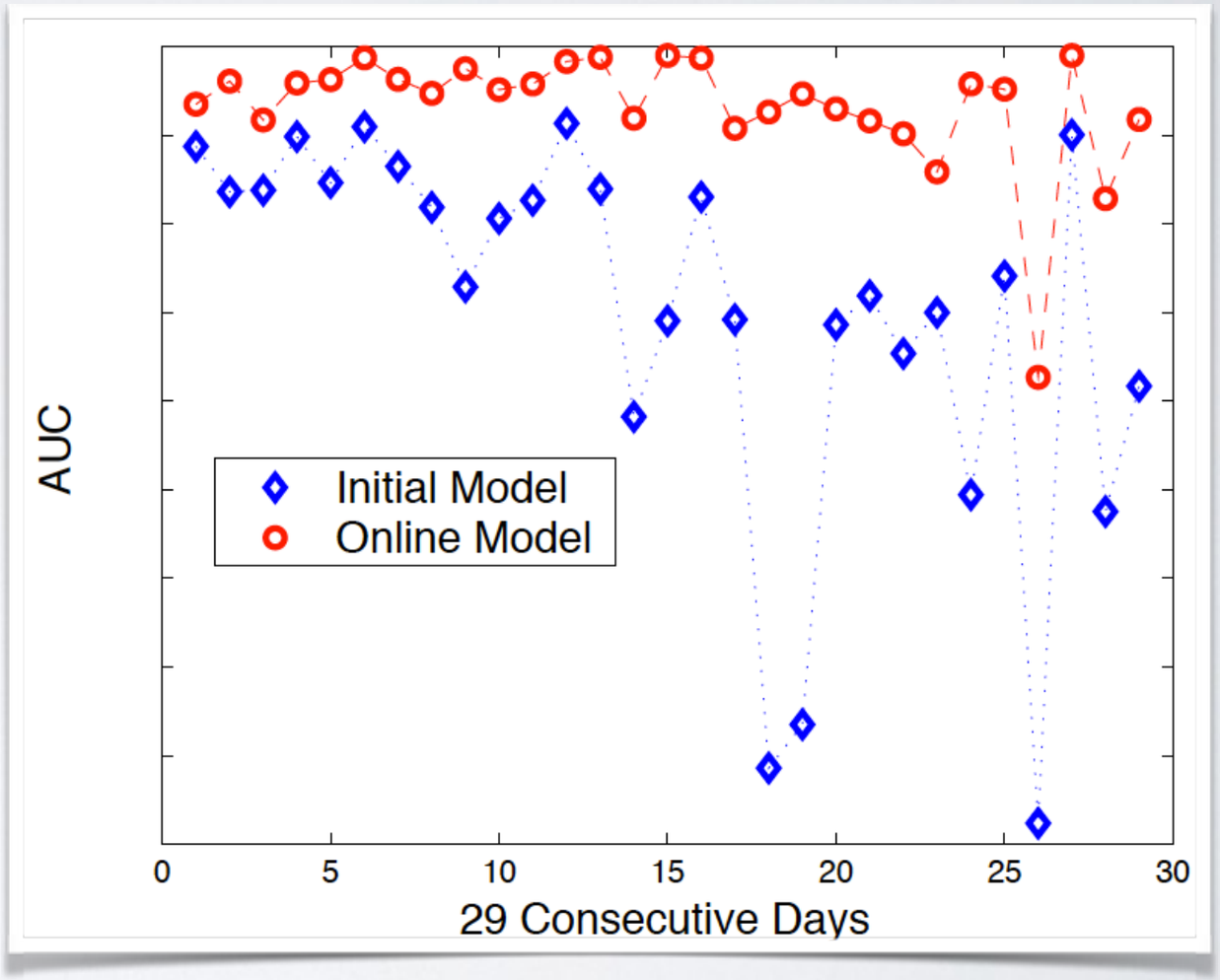


APACHE SA(MOA) VISION

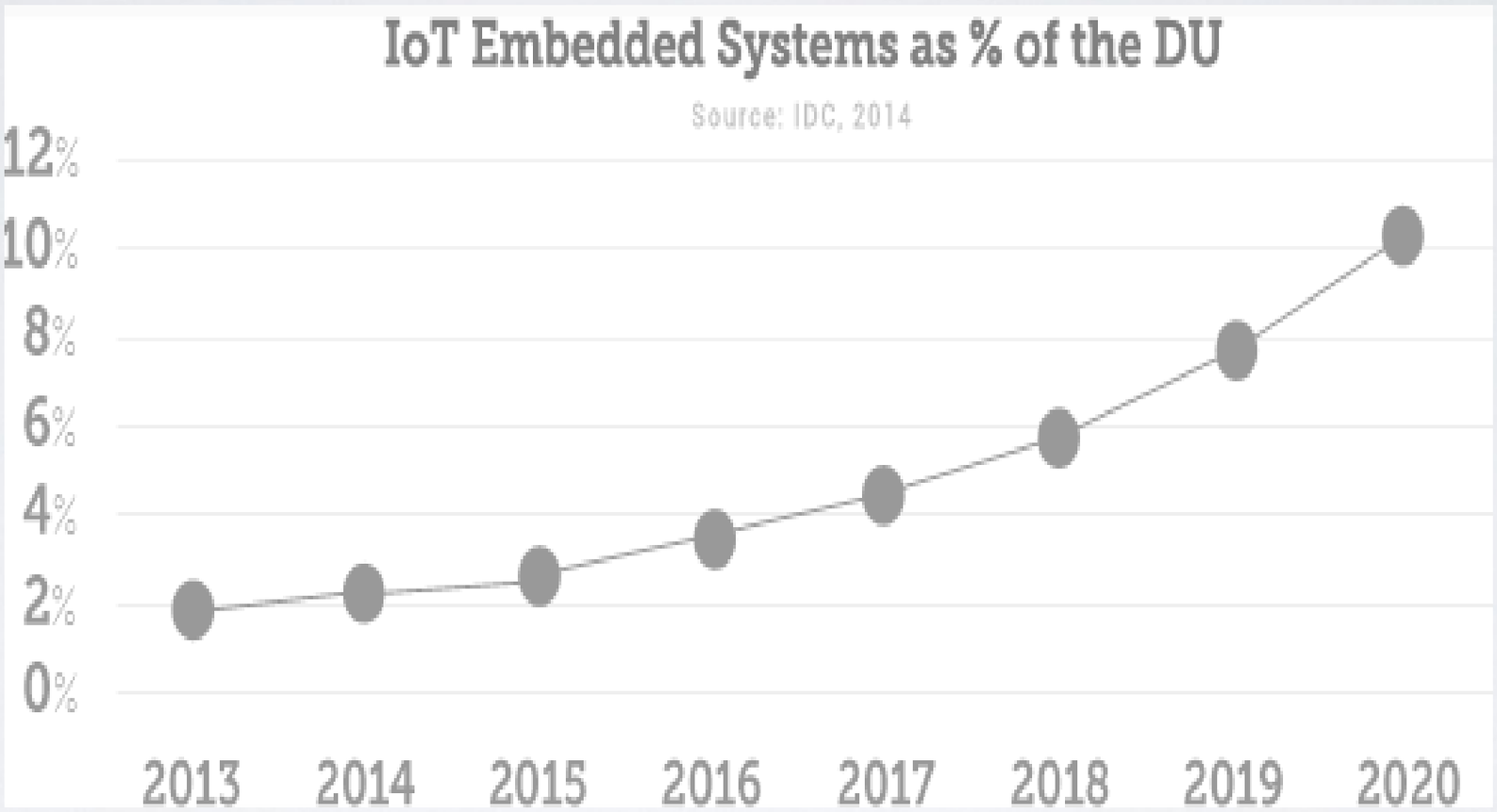
- Data Stream mining platform
  - Library of state-of-the-art algorithms for practitioners
  - Development and collaboration framework for researchers
- Algorithms & Systems

IMPORTANCE

- Example: spam detection in comments on Yahoo News
- Trends change in time
- Need to retrain model with new data



INTERNET OF THINGS



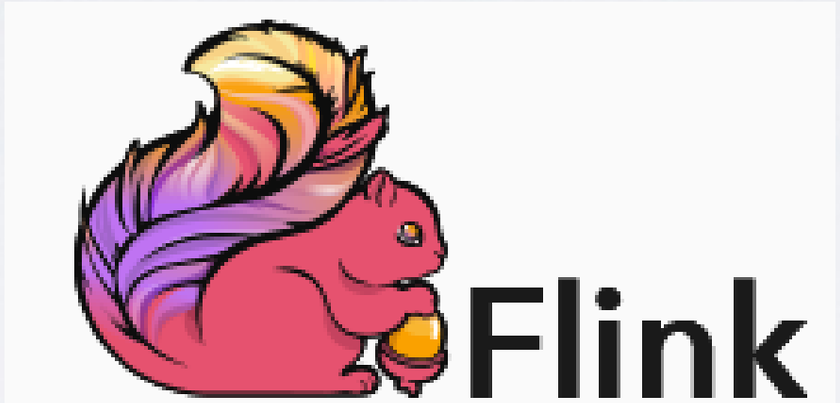
- EMC Digital Universe, 2014

BIG DATA STREAM

- Volume + Velocity (+ Variety)
- Too large for single commodity server main memory
- Too fast for single commodity server CPU
- A solution should be:
  - Distributed
  - Scalable

BIG DATA PROCESSING ENGINES

- Low latency



S4 distributed stream computing platform

APACHE STORM™
Distributed • Resilient • Real-time

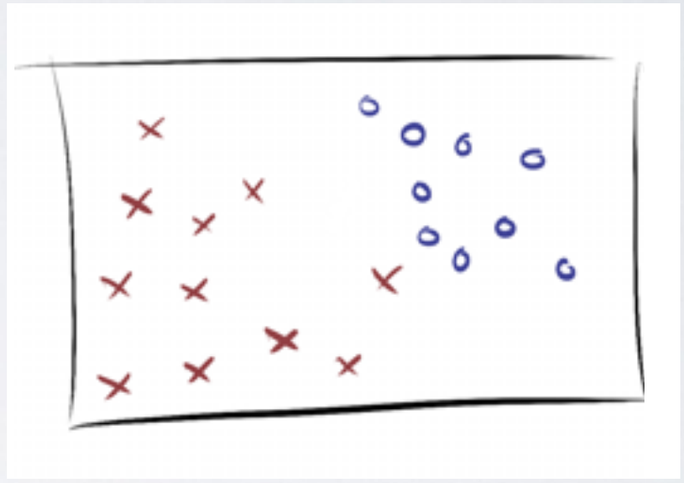
samza

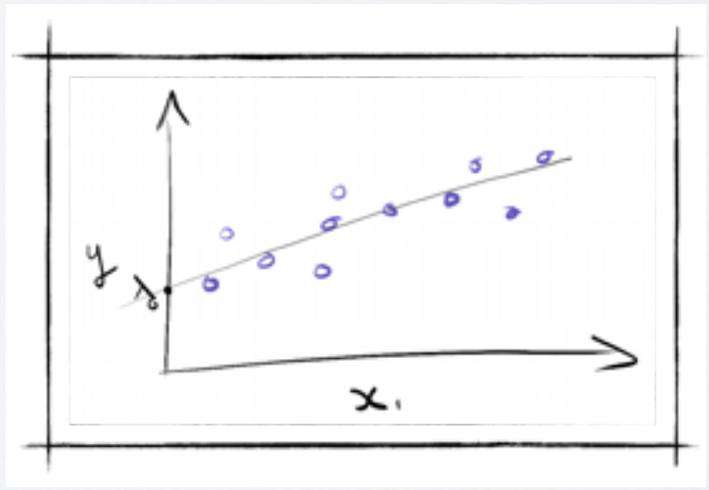
- High Latency (Not real time)

