STREAM PROCESSING FOR REAL TIME ANALYTICS

BALABIT

Zsigmond, Ádám Olivér
adam.zsigmond@balabit.com
Software Engineer | **Balabit-Europe Kft**

Agenda

- Big Data problem to Solve
- How Stream Processing Topology will look like
- Implementing a Single Machine Stream Processing Framework
- Conclusions

Zsigmond, Ádám Olivér
adam.zsigmond@balabit.com
Software Engineer | **Balabit-Europe Kft**

About us

- Balabit - Contextual Security Intelligence
- We prevent data breaches without constraining business.
- Less constrained, more monitoring

Zsigmond, Ádám Olivér
adam.zsigmond@balabit.com
Software Engineer | **Balabit-Europe Kft**

Blindspotter - How it is developed?

- Agile software development
  - Incremental improvement
  - Early delivery to customer

Zsigmond, Ádám Olivér
adam.zsigmond@balabit.com
Software Engineer | **Balabit-Europe Kft**

Problem to solve

- Find suspicious activities in a company
- Analyze **users** behaviour
- Alert on unusual **user** behaviour
- Easy product deployment

BALABIT

Zsigmond, Ádám Olivér
adam.zsigmond@balabit.com
Software Engineer | **Balabit-Europe Kft**

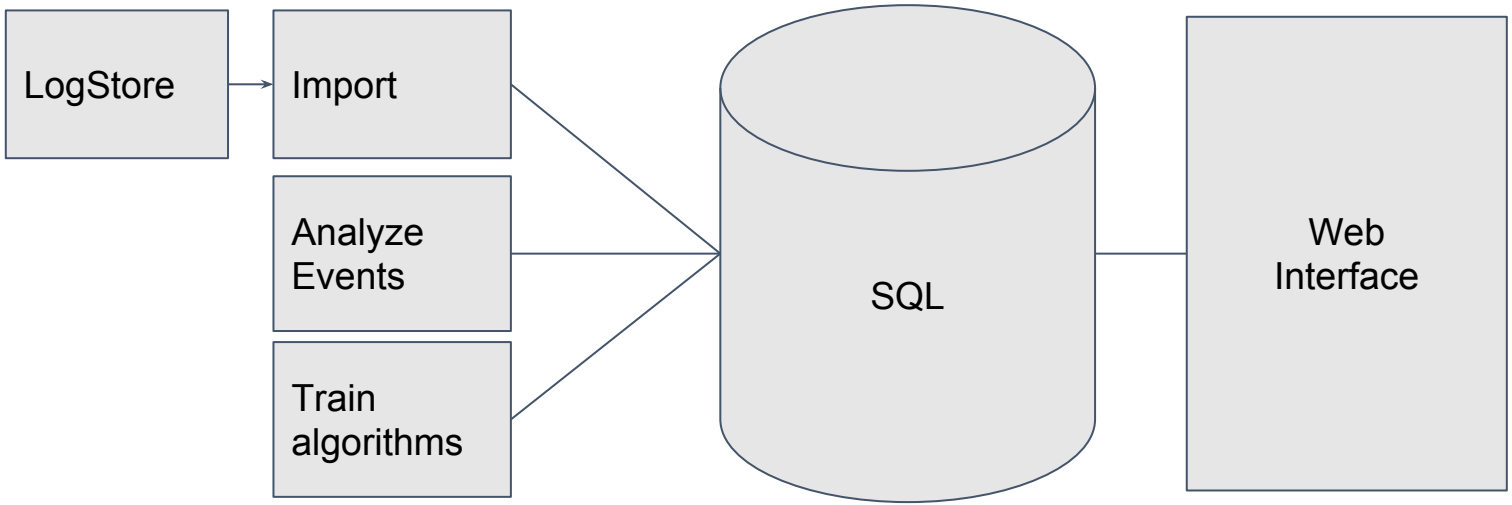
Tools

- Machine Learning
  - Python stack: sklearn, pandas, numpy, scipy
- PostresDB
  - High usage of JSONB columns (postgres 9.4) for storing fields of the logs

BALABIT

Zsigmond, Ádám Olivér
adam.zsigmond@balabit.com
Software Engineer | **Balabit-Europe Kft**

Former Solution



Zsigmond, Ádám Olivér
adam.zsigmond@balabit.com
Software Engineer | **Balabit-Europe Kft**

