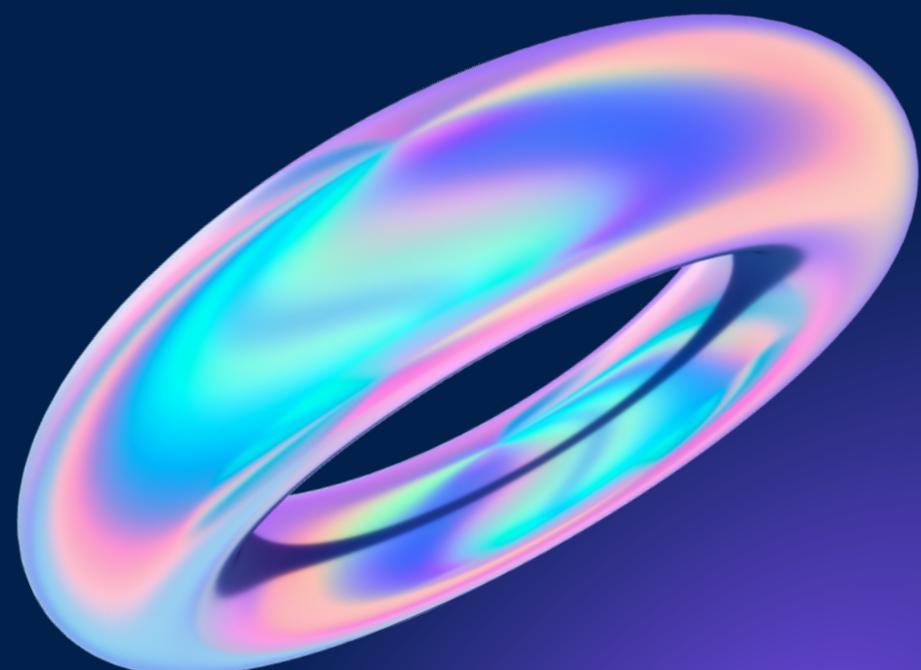




Apache Flink Conceptual Architecture



Team Flink Force

Apache Flink



Powerful and versatile framework
for developing real-time and
batch processing applications

Key Stakeholders

1

Dependents



2

Contributors



3

Apache Software Foundation



Historical Evolution

2010-2014	2015-2016	2016-2021	2021-2023
<p>Inception and Recognition</p> <ul style="list-style-type: none">Apache Flink started as a fork of Stratosphere's distributed execution engine in 2010.Joined Apache incubator in March 2014 and was renamed to Apache Flink	<p>Milestone</p> <ul style="list-style-type: none">2015: Flink Streaming API for real-time data processing was introduced by Apache Flink 0.92016: Flink 1.0 was released which solidified stability and introduced a SQL-like querying API	<p>Performance and Evolution</p> <ul style="list-style-type: none">2016-2018: Performance, fault tolerance, dynamic scaling, and event time processing were prioritized.2018-2021: query optimization, added Python support, and ensuring exactly-once processing semantics	<p>Recent Advances</p> <ul style="list-style-type: none">2021-2023 (Versions 1.15-1.17): Delete and Update API in Flink SQL for batch processing and the “Gateway mode” for SQL Client interaction.

Contribution

Github

Development discussion platform for developers.