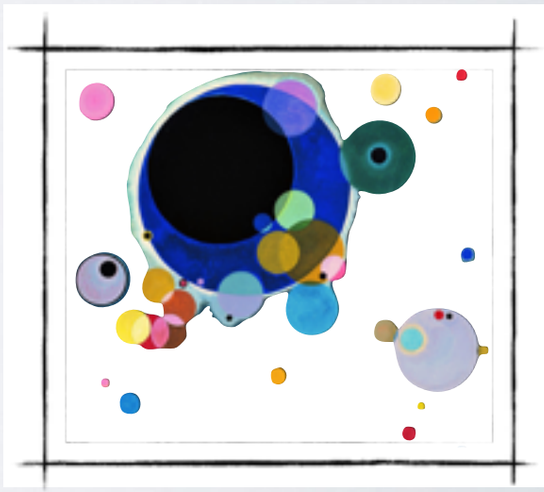
Spark Streaming

MACHINE LEARNING

- Classification
- Regression
- Clustering
- Frequent Pattern Mining







WHAT IS **MOA**?

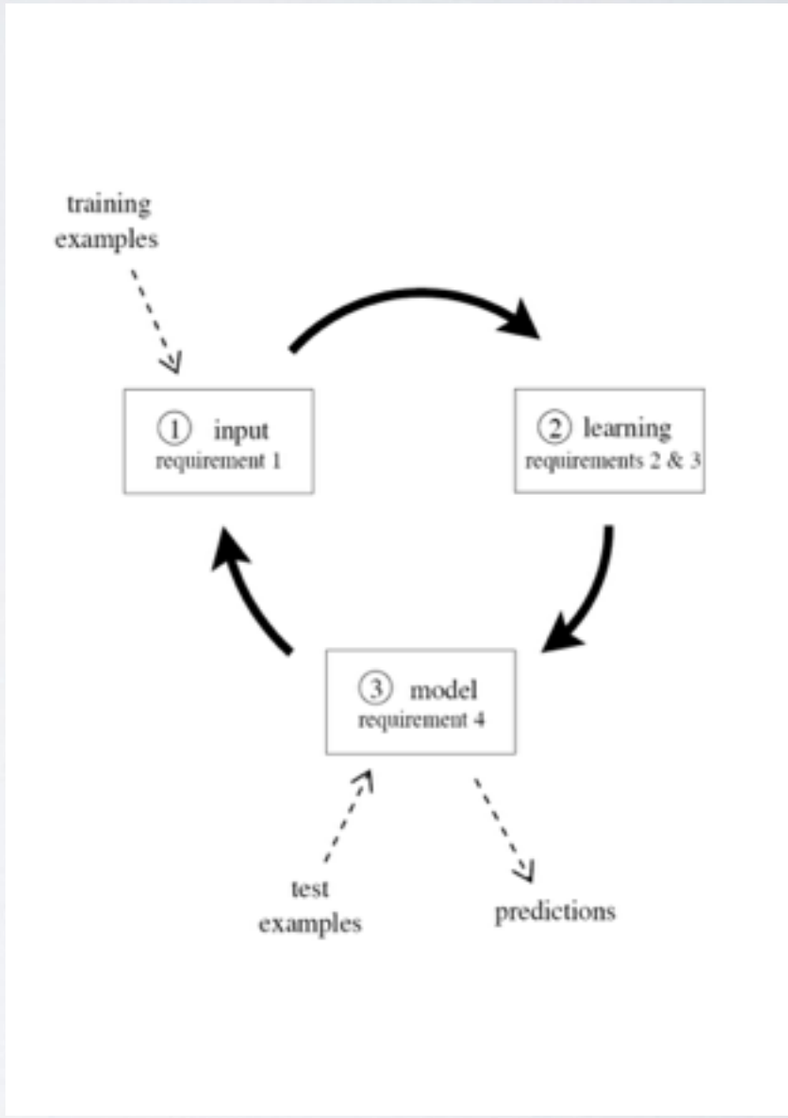
MOA

- {M}assive {O}nline {A}nalysis is a framework for online learning from data streams.
- It is closely related to WEKA
- It includes a collection of offline and online as well as tools for evaluation:
  - classification, regression
  - clustering, frequent pattern mining

- Easy to extend, design and run experiments

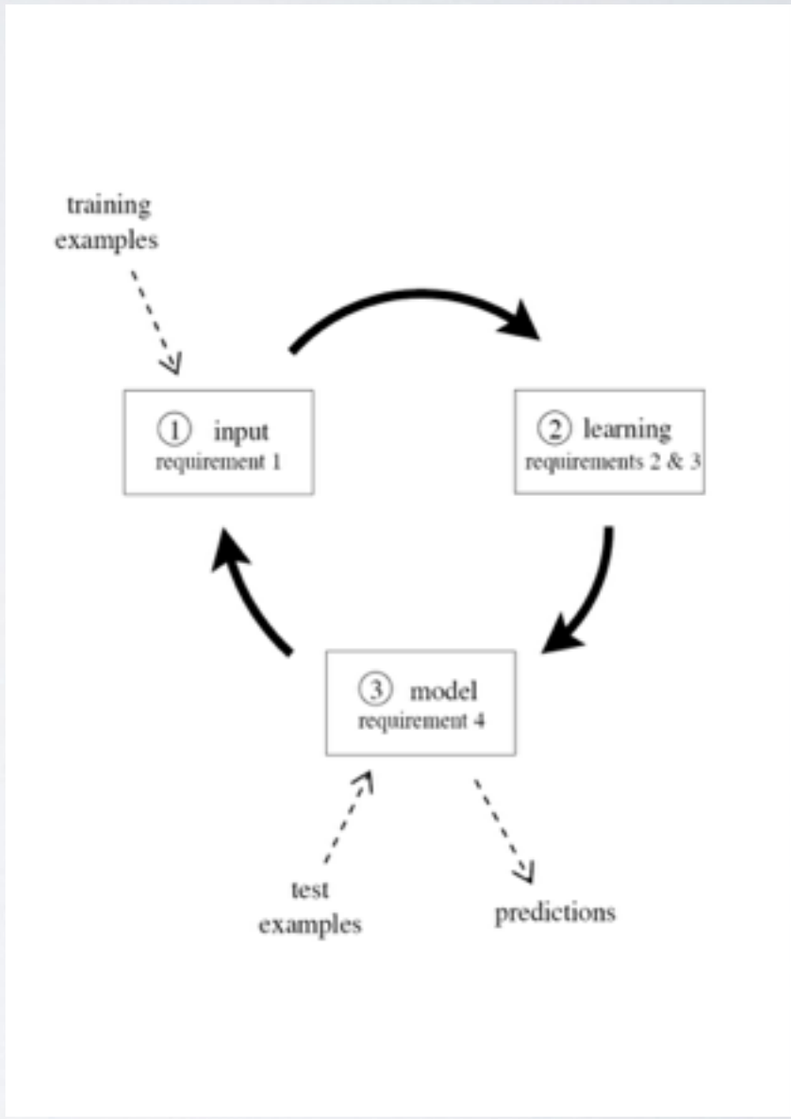
STREAM SETTING

- Process an example at a time,and inspect it only once (at most)
- Use a limited amount of memory
- Work in a limited amount of time
- Be ready to predict at any point