Former Solution

- Easy testing
- Easy development
- Easy DB export
- Not scalable
- No real push interface
- No real time processing

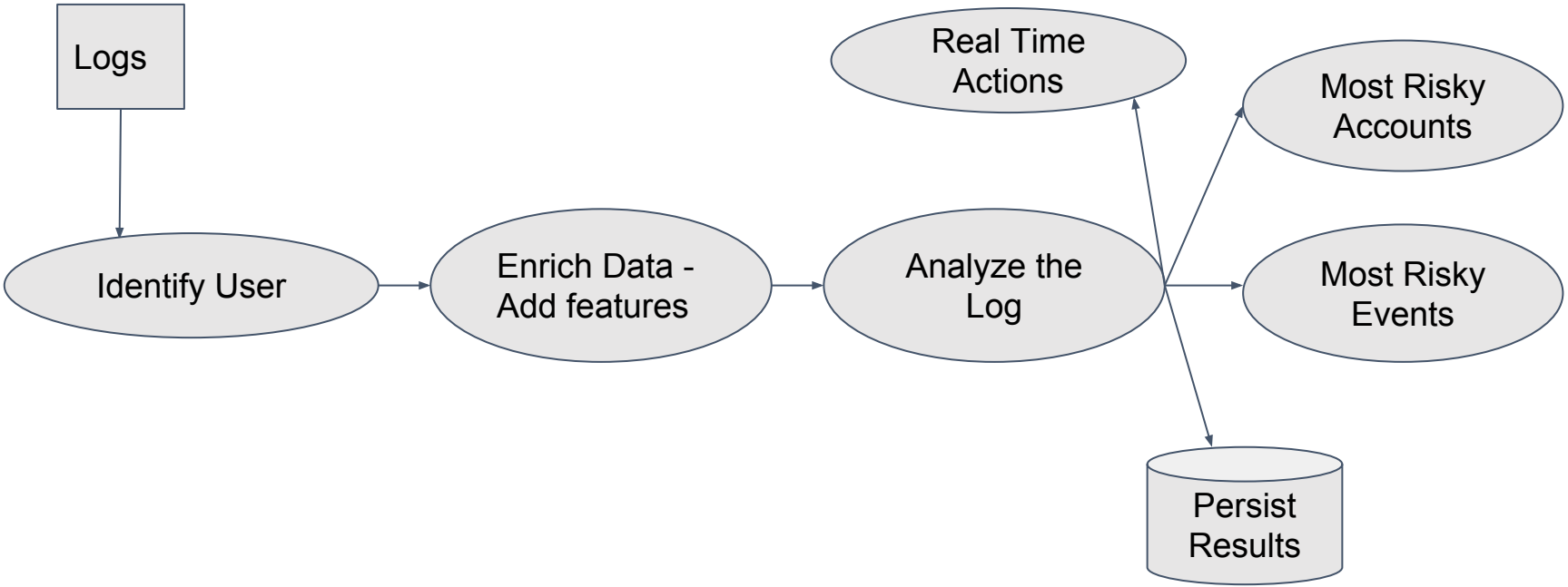
Zsigmond, Ádám Olivér
adam.zsigmond@balabit.com
Software Engineer | **Balabit-Europe Kft**

Problem to solve

- We reached a point, where our architecture failed to handle the data
- **Handle 10 million logs per day (possibly in 8-10 hour) and more...**

Zsigmond, Ádám Olivér
adam.zsigmond@balabit.com
Software Engineer | **Balabit-Europe Kft**

