Better Stream Processing with Python

Taking the Hipster out of Streaming

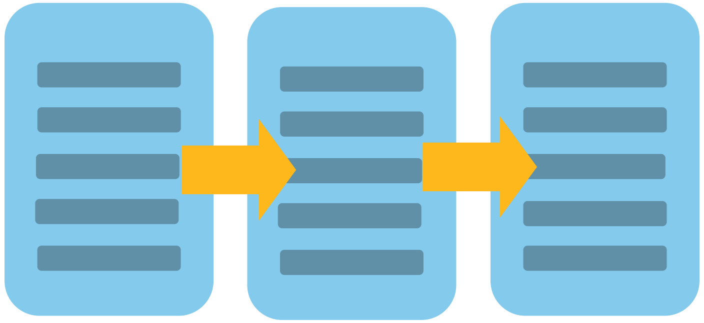
Andreas Heider, Robert Wall

12.07.2017 EuroPython

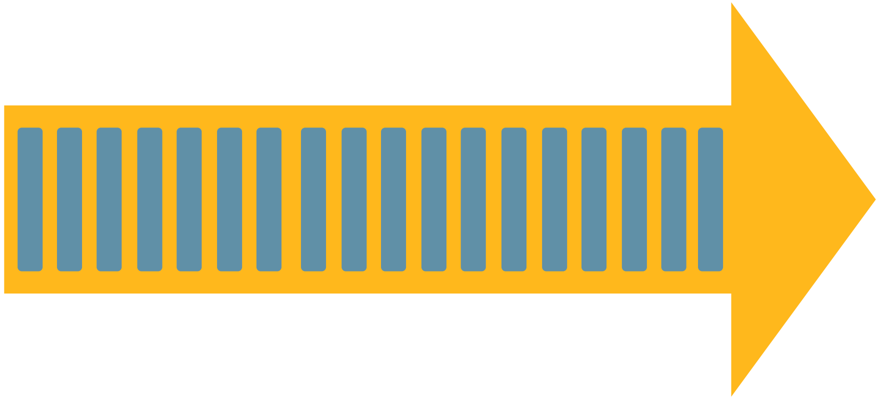
Who are we?

- Developers at Winton
- Winton is a global investment management and data science company, founded in 1997
- We believe the scientific method can be profitably applied to the field of investing

What do we mean by Stream processing?