STREAM EVALUATION

- Holdout Evaluation
- Interleaved Test-Then-Train or Prequential

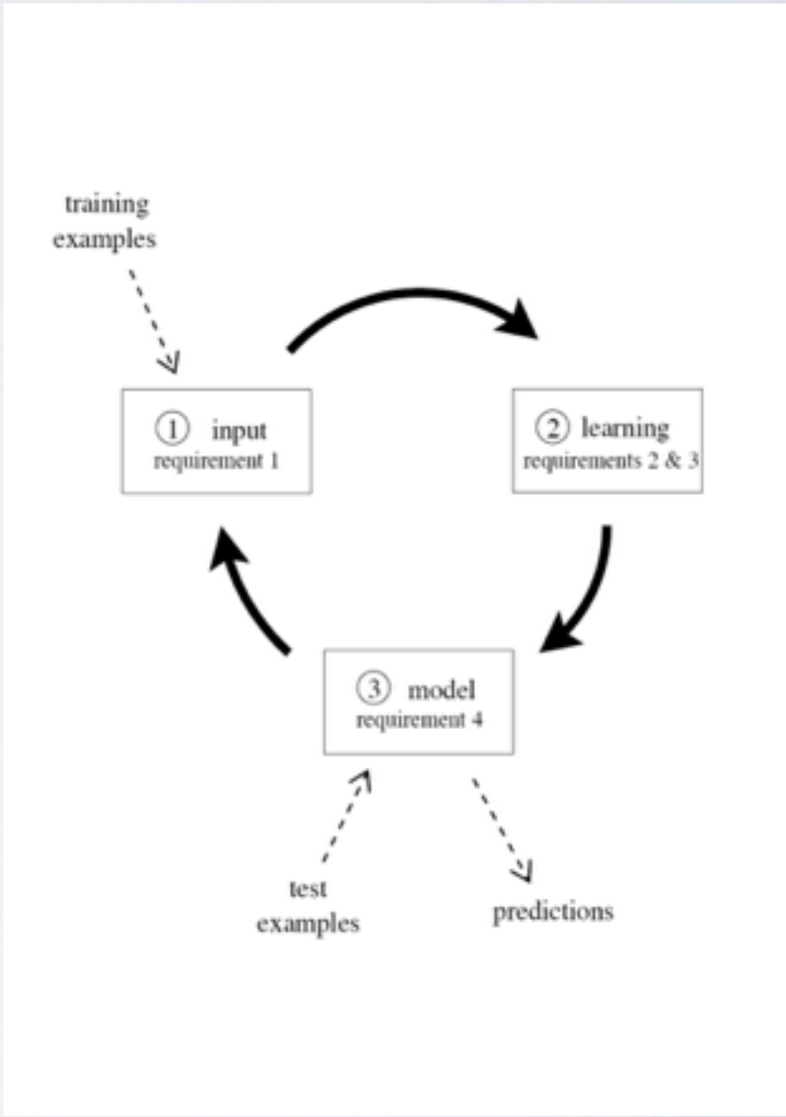


STREAM EVALUATION

Holdout an independent test set

- Apply the current decision model to the test set, at regular time intervals

- The loss estimated in the holdout is an unbiased estimator



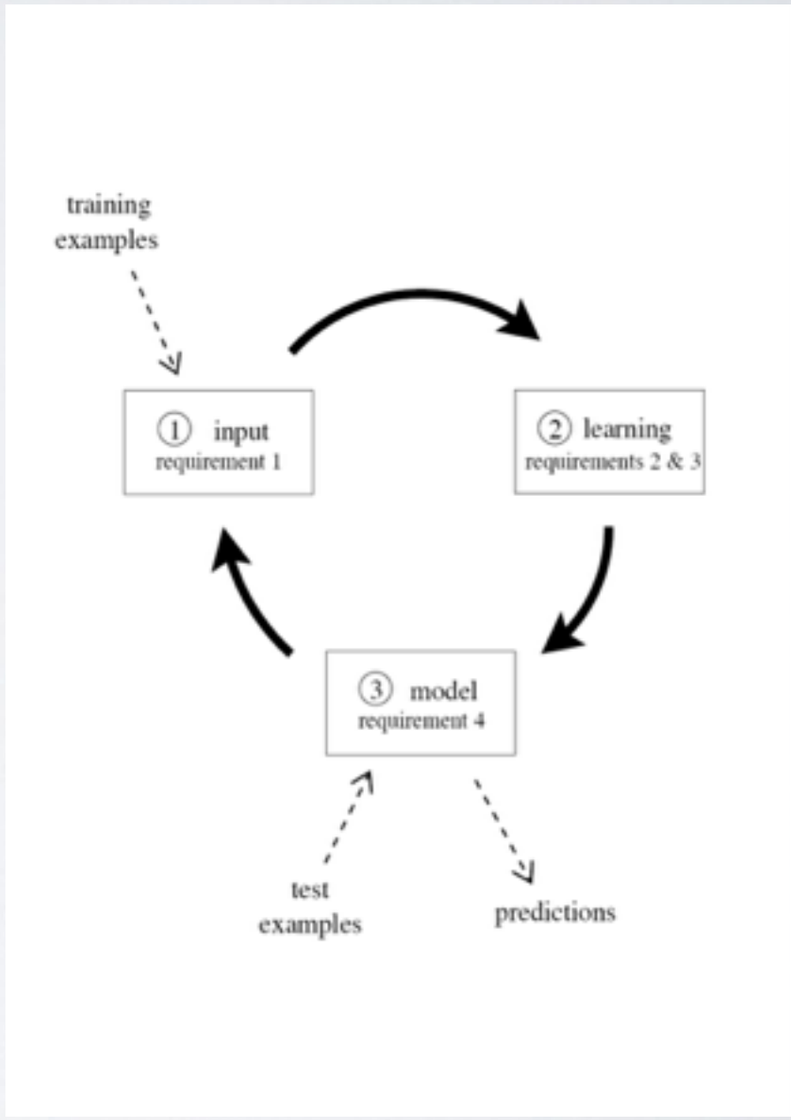
STREAM EVALUATION

Prequential Evaluation

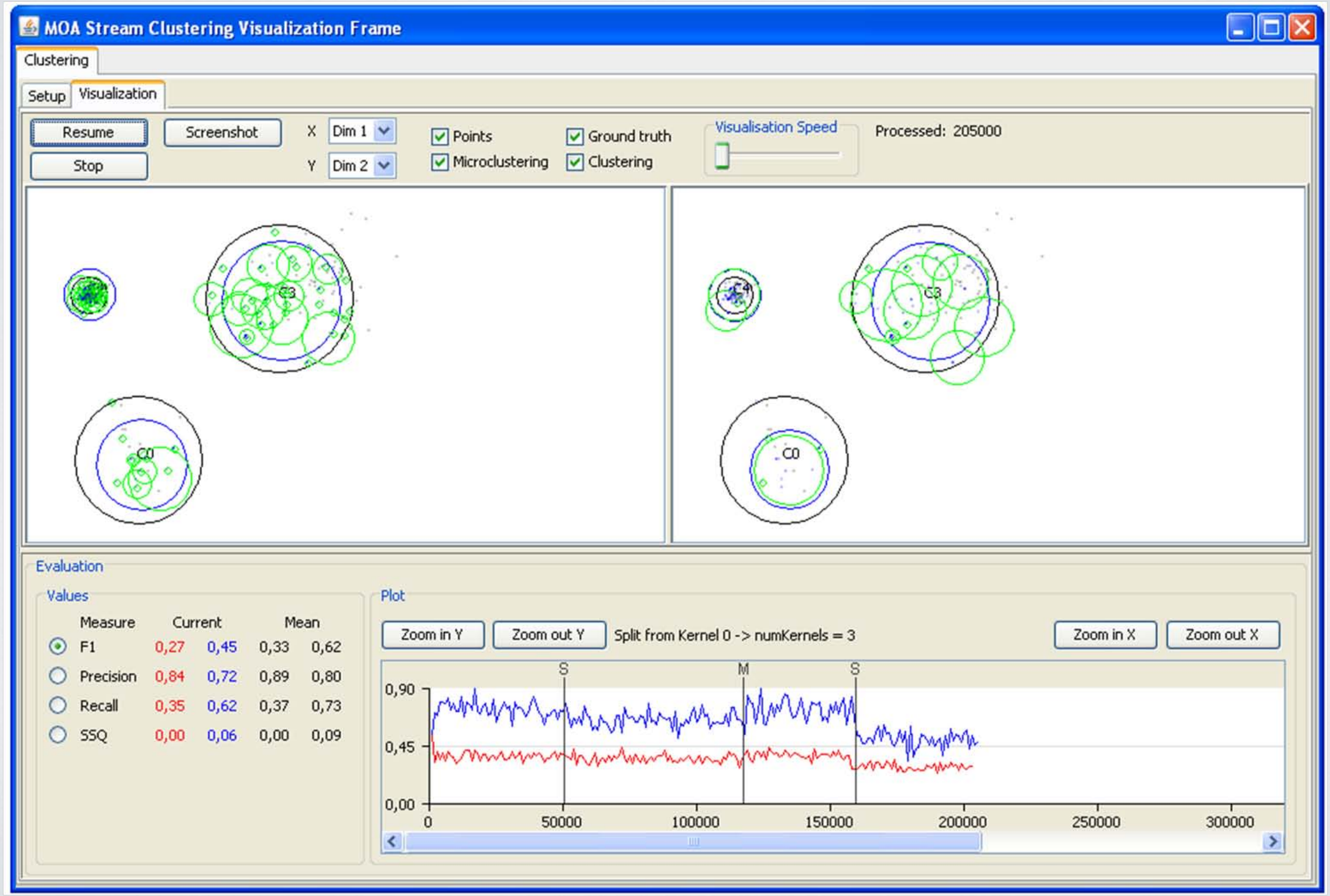
- The error of a model is computed from the sequence of examples.

- For each example in the stream, the actual model makes a prediction based only on the example attribute-values.

$$S = \sum_{i=1}^{n} L(y_i, \hat{y}_i).$$



CLUSTERING


MOA Stream Clustering Visualization Frame
Clustering
Setup
Visualization
Resume
Stop
Screenshot
X
Dim 1
Y
Dim 2
Points
Microclustering
Ground truth
Clustering
Visualisation Speed
Processed: 205000
C3
C4
C0
Evaluation
Values
Measure
Current
Mean
F1 0,27 0,45 0,33 0,62
Precision 0,84 0,72 0,89 0,80
Recall 0,35 0,62 0,37 0,73
SSQ 0,00 0,06 0,00 0,09
Plot
Zoom in Y
Zoom out Y
Split from Kernel 0 -> numKernels = 3
Zoom in X
Zoom out X
S
M
S
0,90
0,45
0,00
0
50000
100000
150000
200000
250000
300000

COMMAND LINE

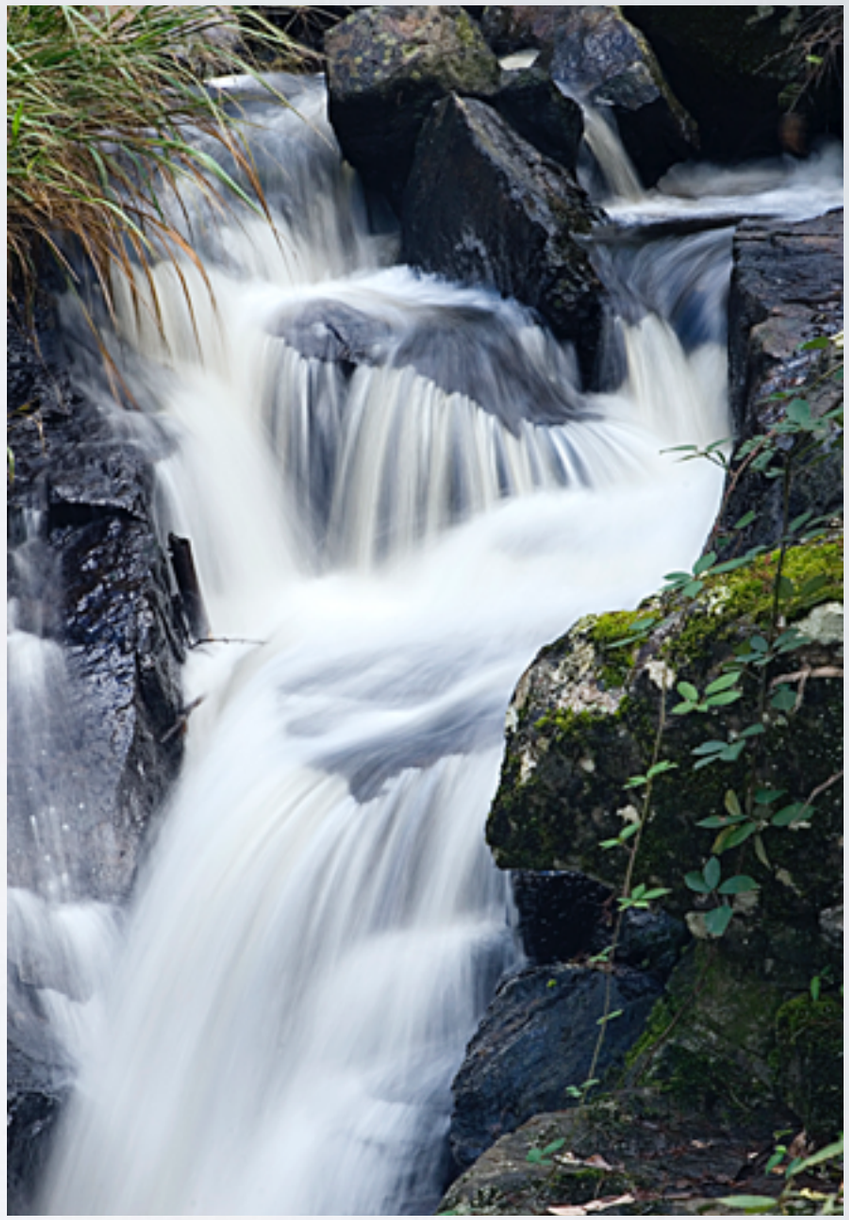
- `java -cp .:moa.jar:weka.jar -javaagent:sizeofag.jar moa.DoTask "EvaluatePeriodicHeldOutTest -l DecisionStump -s generators.WaveformGenerator -n 100000 -i 100000000 -f 1000000" > dsresult.csv`

