Apache Flink® 1.7 and Beyond

公司：dA

职位：Solutions Architect

演讲者：Konstantin Knauf (Slides by Till Rohrmann)

@snntrable

FLINK FORWARD

What is Apache Flink?

Batch Processing
process static and historic data

Data Stream Processing
realtime results from data streams

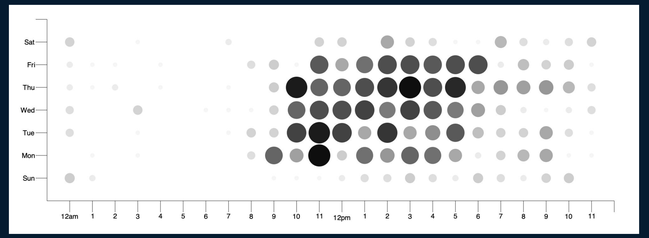
Event-driven Applications
data-driven actions and services

Stateful Computations Over Data Streams

Flink 1.7: What happened so far?

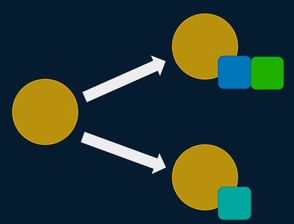
Flink 1.7.0 in Numbers

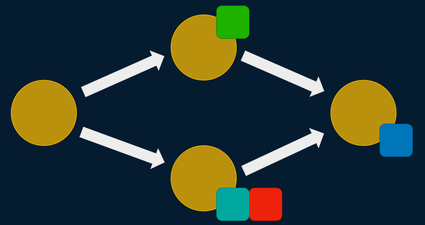
- Contributors: 112
- Resolved issues: 430
- Commits: 970
- Changes LOC: +103824/-63124



Flink Applications Need to Evolve

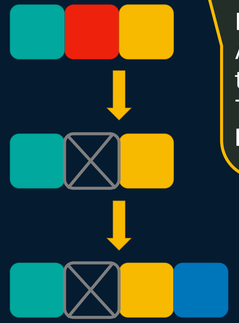
- E.g. changing requirements, new algorithms, better serializers, bug fixes, etc.
- Expensive to restart application from scratch (maintain state)





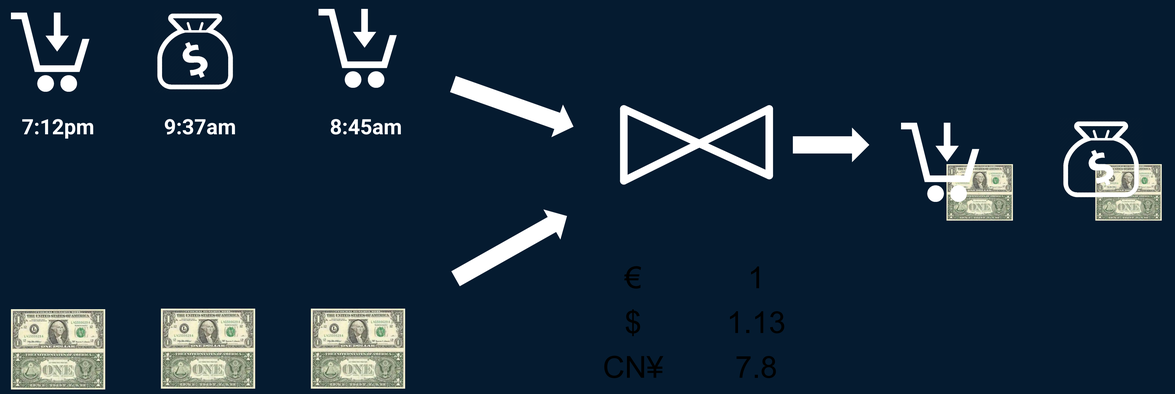
State Schema Evolution

- Support for changing state schema
  - Adding/Removing fields
  - Changing type of fields
- Currently fully supported when using Avro types