Pipeline to implement



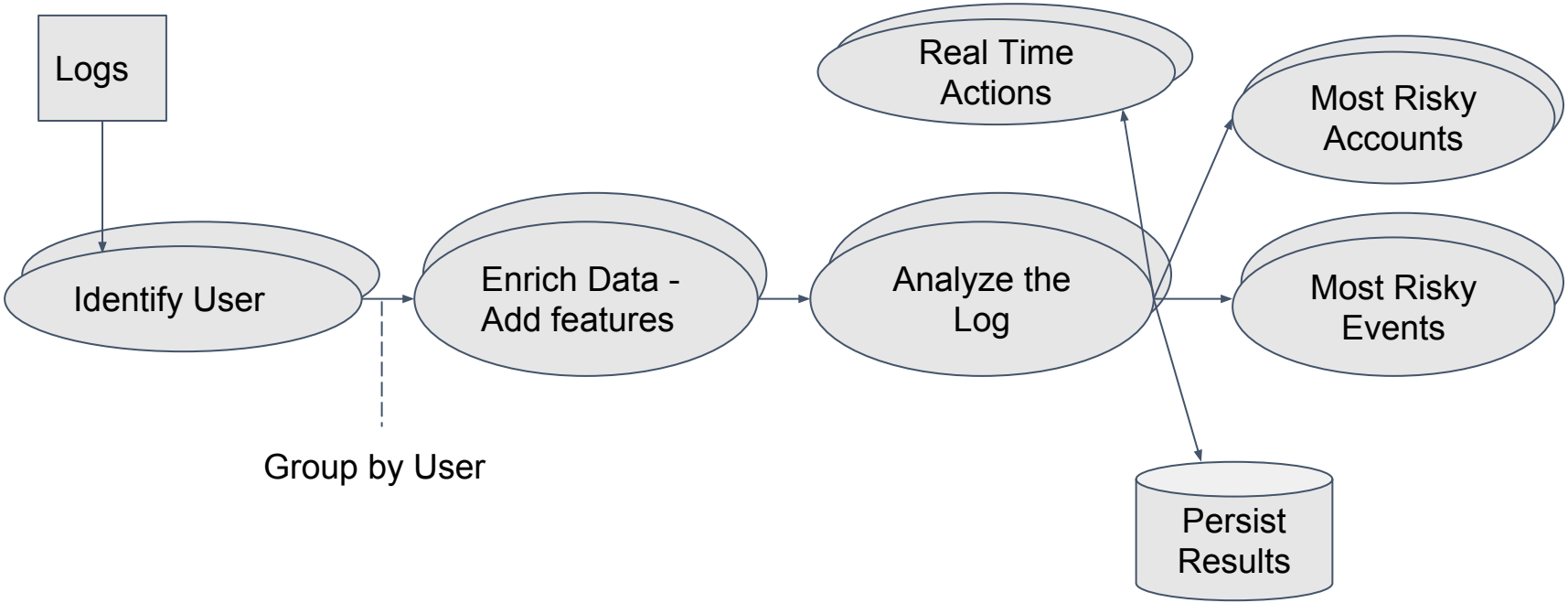
Zsigmond, Ádám Olivér
adam.zsigmond@balabit.com
Software Engineer | **Balabit-Europe Kft**

What is it? Why Stream Processing?

- Stream Processing is made up from pipeline
  - store only the calculated data
  - Combine with persistent message queue
- It can be distributed on the pipeline nodes
- Multiple frameworks available
  - Apache Storm, Apache Flink, …

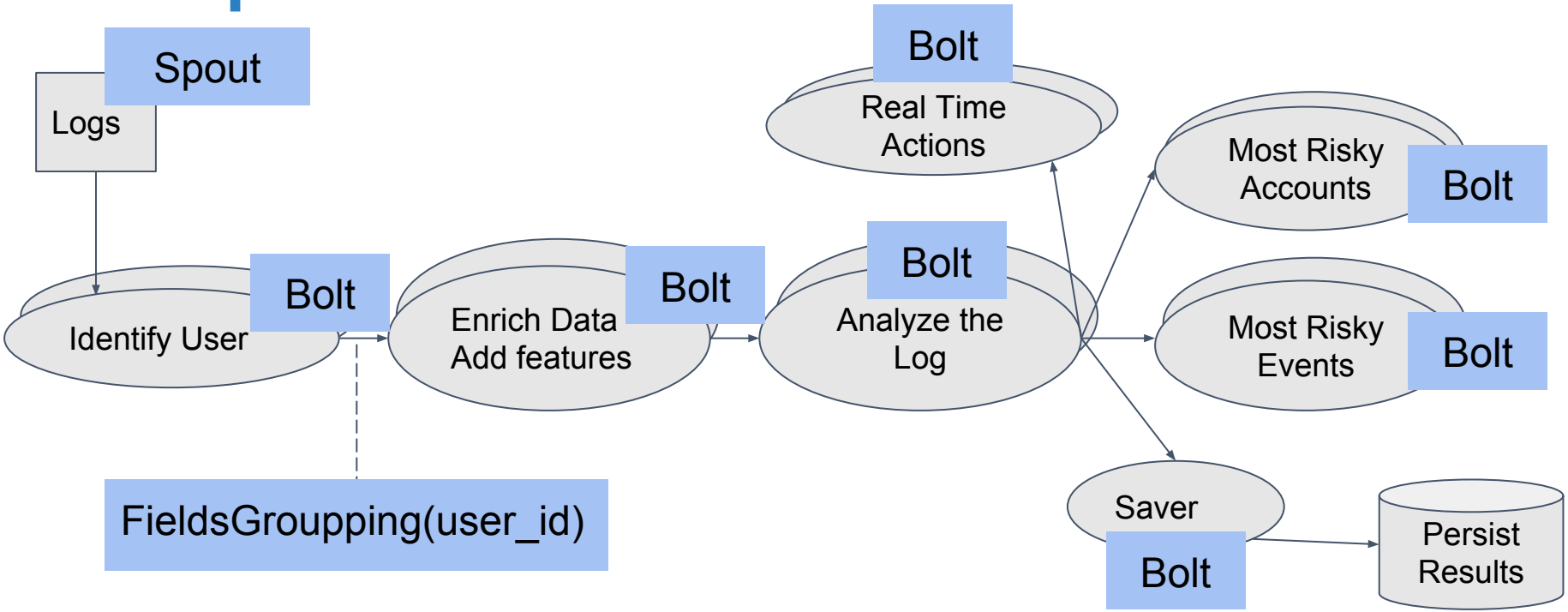
Zsigmond, Ádám Olivér
adam.zsigmond@balabit.com
Software Engineer | **Balabit-Europe Kft**