JIRA

A systematic approach to problem resolution through issue tracking

Stack Overflow

Q & A Forum for common questions

Wiki Documentation

A resource for users to deepen their understanding

Mentorship

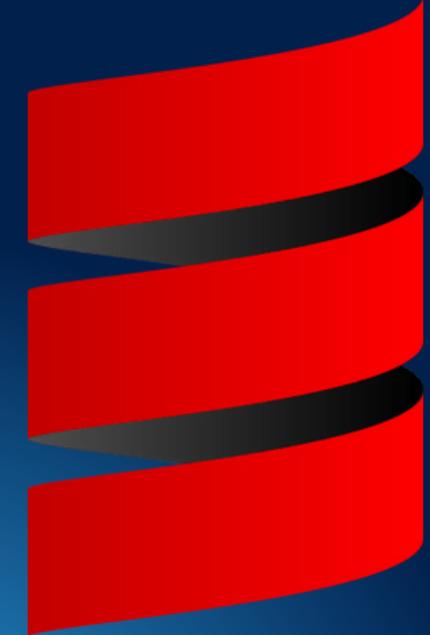
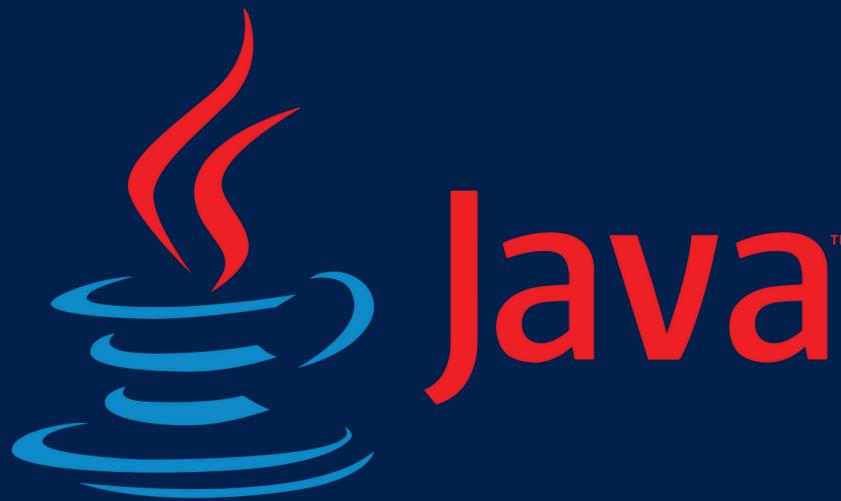
Mentorship and support program for newcomers.



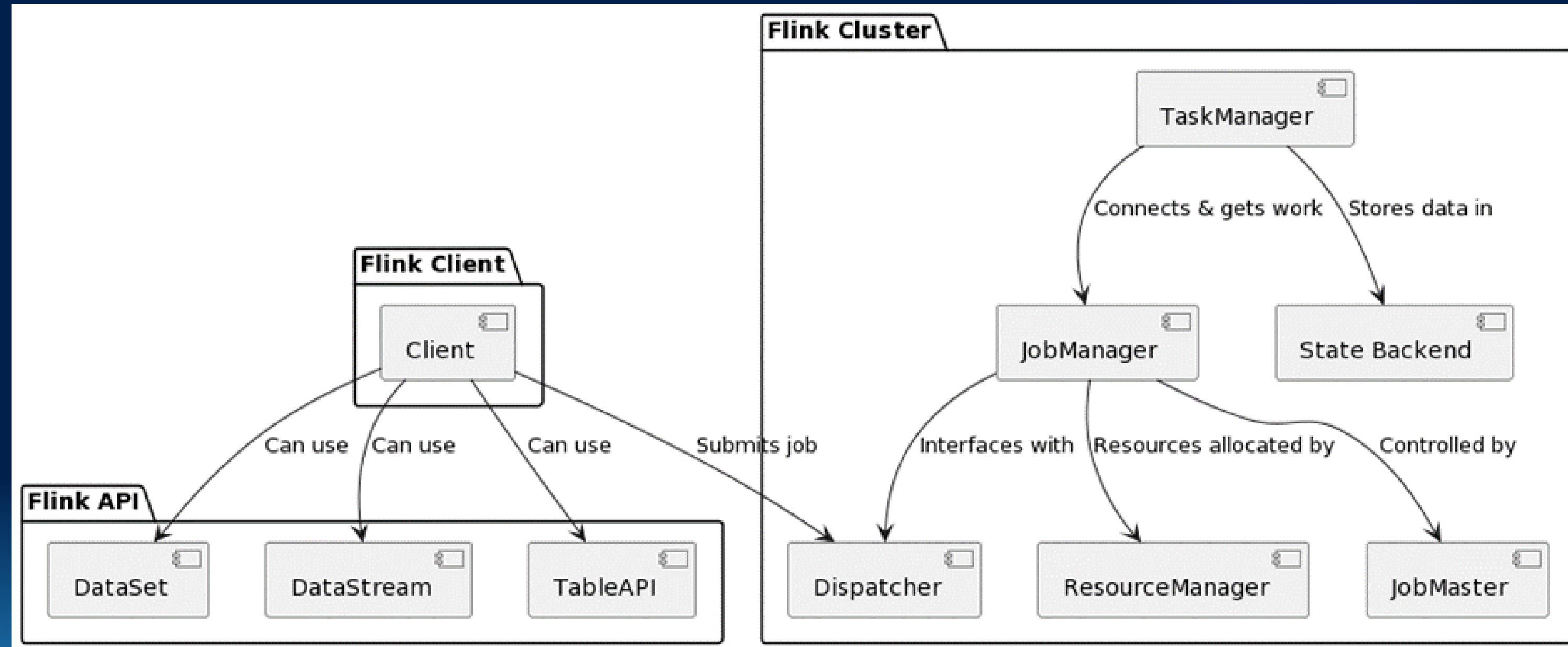
Architecture of Apache Flink & Component Analysis