- This command creates a comma separated values file:
  - training the DecisionStump classifier on the WaveformGenerator data,
  - using the first 100 thousand examples for testing,
  - training on a total of 100 million examples,
  - and testing every one million examples

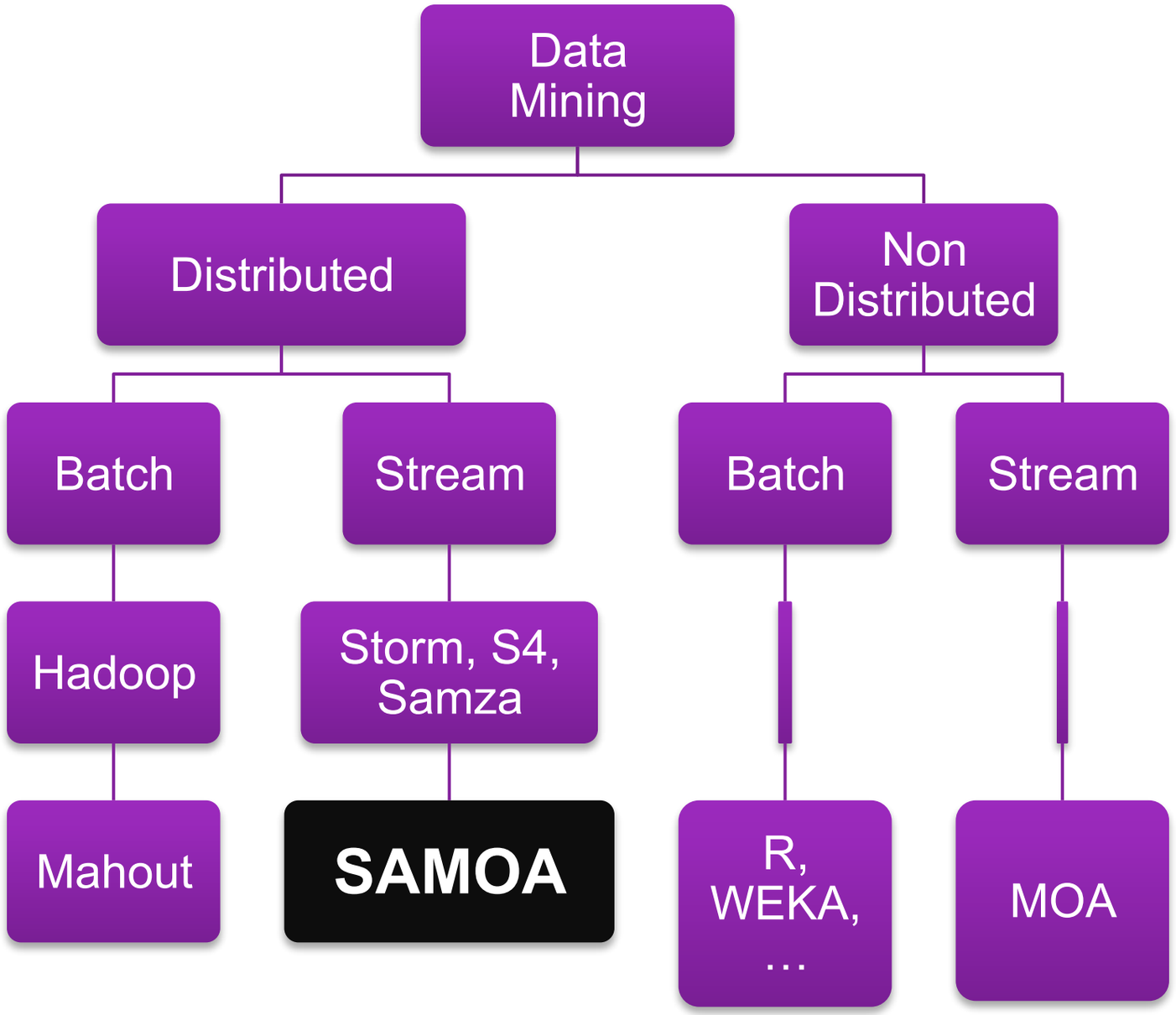
WHAT IS **APACHE SAMOA**?

STREAMING MODEL

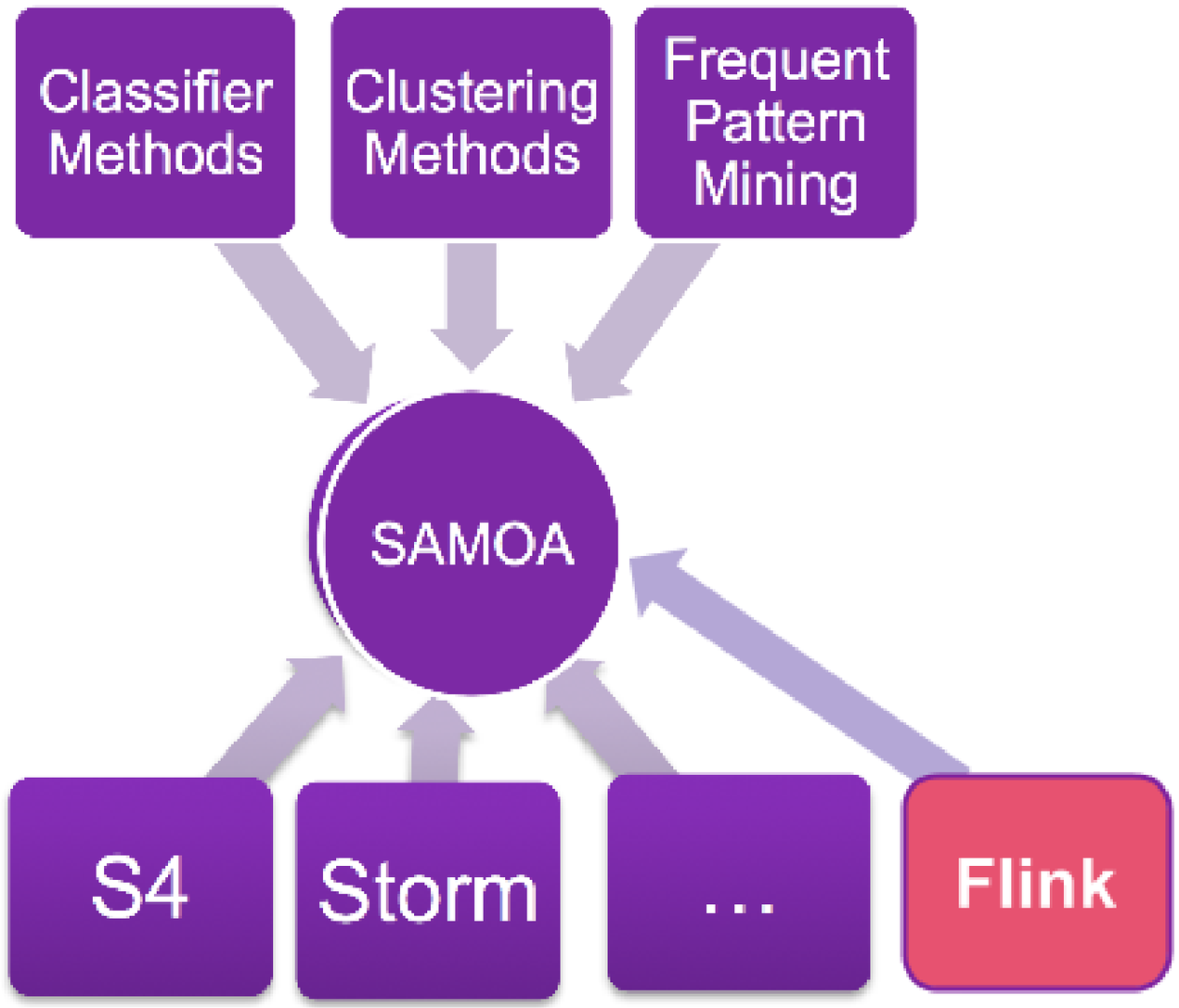
- Sequence is potentially infinite
- High amount of data, high speed of arrival
- Change over time (concept drift)
- Approximation algorithms
  (small error with high probability)
  - Single pass, one data item at a time
  - Sub-linear space and time per data item



TAXONOMY

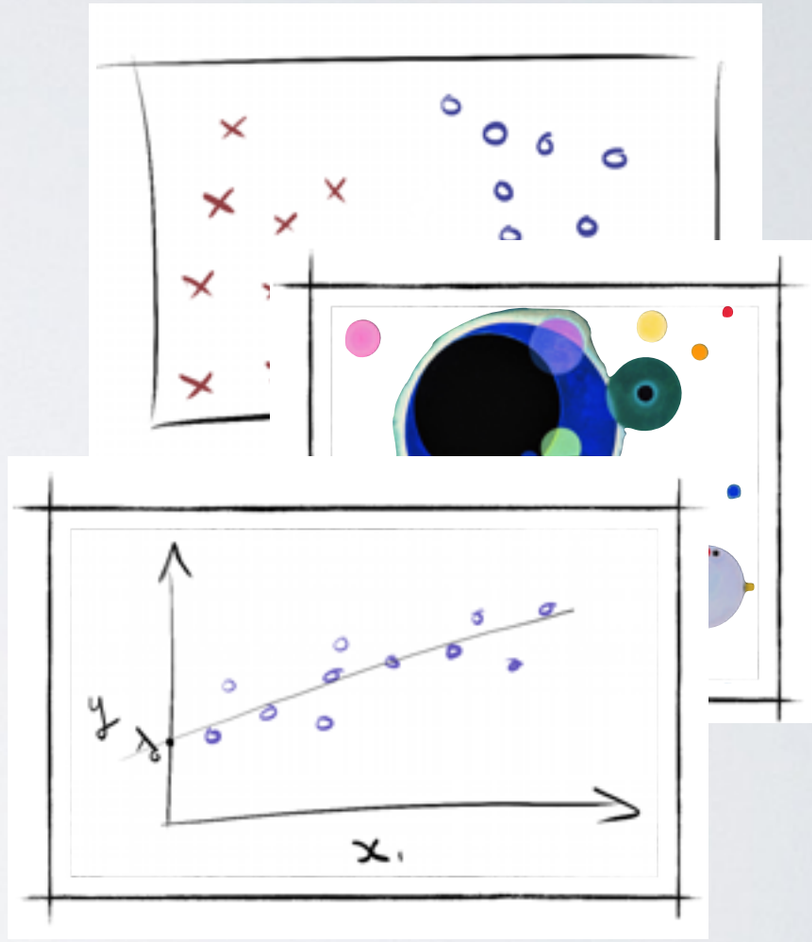

Data Mining
Distributed
Non Distributed
Batch
Stream
Batch
Stream
Hadoop
Storm, S4, Samza
Mahout
SAMOA
R, WEKA, …
MOA