How does it scale?



Zsigmond, Ádám Olivér
adam.zsigmond@balabit.com
Software Engineer | **Balabit-Europe Kft**

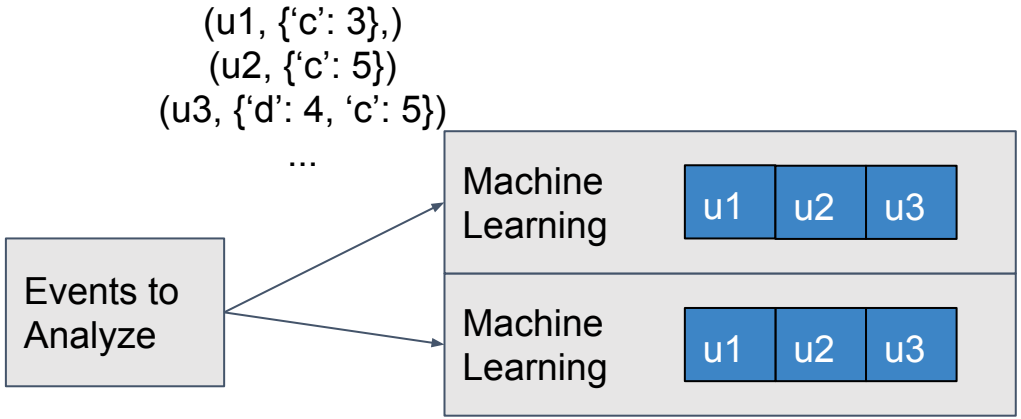
In Apache Storm



Zsigmond, Ádám Olivér
adam.zsigmond@balabit.com
Software Engineer | **Balabit-Europe Kft**

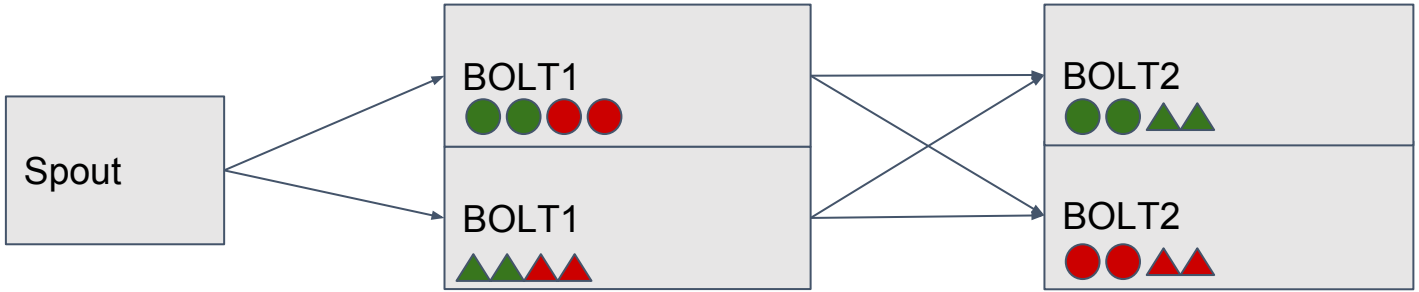
BALABIT

FieldsGrouping



Zsigmond, Ádám Olivér
adam.zsigmond@balabit.com
Software Engineer | **Balabit-Europe Kft**