Technology Stack

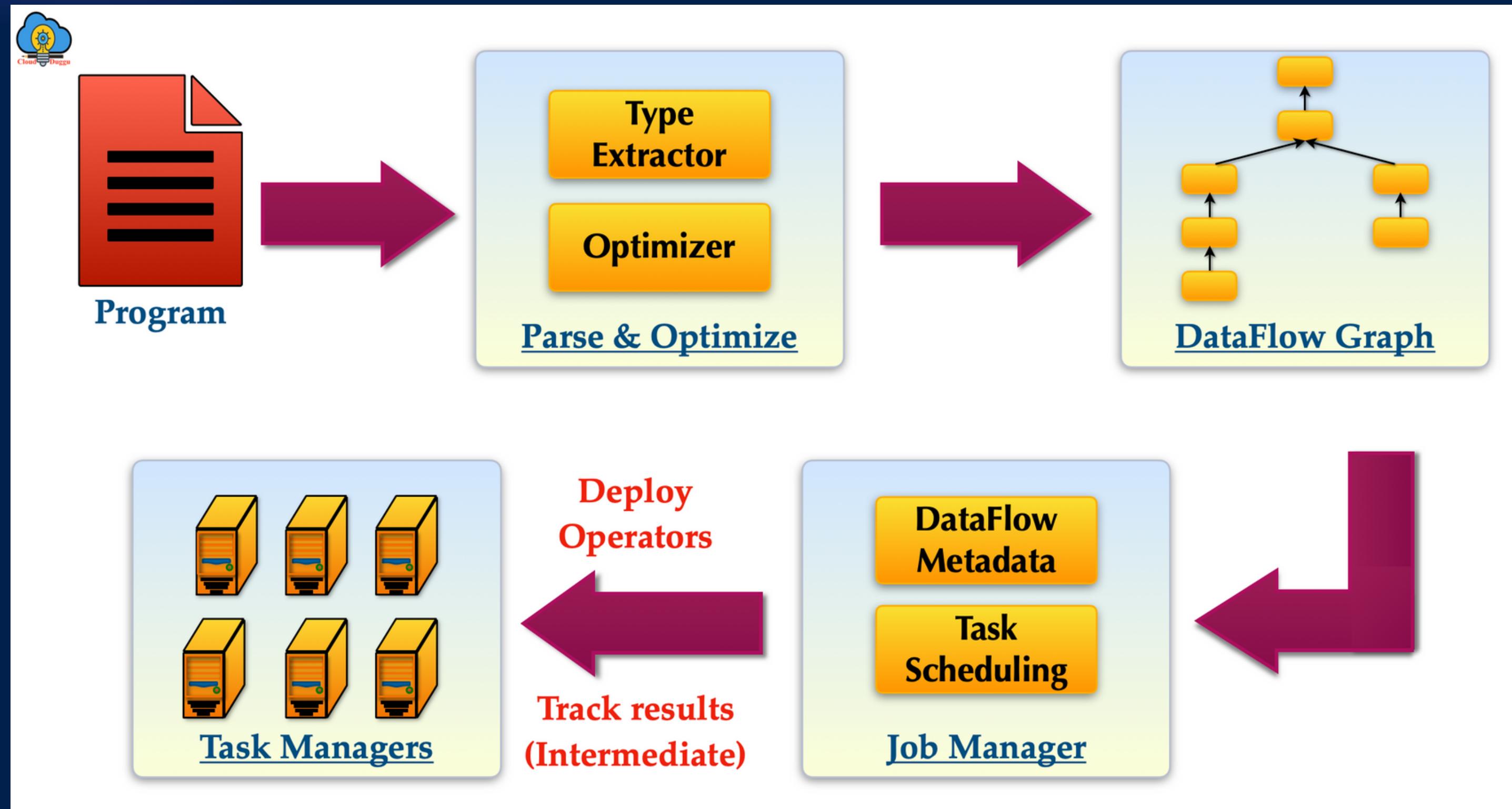
- Core languages
 - Java (85.9%)
 - Scala (9.9%)
 - Python (2.8%)
 - Typescript (0.4%)
 - Other (1.4%)