ARCHITECTURE



STATUS

- Parallel algorithms
  - Classification (Vertical Hoeffding Tree)
  - Clustering (CluStream)
  - Regression (Adaptive Model Rules)



- Execution engines

STORM

S4 distributed stream computing platform

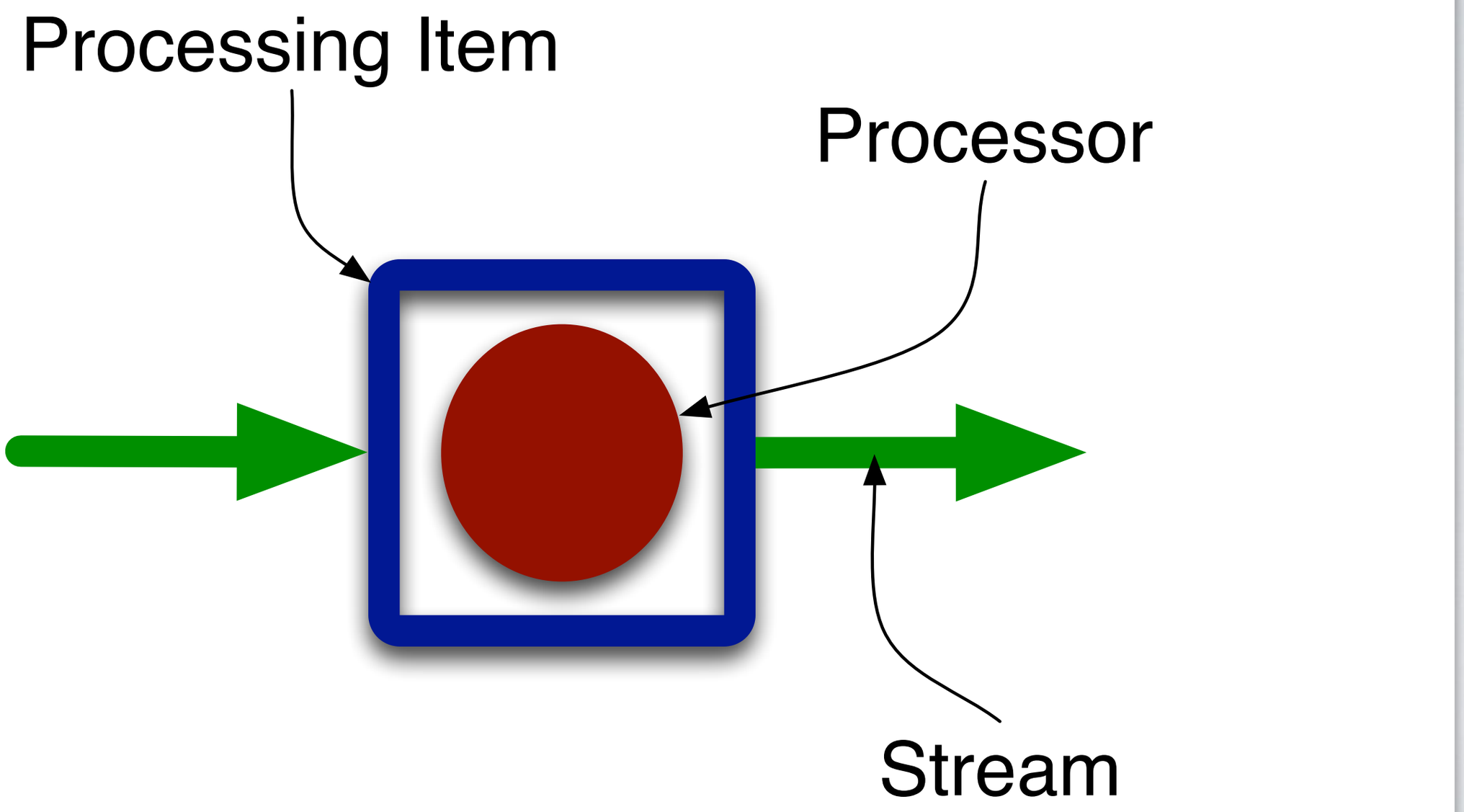
samza

Flink

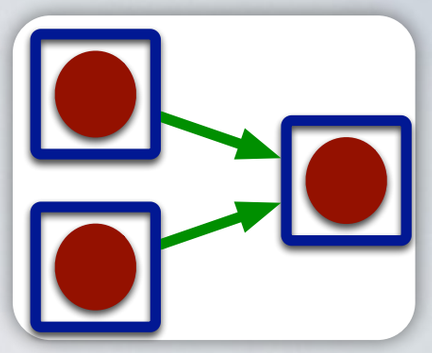
IS SAMOA USEFUL FOR YOU?

- Only if you need to deal with:
  - **Large** *fast* data
    - Evolving process (model updates)
- What is happening now?
  - Use feedback in real-time
  - Adapt to changes faster