FieldsGrouping



Zsigmond, Ádám Olivér
adam.zsigmond@balabit.com
Software Engineer | **Balabit-Europe Kft**

Stream Processing Framework

- Low level API is needed
  - For creating python wrapper
  - We need to define own API for our Plugin framework
- At least Once semantics is good enough
- We need minimal state handling in Nodes
  - for the analytics baselines

Zsigmond, Ádám Olivér
adam.zsigmond@balabit.com
Software Engineer | **Balabit-Europe Kft**

Do we need to implement our own?

- Pros:
  - Deployment is easier
  - Single node version, has less overhead without JAVA-CPython communication
- Cons:
  - More work to be done
  - Might mean more bugs

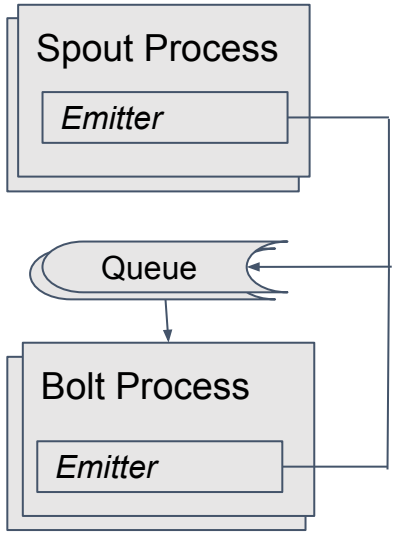
Zsigmond, Ádám Olivér
adam.zsigmond@balabit.com
Software Engineer | **Balabit-Europe Kft**

Only Single Node version

- Wrapper on the API
  - same code can run in our implementation and in Apache Storm
- Learn by doing
  - Lots of experience from implementing our own
- We can get the benefits of both world
  - Easy deployment for first
  - Deploy Storm only if it is needed

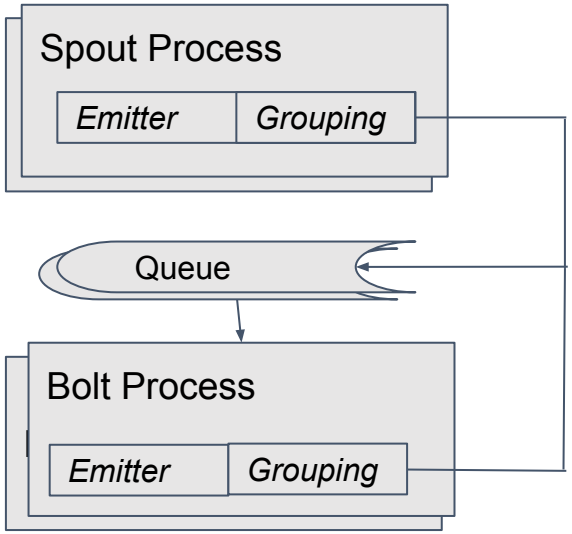
Zsigmond, Ádám Olivér
adam.zsigmond@balabit.com
Software Engineer | **Balabit-Europe Kft**

Components of single node version



Zsigmond, Ádám Olivér
adam.zsigmond@balabit.com
Software Engineer | **Balabit-Europe Kft**

Add Group by key



Zsigmond, Ádám Olivér
adam.zsigmond@balabit.com
Software Engineer | **Balabit-Europe Kft**

Next Problem - Acknowledgment

For every message I sent into my Topology I want to know, when the pipeline has finished processing it.