Component Breakdown Client-Server Interactions



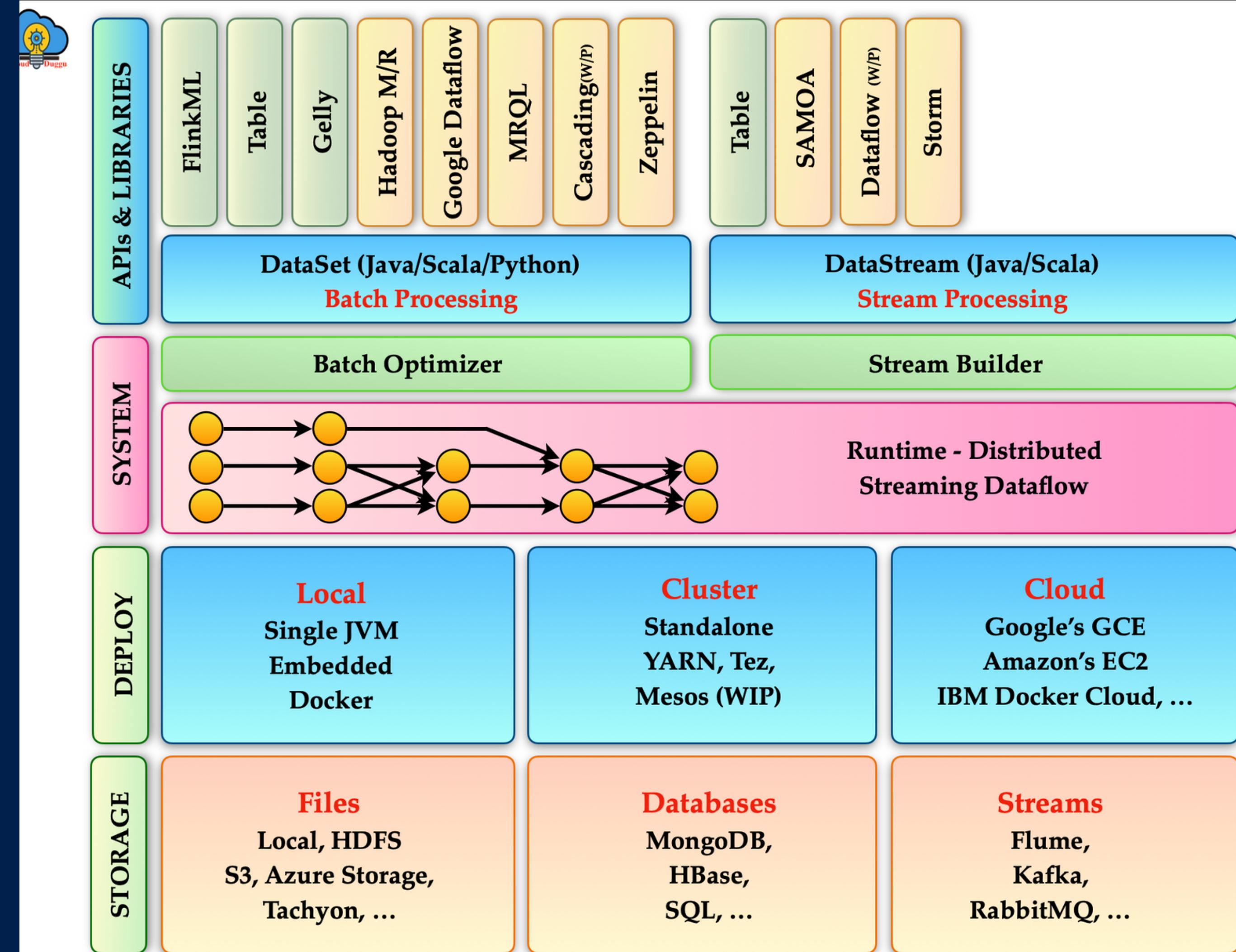
Inter-Component Data Flow



Architectural Styles

- Layered - Core Technology Stack
- Distributed - Scalability & Storage
- Repository - Durable State Storage
- Pipe & Filter - Batch & Stream Processing
- Client Server - Server Based System
- Implicit Invocation - Event Driven

Layered System “Tech Stack”



Flink Web Dashboard

Apache Flink Dashboard

Version: 1.13.1 | Commit: a7f3192 @ 2021-05-25T12:02:11+02:00 | Message: 0

Streaming WordCount | FINISHED | 2

ID: 1d5f6e3c409182d79fd244a9c69410e1 | Start Time: 2021-06-30 13:18:59 | End Time: 2021-06-30 13:19:00 | Duration: 565ms