ML DEVELOPER API



ML DEVELOPER API

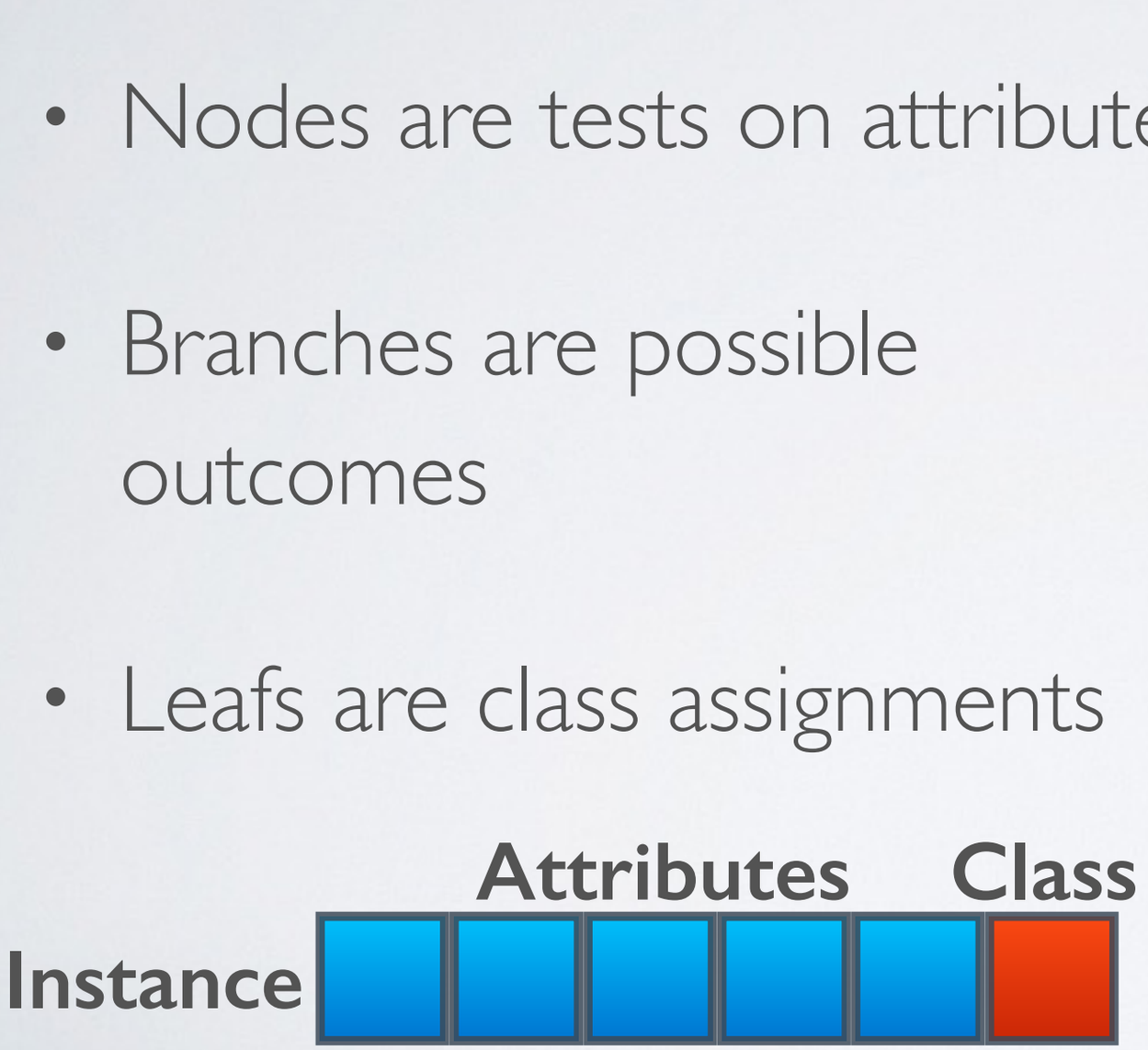


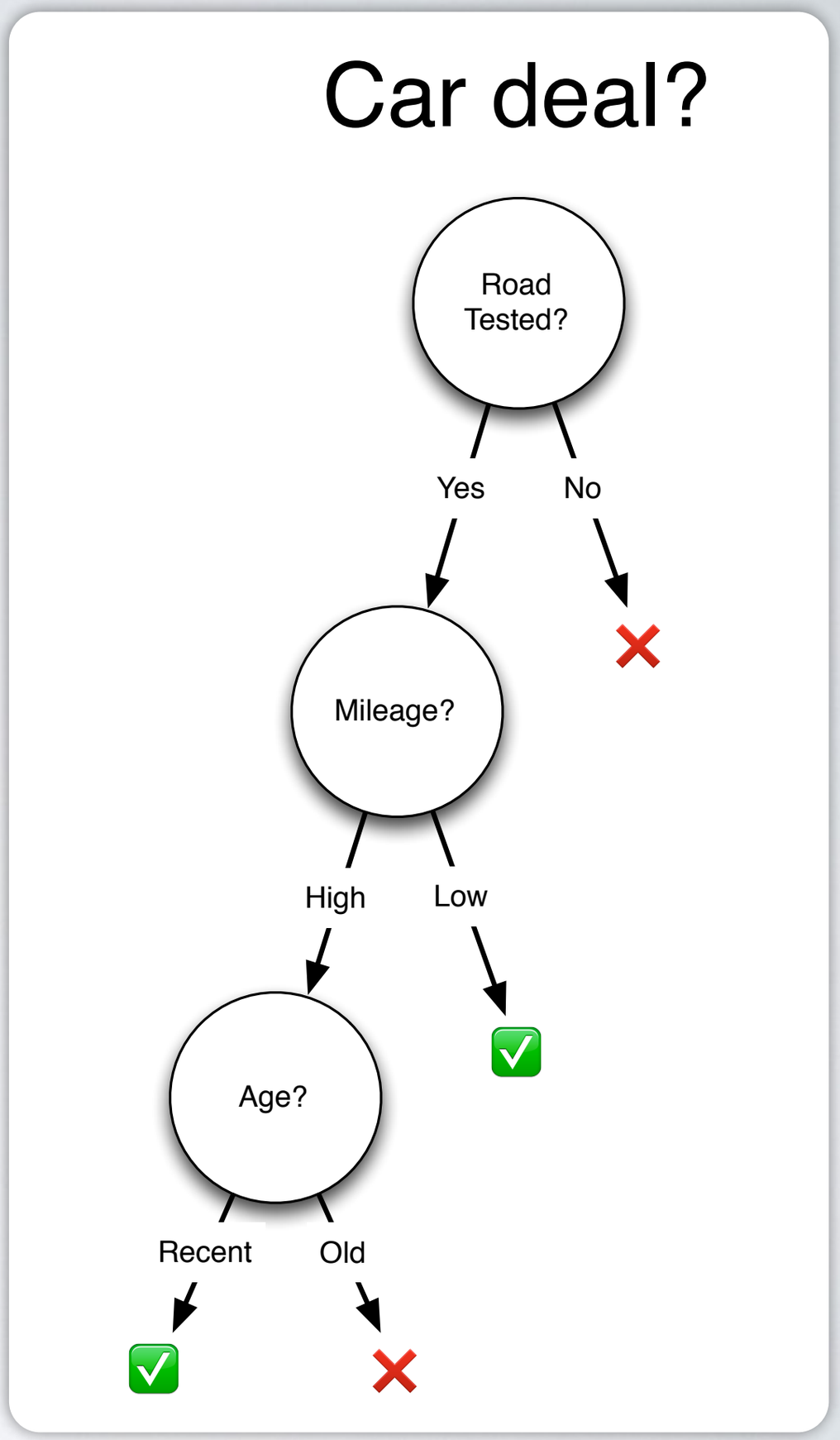
```
TopologyBuilder builder;
Processor sourceOne = new SourceProcessor();
builder.addProcessor(sourceOne);
Stream streamOne = builder.createStream(sourceOne);

Processor sourceTwo = new SourceProcessor();
builder.addProcessor(sourceTwo);
Stream streamTwo = builder.createStream(sourceTwo);

Processor join = new JoinProcessor());
builder.addProcessor(join)
      .connectInputShuffle(streamOne)
      .connectInputKey(streamTwo);
```

VERTICAL HOEFFDING TREE (VHT)

DECISION TREE

- Nodes are tests on attributes
- Branches are possible outcomes
- Leafs are class assignments





P. Domingos and G. Hulten, "Mining High-Speed Data Streams," KDD '00

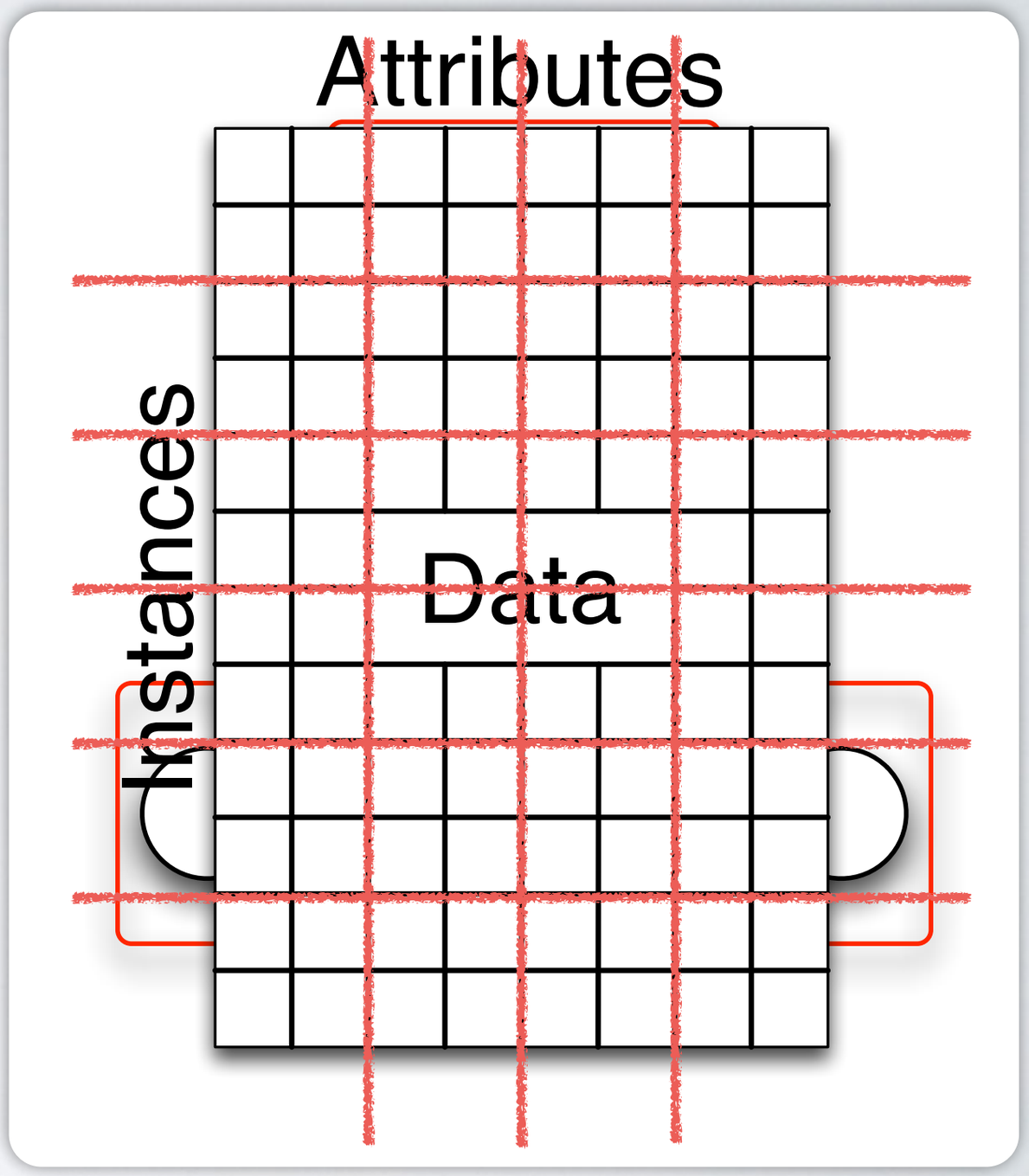
HOEFFDING TREE

- Sample of stream enough for near optimal decision
- Estimate merit of alternatives from prefix of stream
- Choose sample size based on statistical principles
- When to expand a leaf?
  - Let x_1 be the most informative attribute, x_2 the second most informative one
  - Hoeffding bound: split if

$$\Delta G(x_1, x_2) > \epsilon = \sqrt{\frac{R^2 \ln(1/\delta)}{2n}}$$

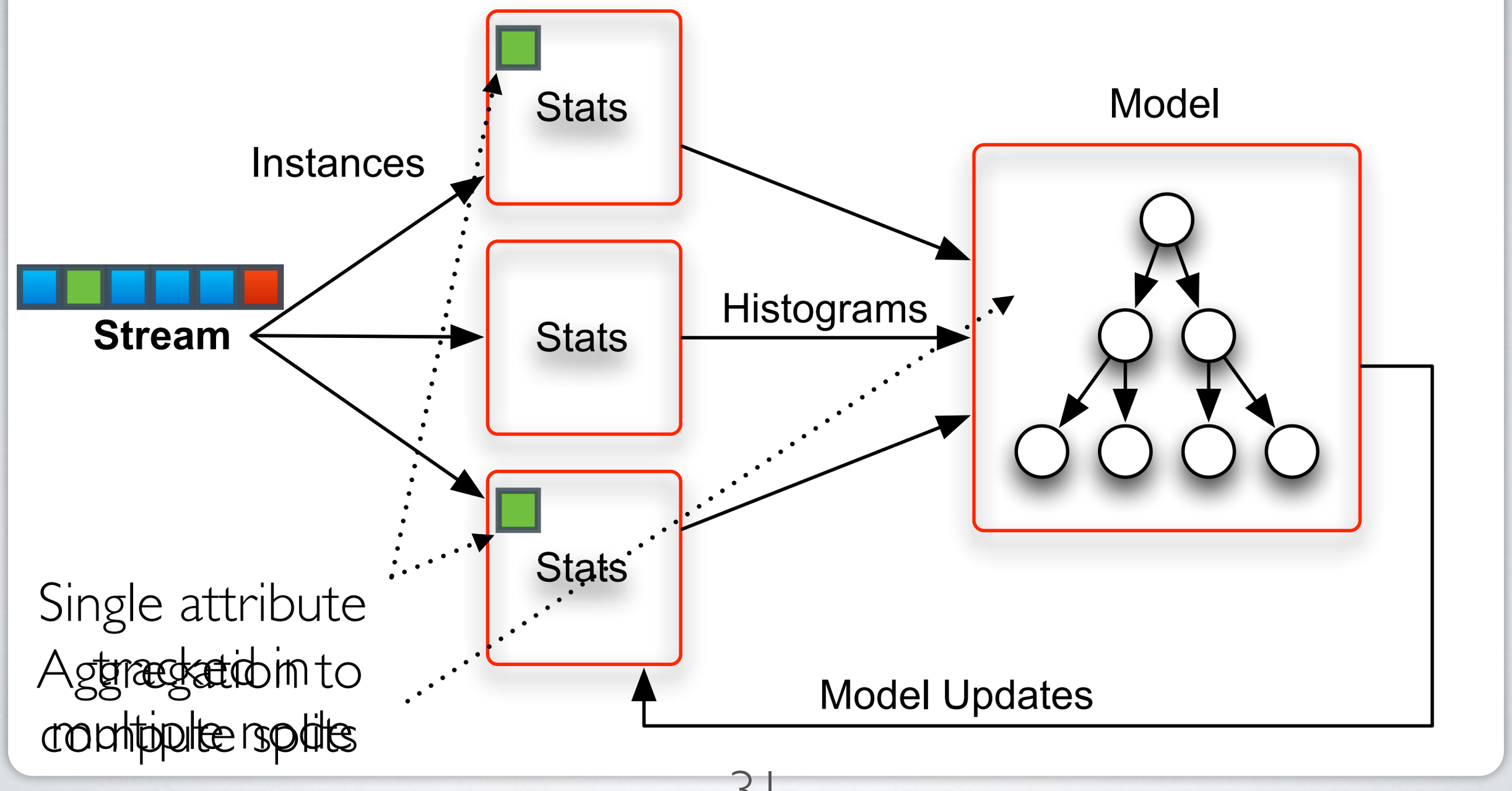
PARALLEL DECISION TREES

- Which kind of parallelism?
  - Task
  - Data
    - Horizontal
    - **Vertical**