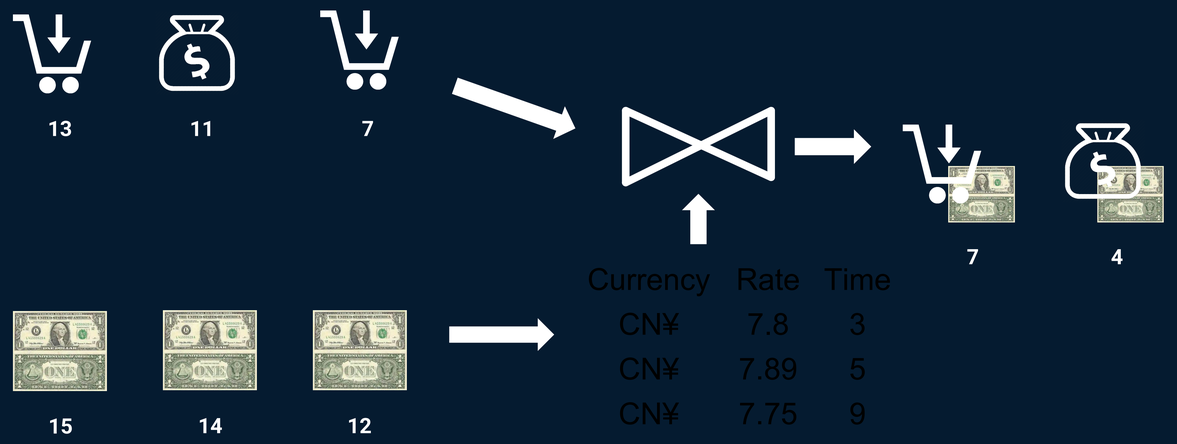


"Upgrading Stateful Flink Streaming Applications: State of the Union" by Tzu-Li Tai Today @ **5:20 pm Room 1**

Converting Currencies

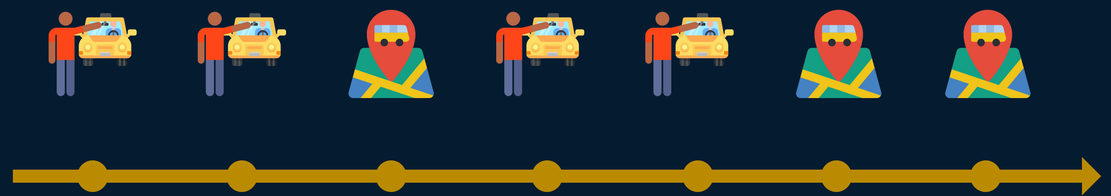


Temporal Tables and Joins



FLINK FORWARD

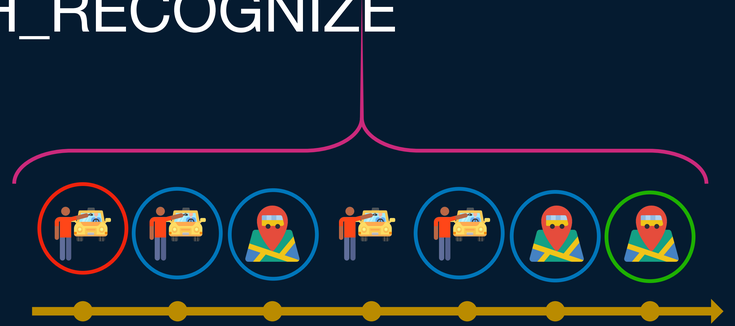
SQL for Pattern Analysis



```
SELECT * from ?
```

MATCH_RECOGNIZE

```
SELECT *
FROM TaxiRides
MATCH_RECOGNIZE (
   PARTITION BY driverId
   ORDER BY rideTime
   MEASURES
       S.rideId as sRideId
   AFTER MATCH SKIP PAST LAST ROW
   PATTERN (S M{2,} E)
   DEFINE
       S AS S.isStart = true,
       M AS M.rideId <> S.rideId,
       E AS E.isStart = false
           AND E.rideId = S.rideId
)
```



More SQL Improvements

- ElasticSearch 6 Table Sink
- Support for views in SQL Client
- More built-in functions: `TO_BASE64`, `LOG2`, `REPLACE`, `COSH`, …

"Flink Streaming SQL 2018" by Piotr Nowojski
Today @ **4:00 pm Room 1**

FLINK FORWARD

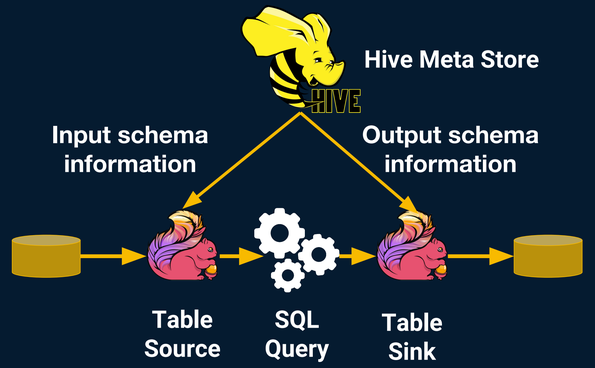
Other Notable Features

- Scala 2.12 Support
- Exactly-once S3 StreamingFileSink
- Kafka 2.0 connector
- Versioned REST API
- Removal of legacy mode

Flink 1.8+: What is happening next?