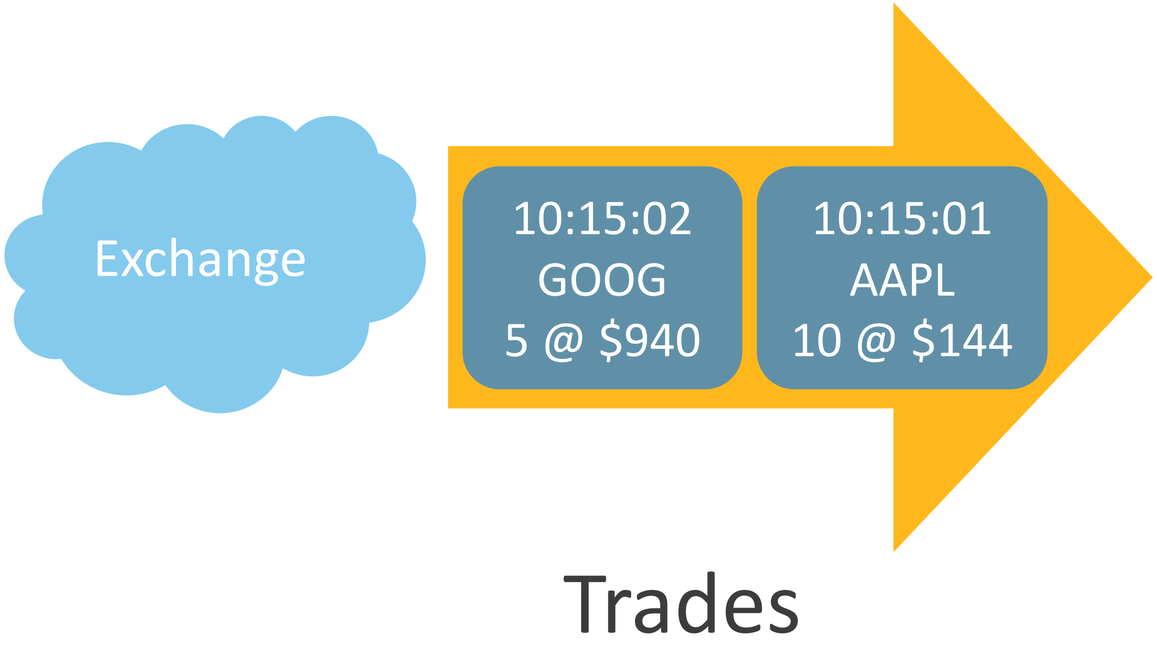


Batch



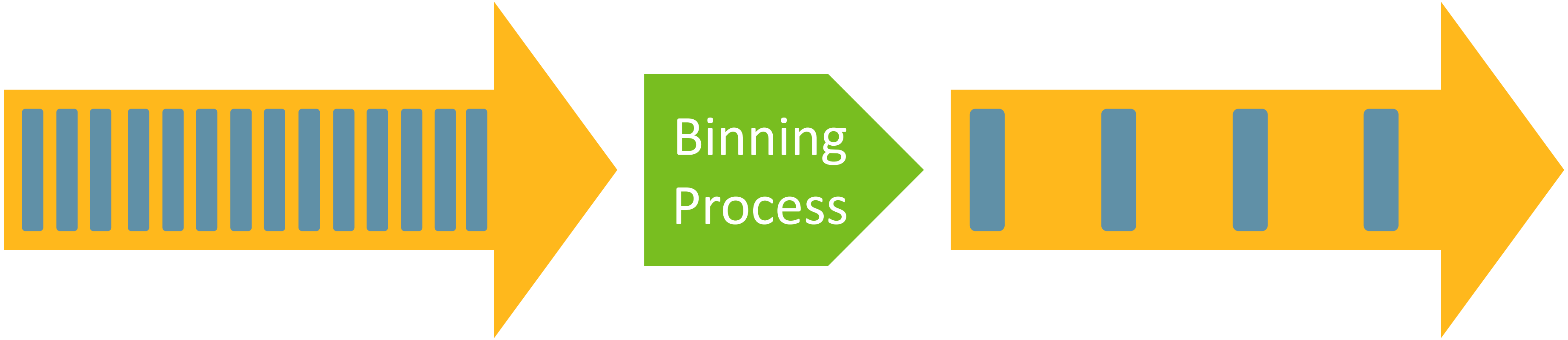
Stream

Example: Real Time Financial Market Data



Time	Symbol	Price	Qty
10:15:01	AAPL	$144	10
10:15:02	GOOG	$940	5
10:15:03	AAPL	$145	11
…			