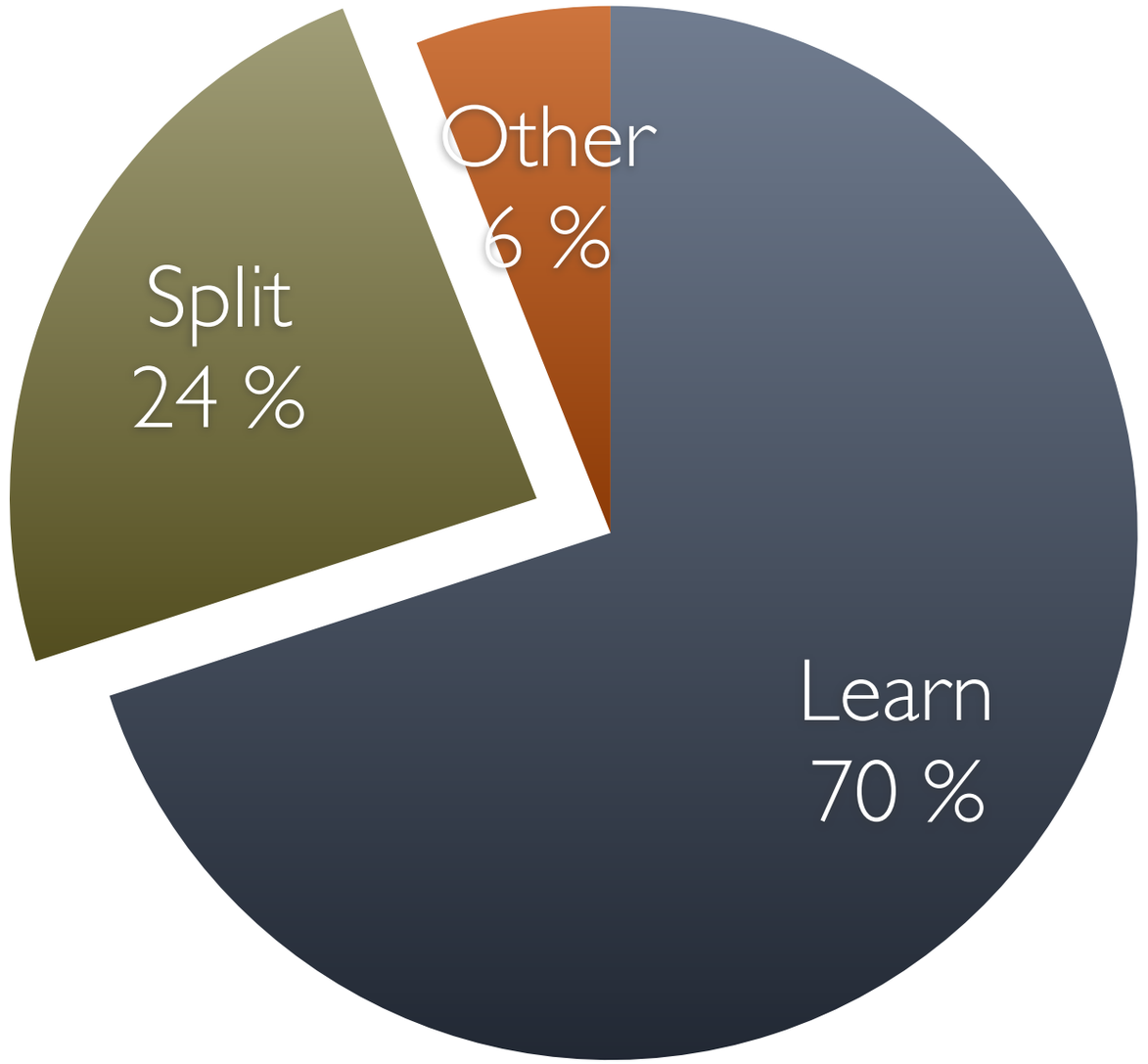
Y. Ben-Haim and E. Tom-Tov, "A Streaming Parallel Decision Tree Algorithm," JMLR, vol. 11, pp. 849–872, 2010