Overview Exceptions Timeline Checkpoints Configuration

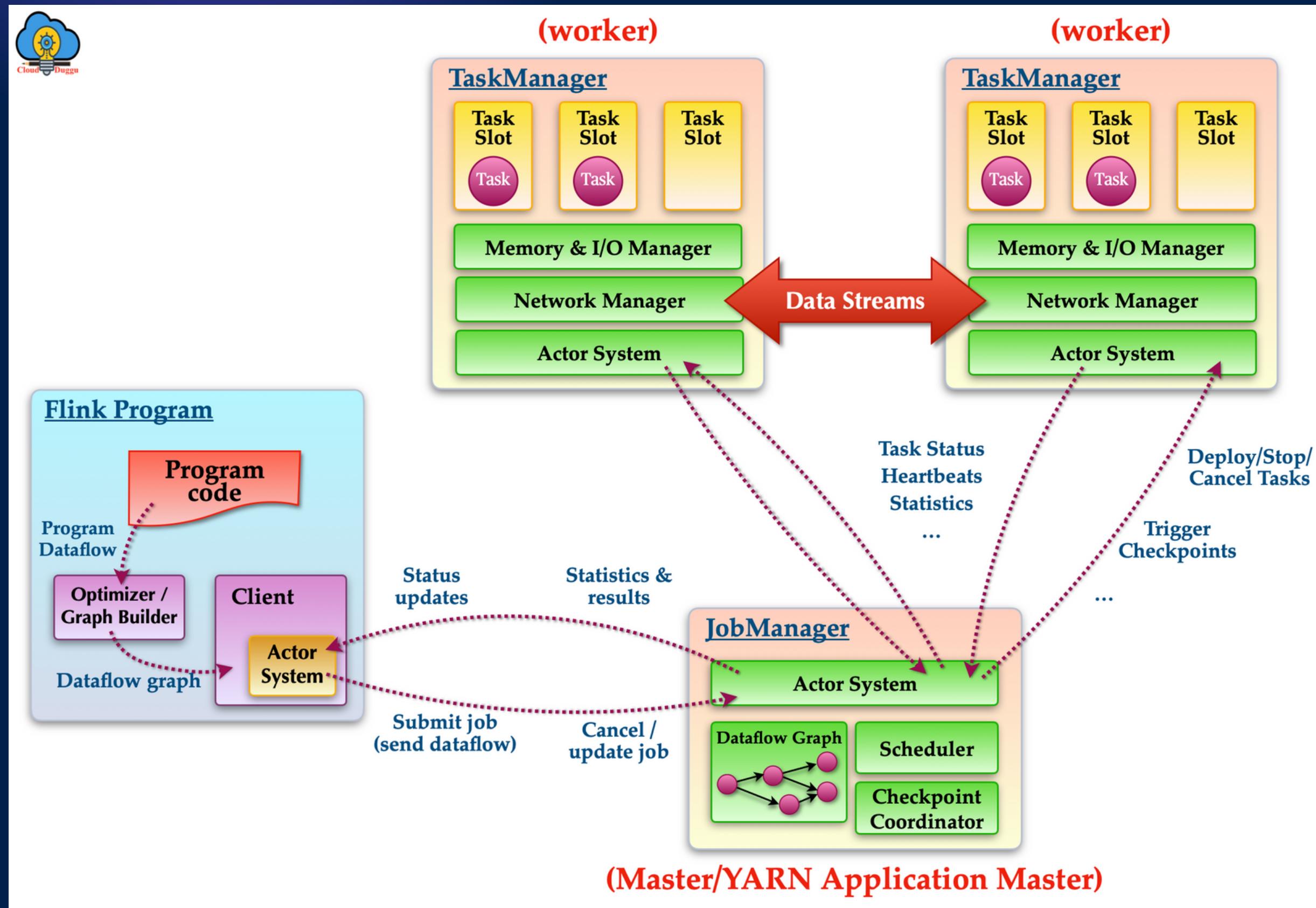
Source: Collection Source -> Flat Map
Parallelism: 1
Backpressured (max): N/A
Busy (max): N/A

Keyed Aggregation -> Sink: Print to Std. Out
Parallelism: 1
Backpressured (max): N/A
Busy (max): N/A