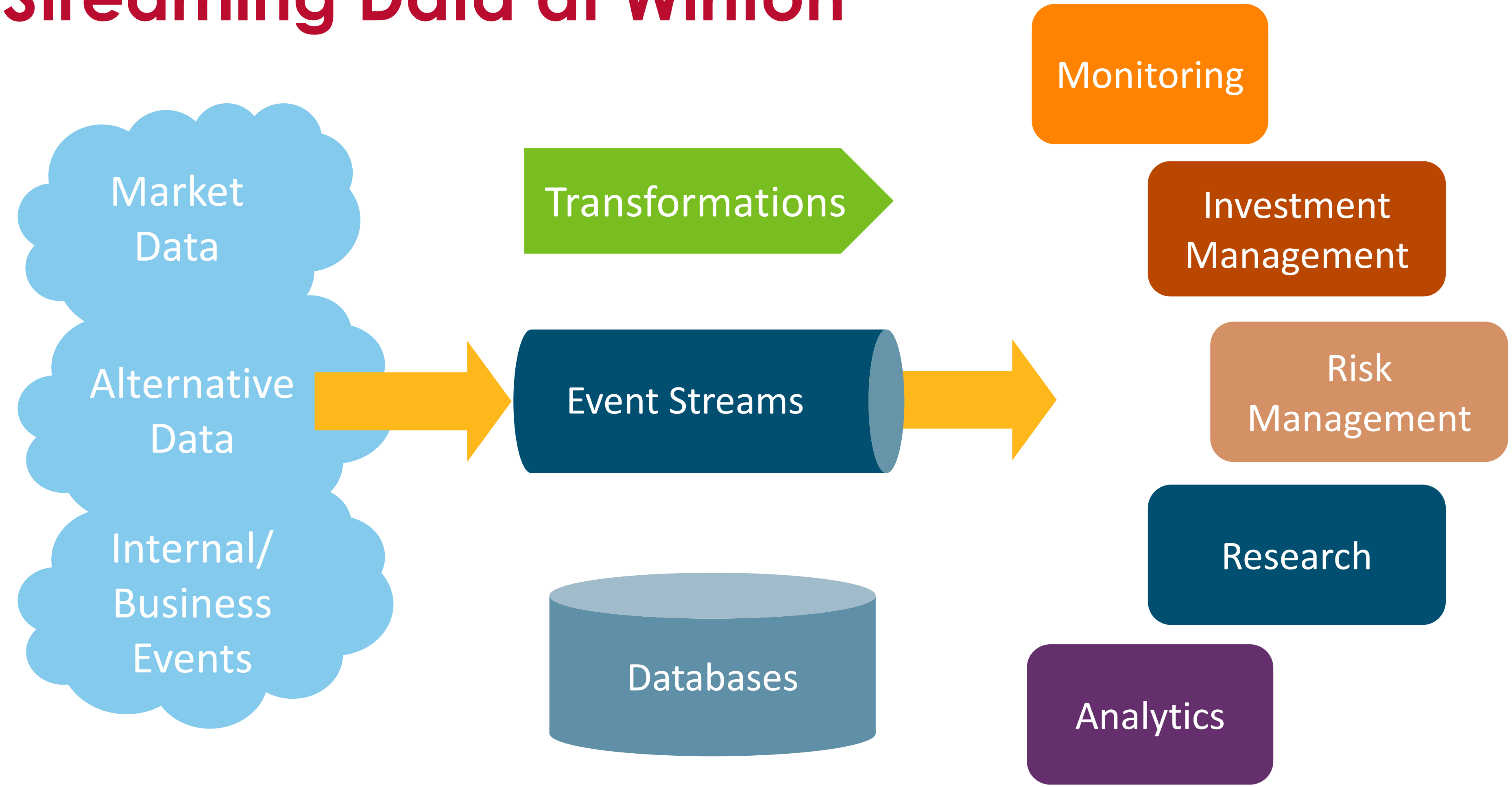
Stream processing: Binning



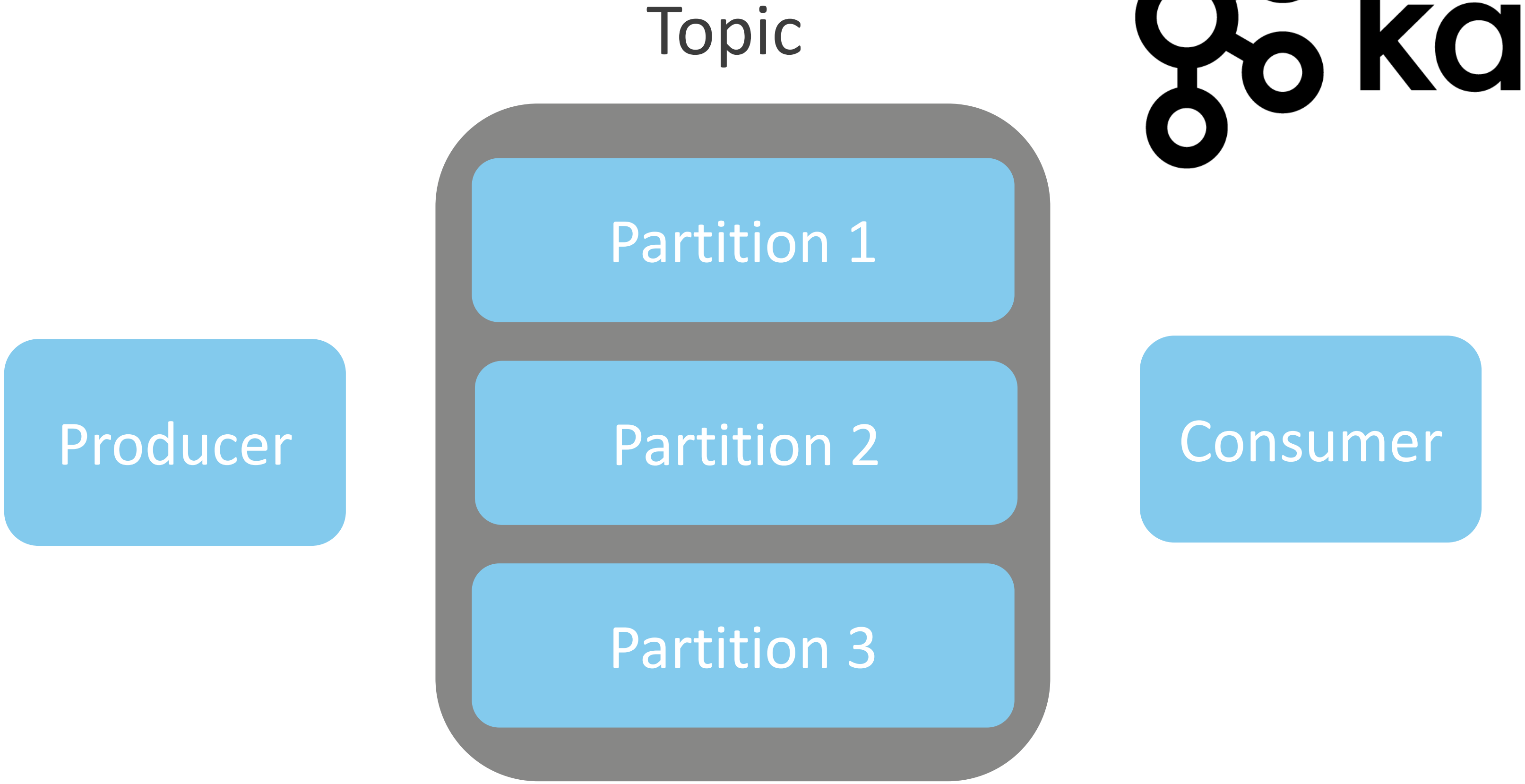
Time	Symbol	Price	Qty
10:15:01	AAPL	$144	10
10:15:02	GOOG	$940	5
10:15:03	AAPL	$145	11
…			

Time	Symbol	Avg. Price	Volume
10:15	AAPL	$144.5	1300
10:15	GOOG	$943	1250
10:16	AAPL	$145.3	1450
…			

Streaming Data at Winton



Apache Kafka



Topic

Producer

Partition 1

Partition 2

Partition 3

Consumer