HORIZONTAL PARALLELISM

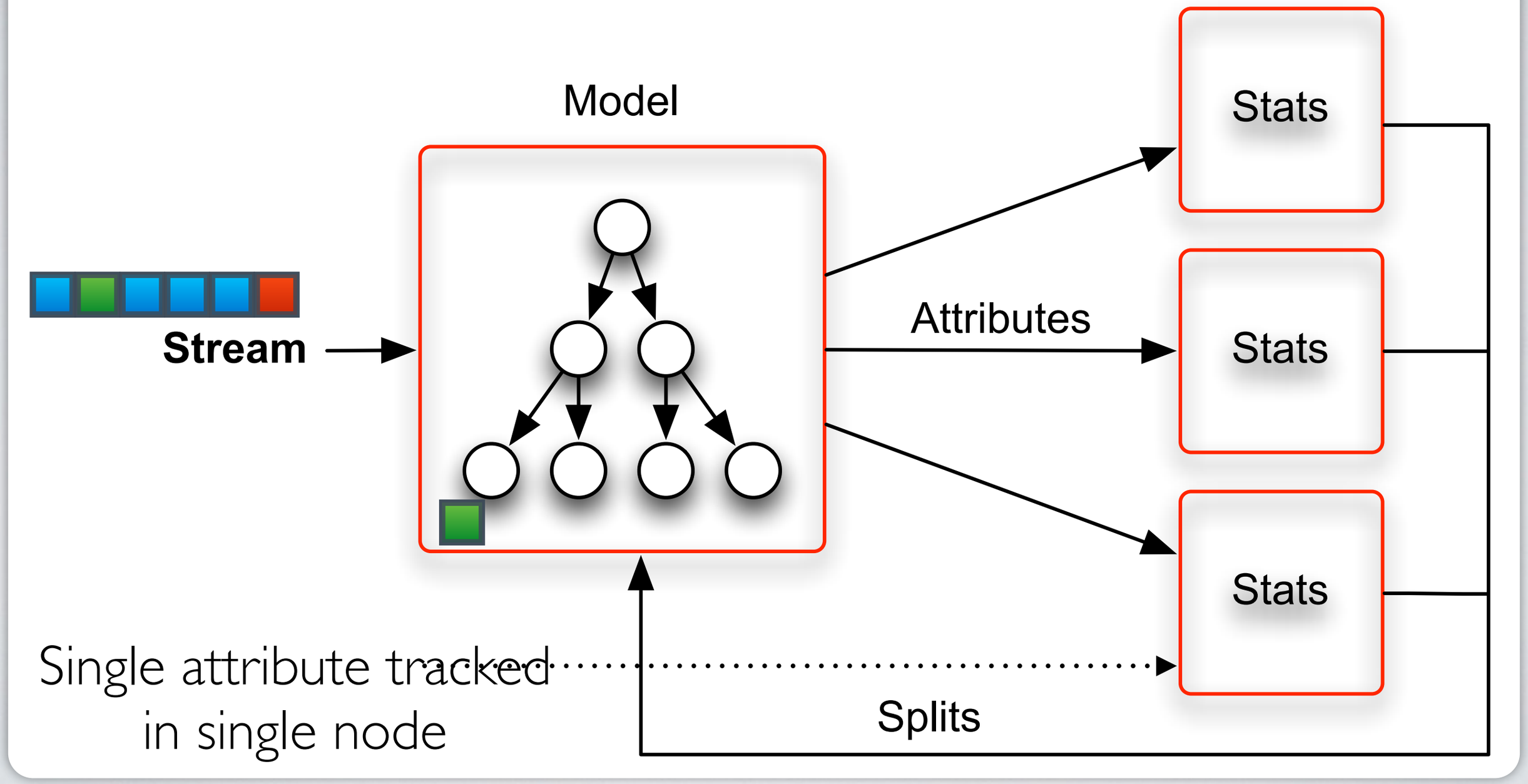


HOEFFDING TREE PROFILING

CPU time for training 100 nominal and 100 numeric attributes



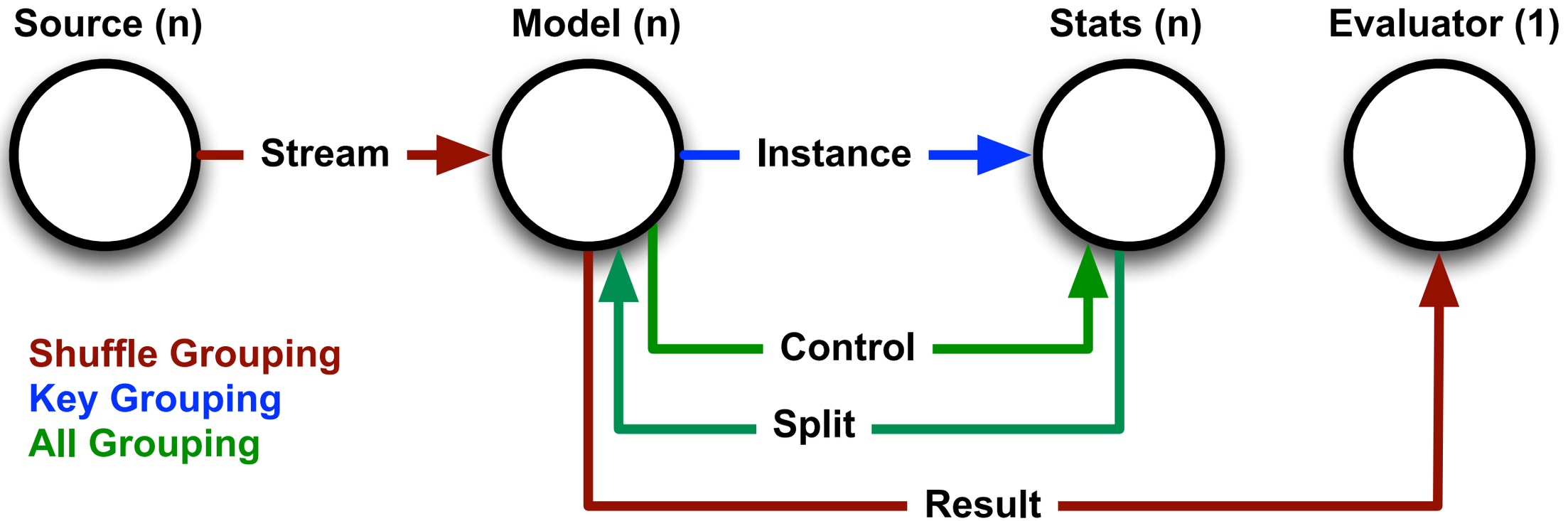
VERTICAL PARALLELISM



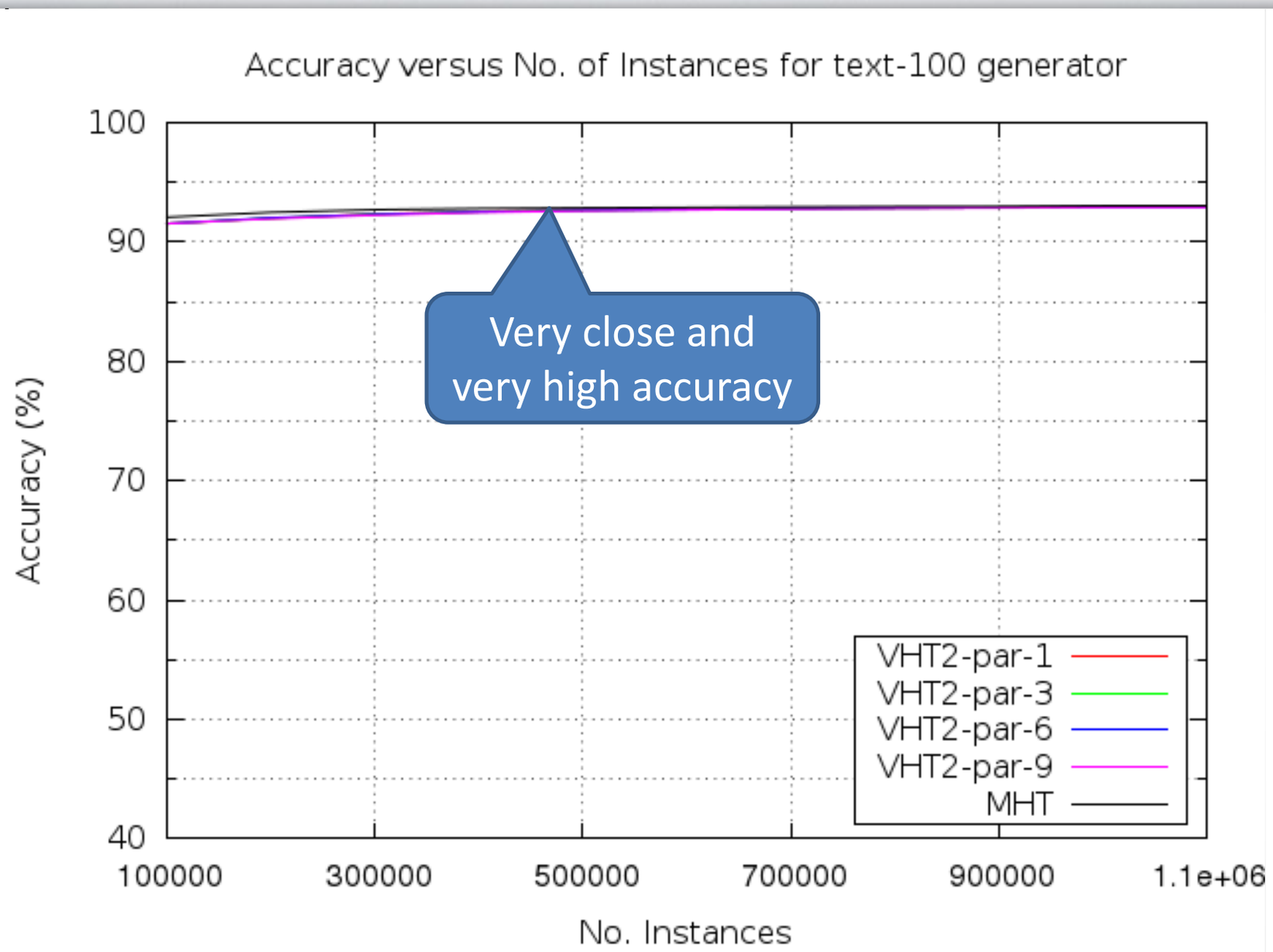
ADVANTAGES OF VERTICAL

- High number of attributes => high level of parallelism (e.g., documents)
- Vs task parallelism
    - Parallelism observed immediately
- Vs horizontal parallelism
    - Reduced memory usage (no model replication)
    - Parallelized split computation

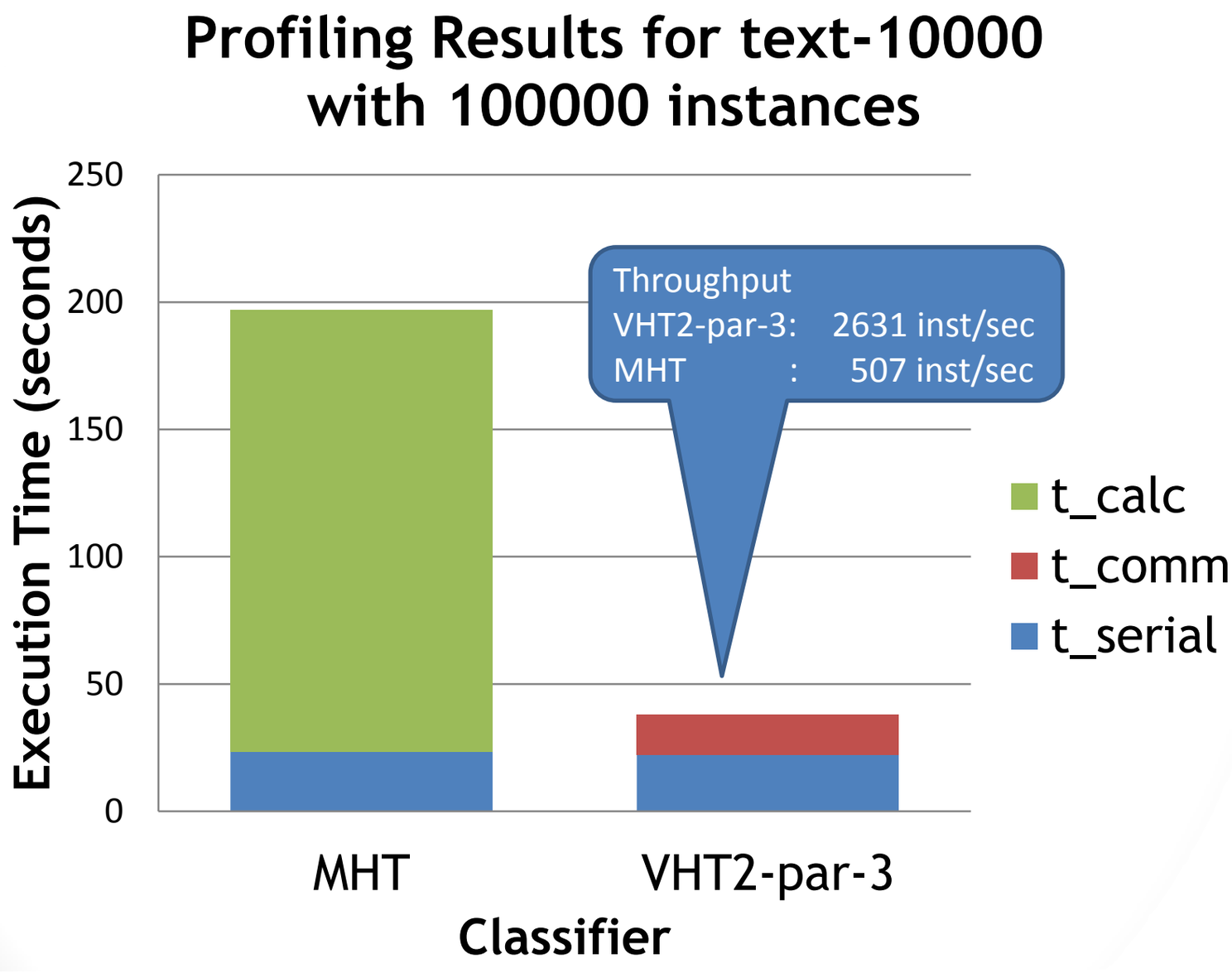
VERTICAL HOEFFDING TREE



ACCURACY


Accuracy versus No. of Instances for text-100 generator
Accuracy (%)
100
90
80
70
60
50
40
100000
300000
500000
700000
900000
1.1e+06
No. Instances
VHT2-par-1
VHT2-par-3
VHT2-par-6
VHT2-par-9
MHT
Very close and
very high accuracy

PERFORMANCE



SUMMARY

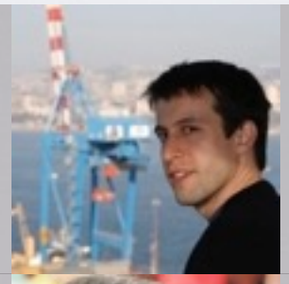
- Streaming is an important V of Big Data
- Mining big data streams is an open field
- MOA: Massive Online Analytics
  - Available and open-source http://moa.cms.waikato.ac.nz/
- SAMOA: A Platform for Mining Big Data Streams
  - Available and open-source (incubating @ASF) http://samoa.incubator.apache.org

OPEN CHALLENGES

- Distributed stream mining algorithms
  - Active & semi-supervised learning + crowdsourcing
    - Millions of classes (e.g., Wikipedia pages)
    - Multi-target learning
- System issues (load balancing, communication)
- Programming paradigms and abstractions

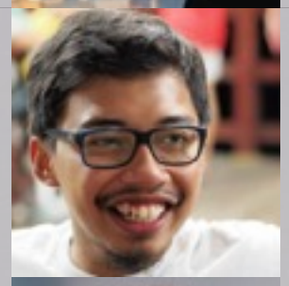
SAMOA TEAM

Albert Bifet

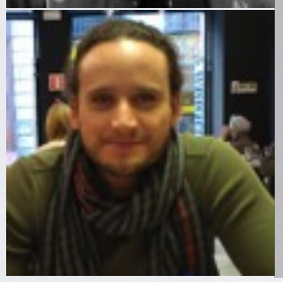


Matthieu Morel

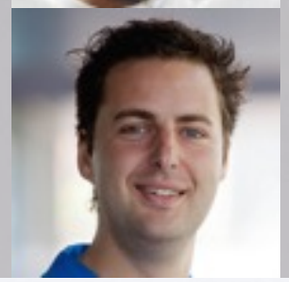
Gianmarco De Francisci Morales



Arinto Murdopo



Nicolas Kourtellis



Olivier Van Laere

SUPPORTING ORGANISATIONS

YAHOO! LABS

HUAWEI

Telefónica I+D

TELECOM ParisTech

Aalto University

THE UNIVERSITY OF WAIKATO
Te Whare Wānanga o Waikato

THANKS!

@ApacheSAMOA

https://samoa.incubator.apache.org