For this we need:

- Track messages and messages emitted by those messages
- Do not use more memory for every new message
- Get notified, about errors raised by processing a message

Zsigmond, Ádám Olivér
adam.zsigmond@balabit.com
Software Engineer | **Balabit-Europe Kft**

Apache Storm implementation of ACK

- Messages are integers: XOR them with each other
- Each message get XOR-ed 2 times to the first key
- $Y \text{ XOR } X \text{ XOR } X = Y$

Example:

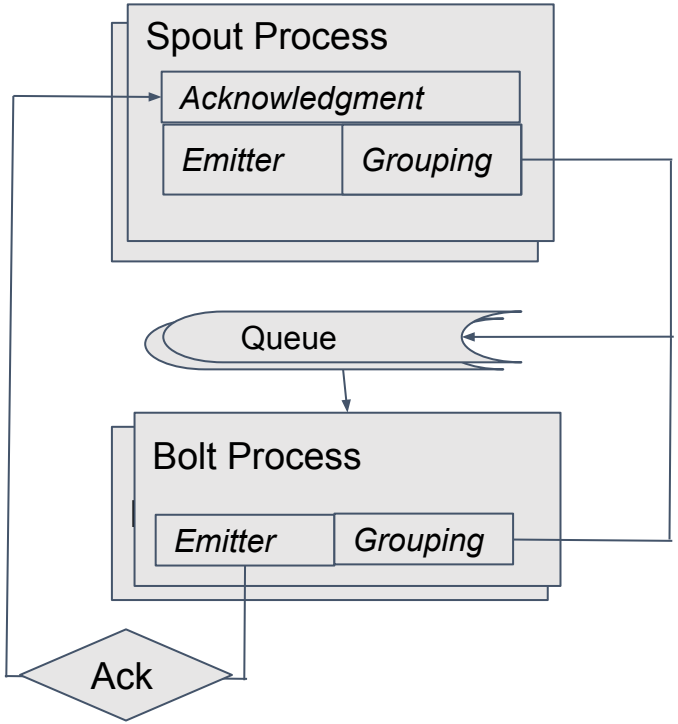
Ids: 10010, 11000, 00101

Ack stream: 10010, 11000, 10010, 00101, 11000, 00101

Track: 10010, 01010, 11000, 11101, 00101, 00000

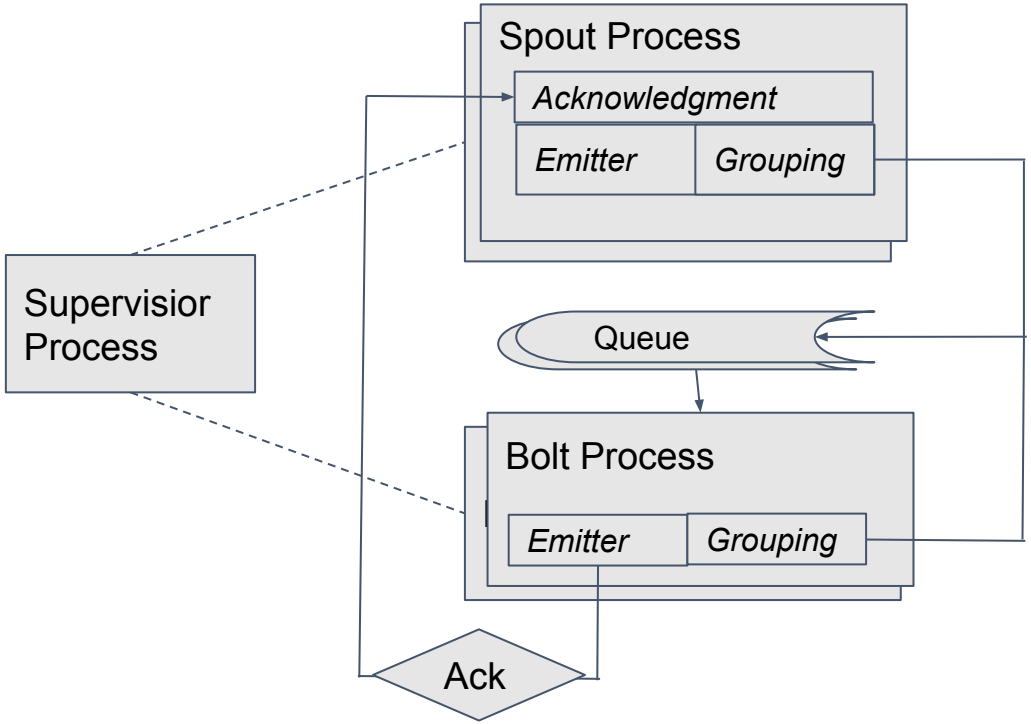
Zsigmond, Ádám Olivér
adam.zsigmond@balabit.com
Software Engineer | **Balabit-Europe Kft**