Sprawl of Stream Processing systems

Flink
beam
APACHE nifi
STORM
Spark Streaming
samza

Kafka Streams

- Simple library, not a framework
- Event at a time stream processing
- Stateful processing, joins and aggregations
- Distributed processing and fault tolerance
- Part of main Apache Kafka project
- Java only so far :(

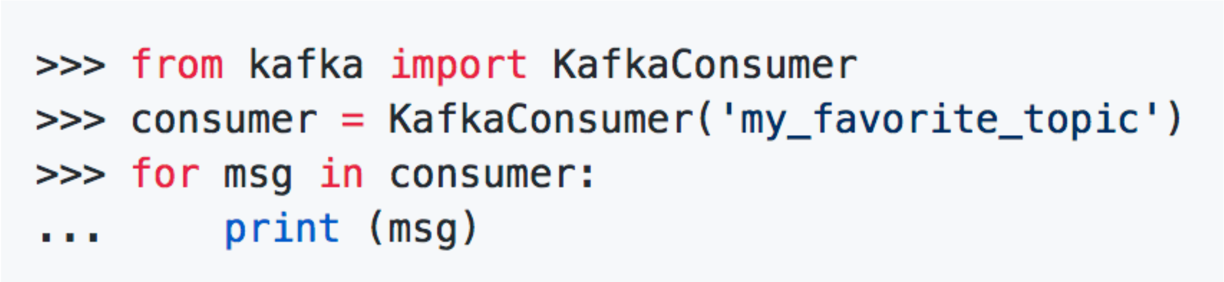
Python at Winton

Many users, with different skillsets:

- Developers
- Researchers
- Operations
- …

Talking to Kafka using kafka-python

Hipster Stream Processing



```
>>> from kafka import KafkaConsumer
>>> consumer = KafkaConsumer('my_favorite_topic')
>>> for msg in consumer:
...     print (msg)
```

Python Kafka Clients

https://github.com/dpkp/kafka-python

- Pure Python implementation
- Friendly, pythonic interface

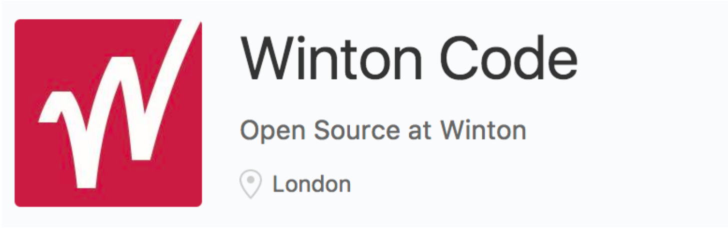
https://github.com/confluentinc/confluent-kafka-python

- Wrapper around C library
- Amazingly high performance and robustness

Experiences using low-level client

- What starts out as a 10 line script ends up as yet another homegrown streaming framework
- The devil is in the details:
  - Guaranteeing at least once (or even exactly-once processing)
  - Handling stateful processing
  - Distributing load over various machines
  - Microbatching
  - Handling rebalances nicely

Kafka Streams for Python



Winton Code

Open Source at Winton

London

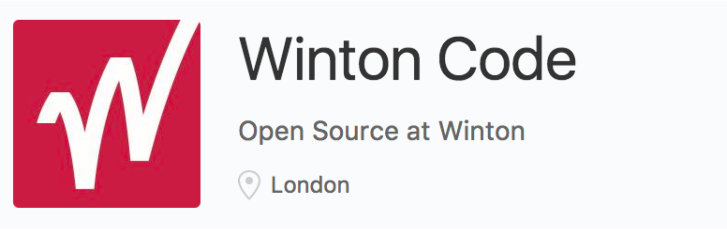
https://github.com/wintoncode/winton-kafka-streams

Demo

Goals / Roadmap

1. Clean implementation of Kafka’s core streams API in Python
2. Experiment with more pythonic API/DSL
3. Optimise performance via batching/numpy/Arrow
4. Implement more advanced features of Kafka’s streams API (exactly once, …)

Get in touch!



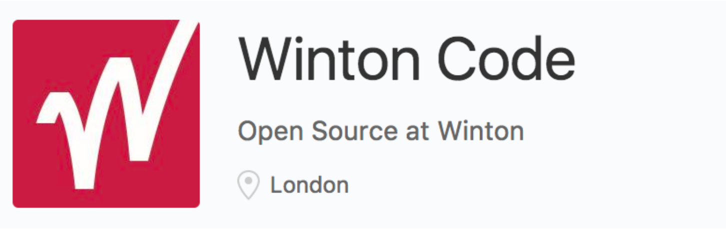
Winton Code

Open Source at Winton

London

- Project on GitHub: https://github.com/wintoncode/winton-kafka-streams
- Roadmap: https://github.com/wintoncode/winton-kafka-streams/blob/master/ROADMAP.md
- Announcement on kafka-dev
- Come to our stand and talk to us
- Thanks to Confluent

Questions?



Winton Code

Open Source at Winton

London

- Project on GitHub: https://github.com/wintoncode/winton-kafka-streams
- Roadmap: https://github.com/wintoncode/winton-kafka-streams/blob/master/ROADMAP.md
- Announcement on kafka-dev
- Come to our stand and talk to us
- Thanks to Confluent

Backup

Some words of experience

- Not everything fits the streaming model
- Manually changing data is tricky
  - Be careful what you put in, have recovery method
- Stable deployment can be challenging
  - Especially Zookeeper and buggy clients
- Set up monitoring from the start
  - We use Prometheus and Grafana
  - https://github.com/yahoo/kafka-manager