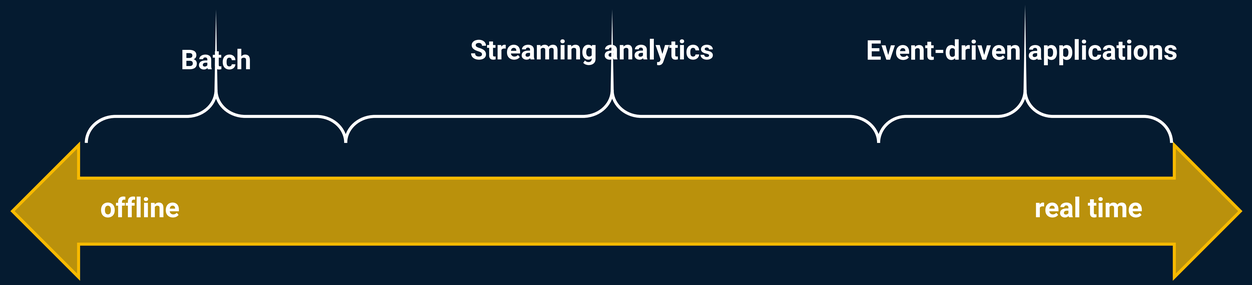
End-to-end SQL Only Pipelines

- Support for external catalogs (Confluent Schema Registry, Hive Meta Store)
- Data definition language (DDL)



FLINK FORWARD

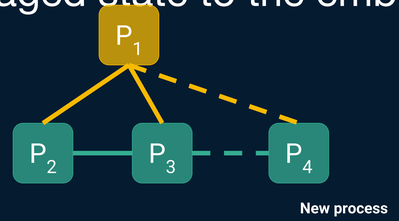
Capability Spectrum



Flink

Flink as a Library

- Deploying Flink applications should be as easy as starting a process
- Bundle application code and Flink into a single image
- Process connects to other application processes and figures out its role
- Expose Flink's managed state to the embedding application

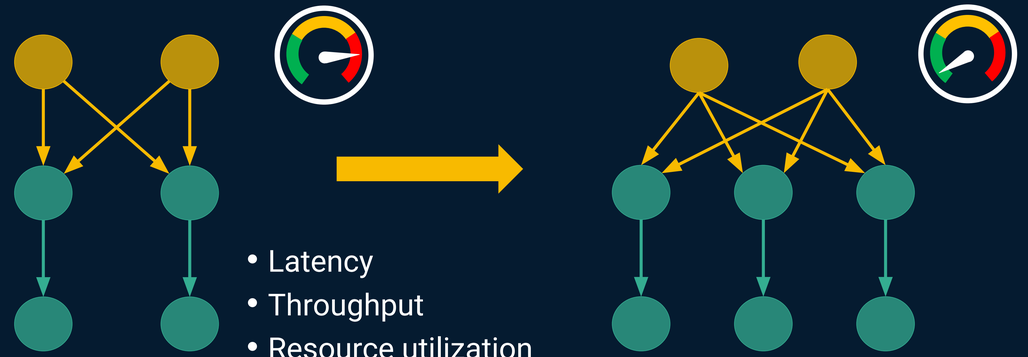


Reactive vs. Active

- Active mode
  - Flink is aware of underlying cluster framework
  - Flink allocate resources
  - E.g. existing YARN and Mesos integration

- Reactive mode
  - Flink is oblivious to its runtime environment
  - External system allocates and releases resources
  - Flink scales with respect to available resources
  - Relevant for environments: Kubernetes, Docker, as a library

Dynamic Scaling



- Latency
- Throughput
- Resource utilization
- Connector signals

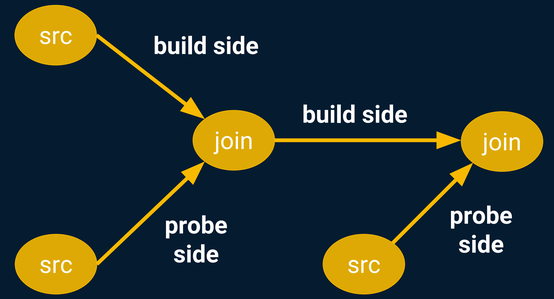
FLINK FORWARD

Batch-Streaming Unification

- No fundamental difference between batch and stream processing
- Batch allows optimizations because data is bounded and ”complete”
- Batch and streaming still separately treated from task level upwards
- Working toward a single runtime for batch and streaming workloads