Track messages - Acknowledgment



Zsigmond, Ádám Olivér
adam.zsigmond@balabit.com
Software Engineer | **Balabit-Europe Kft**

Supervising - `bspctl status`



Zsigmond, Ádám Olivér
adam.zsigmond@balabit.com
Software Engineer | **Balabit-Europe Kft**

How does it look like?

```
activity_stream = tb.register_spout(ActivitySpout)
previous_node = tb.register_bolt(ActivityEnricherBolt) \
    .subscribe(previous_node, FieldGroupping('user_id'))
scored_stream = tb.register_bolt(ActivityScoringBolt)
    .subscribe(previous_node, FieldGroupping('user_id'))

tb.register_bolt(EntityScorerBolt) \
    .subscribe(scored_stream, FieldGroupping('user_id'))
tb.register_bolt(AlertingBolt).subscribe(scored_stream)
tb.register_bolt(ActivitySaverBolt).subscribe(scored_stream)
```

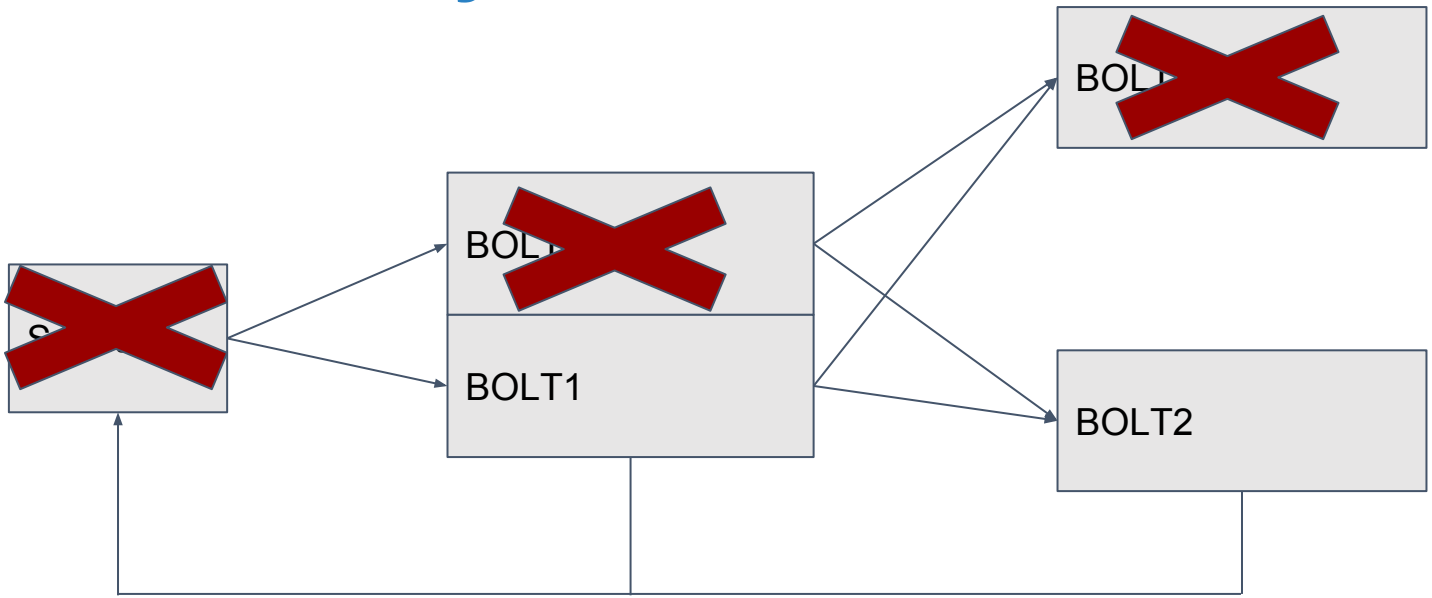

Zsigmond, Ádám Olivér
adam.zsigmond@balabit.com
Software Engineer | **Balabit-Europe Kft**

What is still missing for multiple nodes?