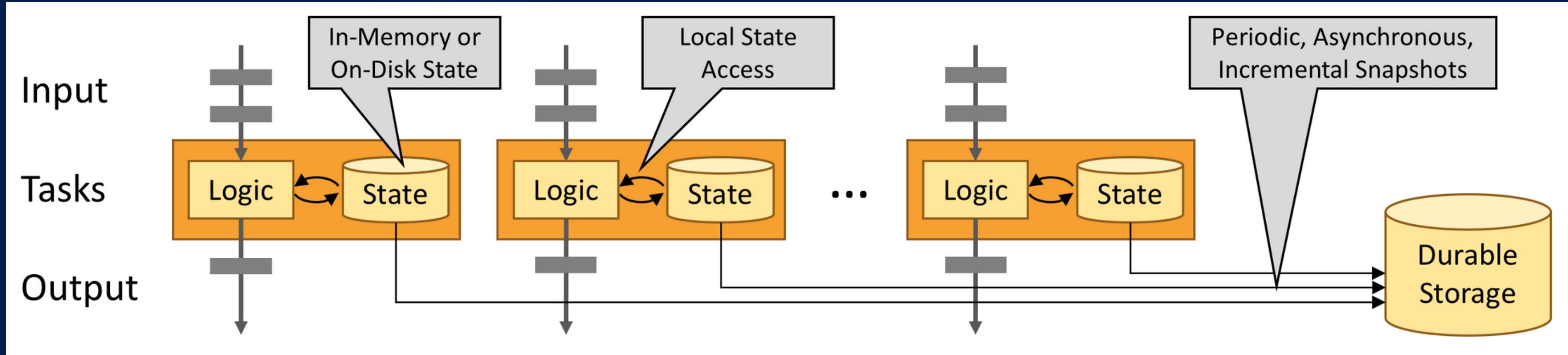
HASH

Name	Status	Bytes Received	Records Received	Bytes Sent	Records Sent	Parallelism	Start Time	Duration	End Time	Tasks
Source: Collection Source -> Flat Map	FINISHED	0 B	0	3.95 KB	287	1	2021-06-30 13:18:59	285ms	2021-06-30 13:19:00	1
Keyed Aggregation -> Sink: Print to Std. Out	FINISHED	3.96 KB	287	0 B	0	1	2021-06-30 13:18:59	291ms	2021-06-30 13:19:00	1