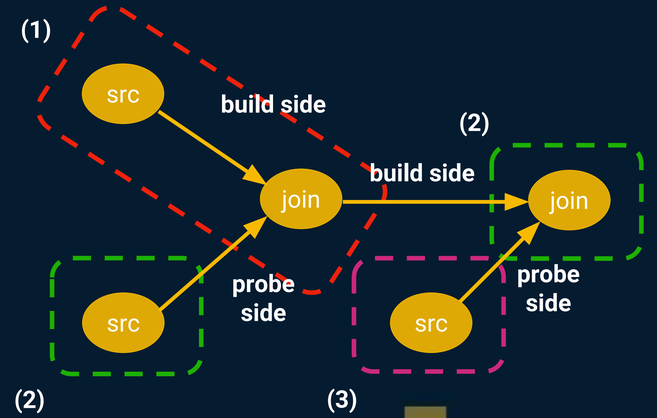
Flink Scheduler

- Lazy scheduling (batch case)
  - Deploy tasks starting from the sources
  - Whenever data is produced start consumers
- Scheduling of idling tasks → resource under-utilization



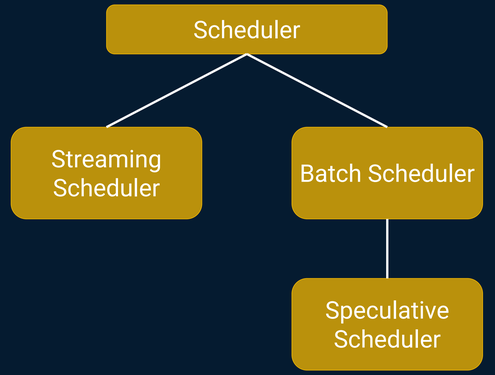
Batch Scheduler

- More efficient scheduling by taking dependencies into account
- E.g. probe side is only scheduled after build side has been processed



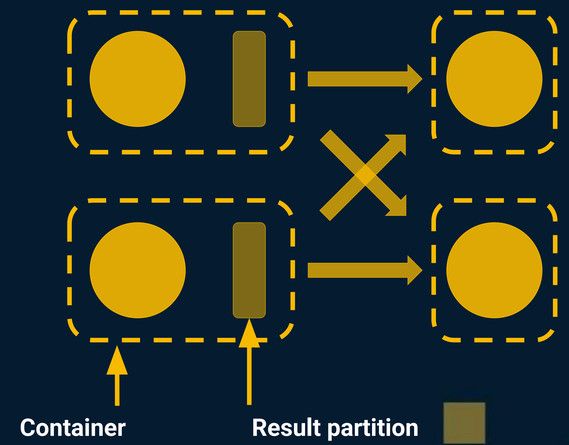
Extendable Scheduler

- Make Flink’s scheduler extendable & pluggable
- Scheduler considers dependencies and reacts to signals from ExecutionGraph
- Specialized scheduler for different use cases



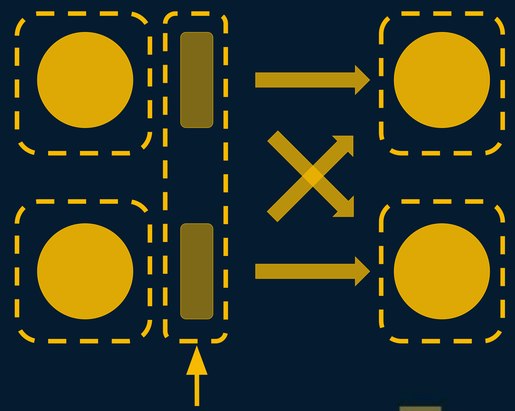
Flink’s Shuffle Service

- Tasks own produced result partitions
- Containers cannot be freed until result is consumed
- One implementation for streaming and batch loads



External & Persistent Shuffle Service

- Result partitions are written to an external shuffle service
- Containers can be freed early
- Different implementations based on use case



FLINK FORWARD

TL;DL

- Flink 1.7.0 added many new features around SQL, connectors and state evolution
- A lot of new features in the pipeline
- Join the community!
  - Subscribe to mailing lists
  - Participate in Flink development
  - Become active

THANKS