- Resend activities on failures
  - This would result ‘at least once’ semantics
- Spawn process on a different machine
- ReSpawn dead processes

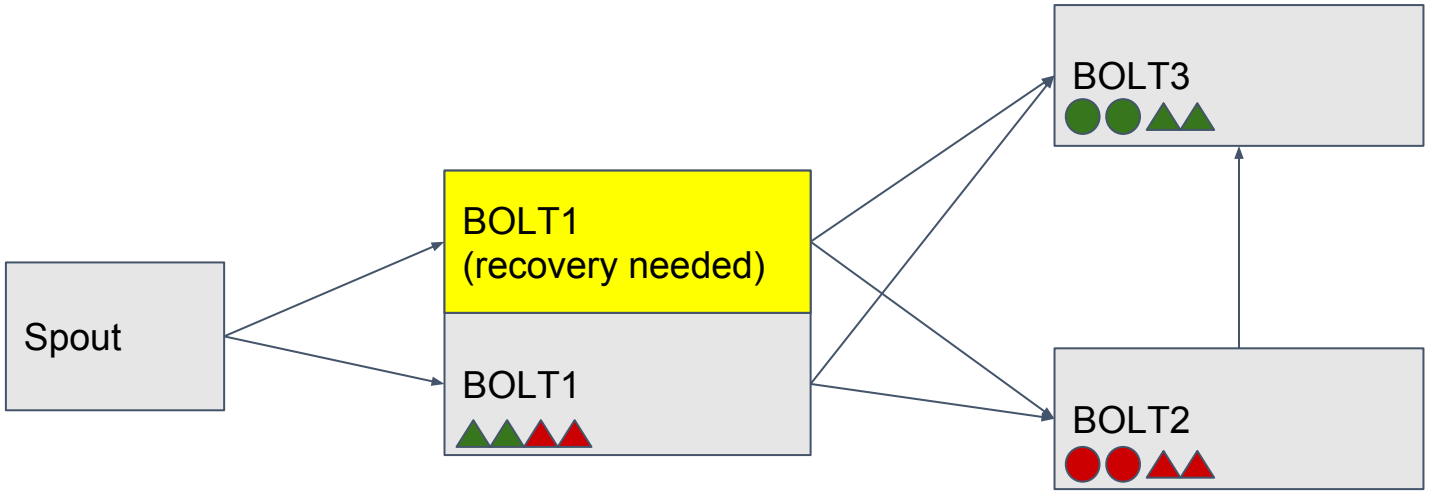
Zsigmond, Ádám Olivér
adam.zsigmond@balabit.com
Software Engineer | **Balabit-Europe Kft**

Why it is difficult when it comes to Distributed System?



Zsigmond, Ádám Olivér
adam.zsigmond@balabit.com
Software Engineer | **Balabit-Europe Kft**

Bolt State Problem



BALABIT

Zsigmond, Ádám Olivér
adam.zsigmond@balabit.com
Software Engineer | **Balabit-Europe Kft**

Some notable problem

- Python multiprocessing.Queue is slow
  - Use SimpleQueue instead, when it is enough
- Python uuid generating was too slow (for message ids)
  - We created some hash function for incrementally create ids
- Redis was really matched the Python semantic
  - It was easy to use for sharing data between processes

Zsigmond, Ádám Olivér
adam.zsigmond@balabit.com
Software Engineer | **Balabit-Europe Kft**

Conclusions

- I would do it again
  - The experience we got is the main value
  - Still think, that JAVA-CPython serialization would be too much overhead
  - It easy to replace the Framework, since we use 2 different
  - 1114 line of python code (with 903 line of tests)

Zsigmond, Ádám Olivér
adam.zsigmond@balabit.com
Software Engineer | **Balabit-Europe Kft**

Conclusions

- Speed is growing with new cores with near to 0.9
  - we got 7.2x faster on a 8 CPU

Zsigmond, Ádám Olivér
adam.zsigmond@balabit.com
Software Engineer | **Balabit-Europe Kft**

Conclusion

- It was not necessarily needed
  - Storm/Flink has their own type of single-node debuggable version for development
  - We would use this if we use JVM based language already

Zsigmond, Ádám Olivér
adam.zsigmond@balabit.com
Software Engineer | **Balabit-Europe Kft**

Questions?

BALABIT

Zsigmond, Ádám Olivér
adam.zsigmond@balabit.com
Software Engineer | **Balabit-Europe Kft**