Distributed System

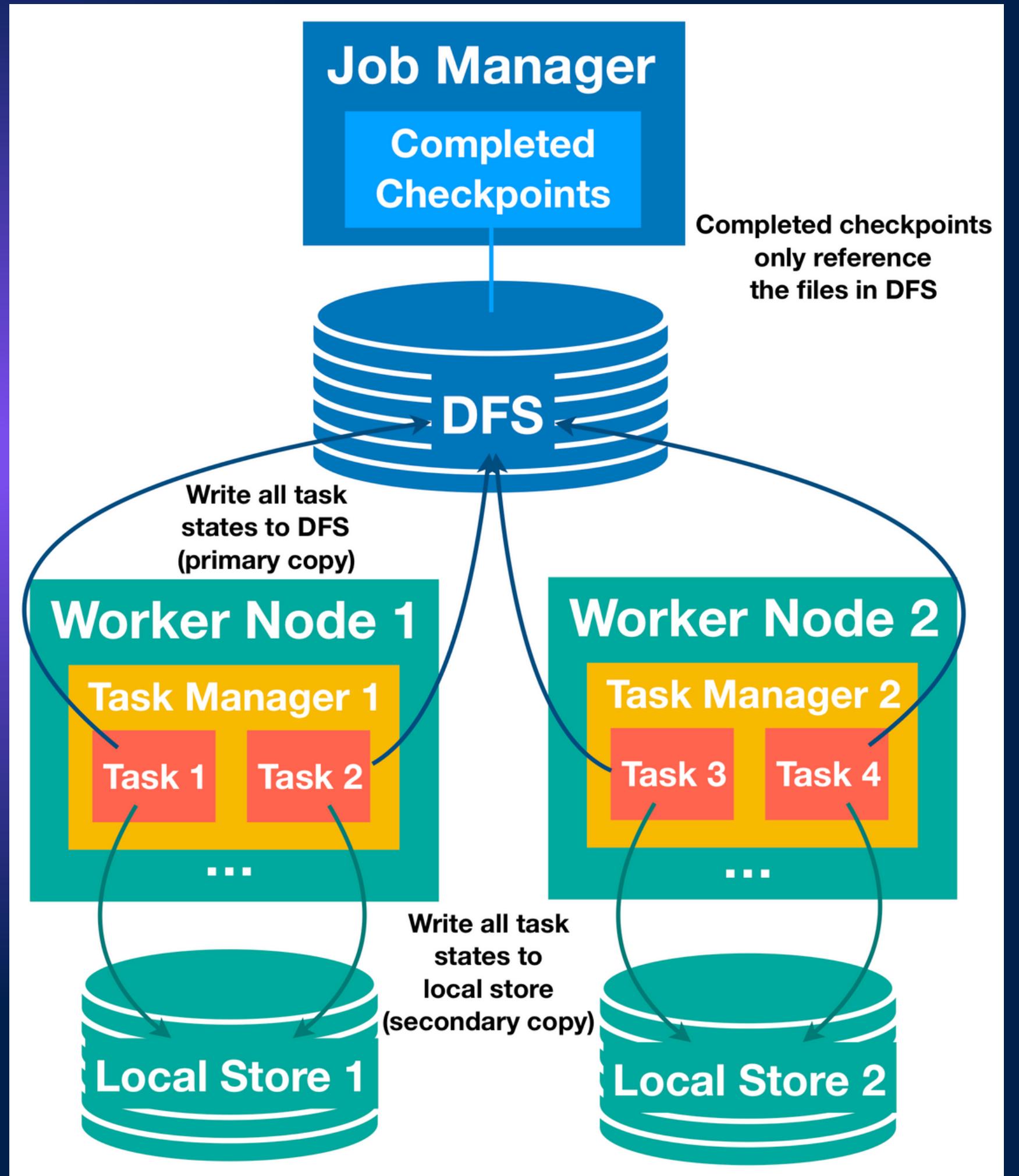


Repository



Guarantees “exactly-once” state consistency

- use of in memory state snapshots
- specialized disk data structures for backups

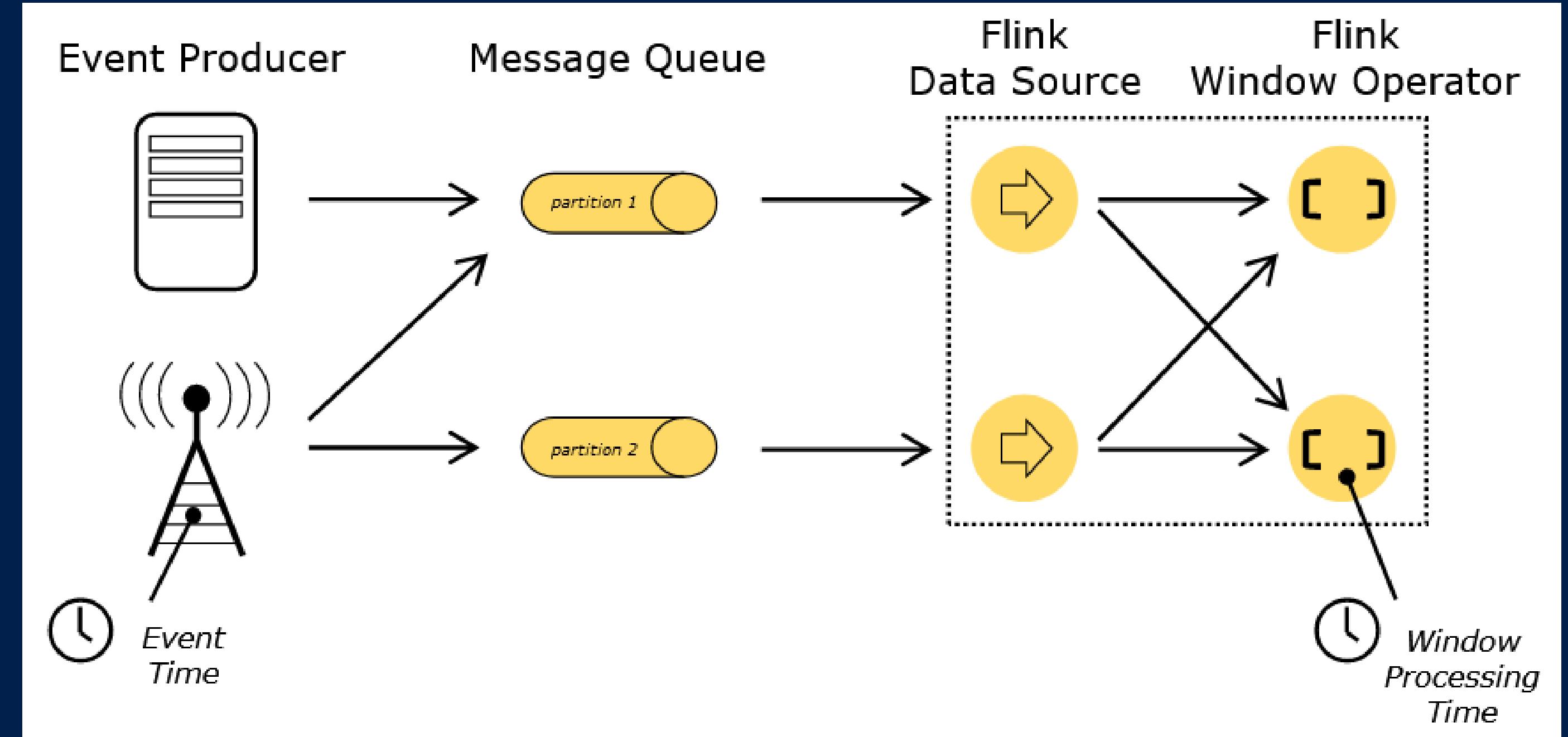


Distributed Storage & Execution

+

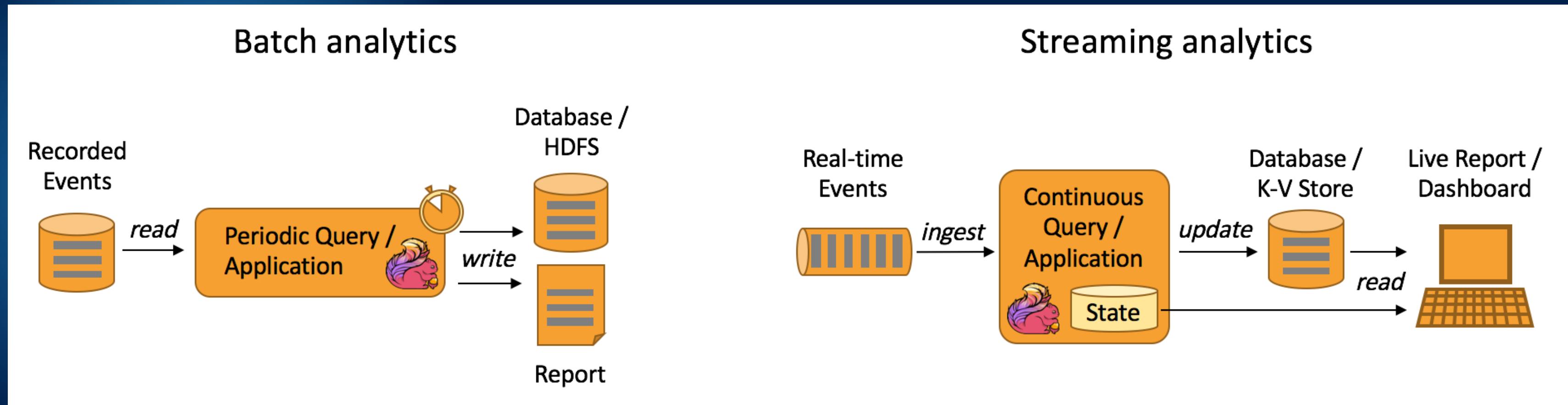
Repository

Pipe & Filter

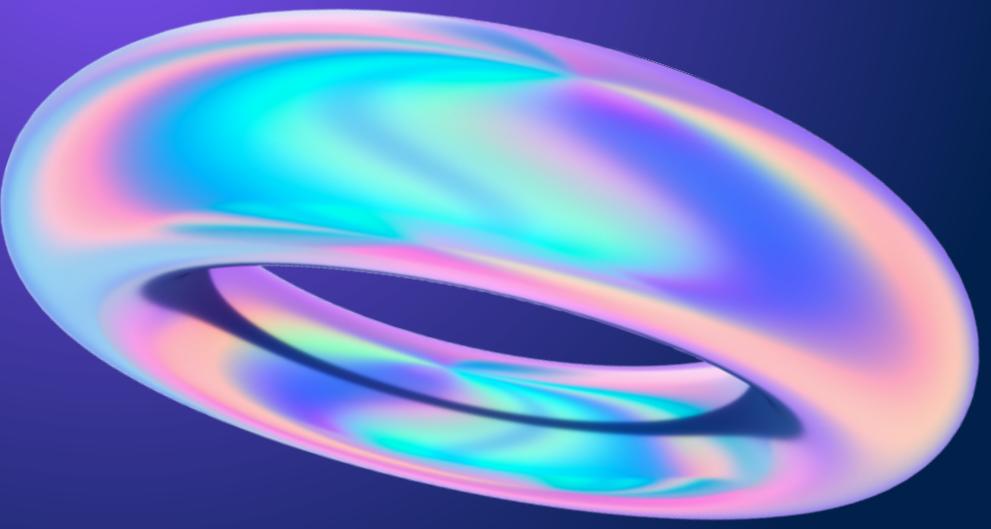


Flink serves as the component for applications to stream data from any sources to any sink

Implicit Invocation - Event Driven

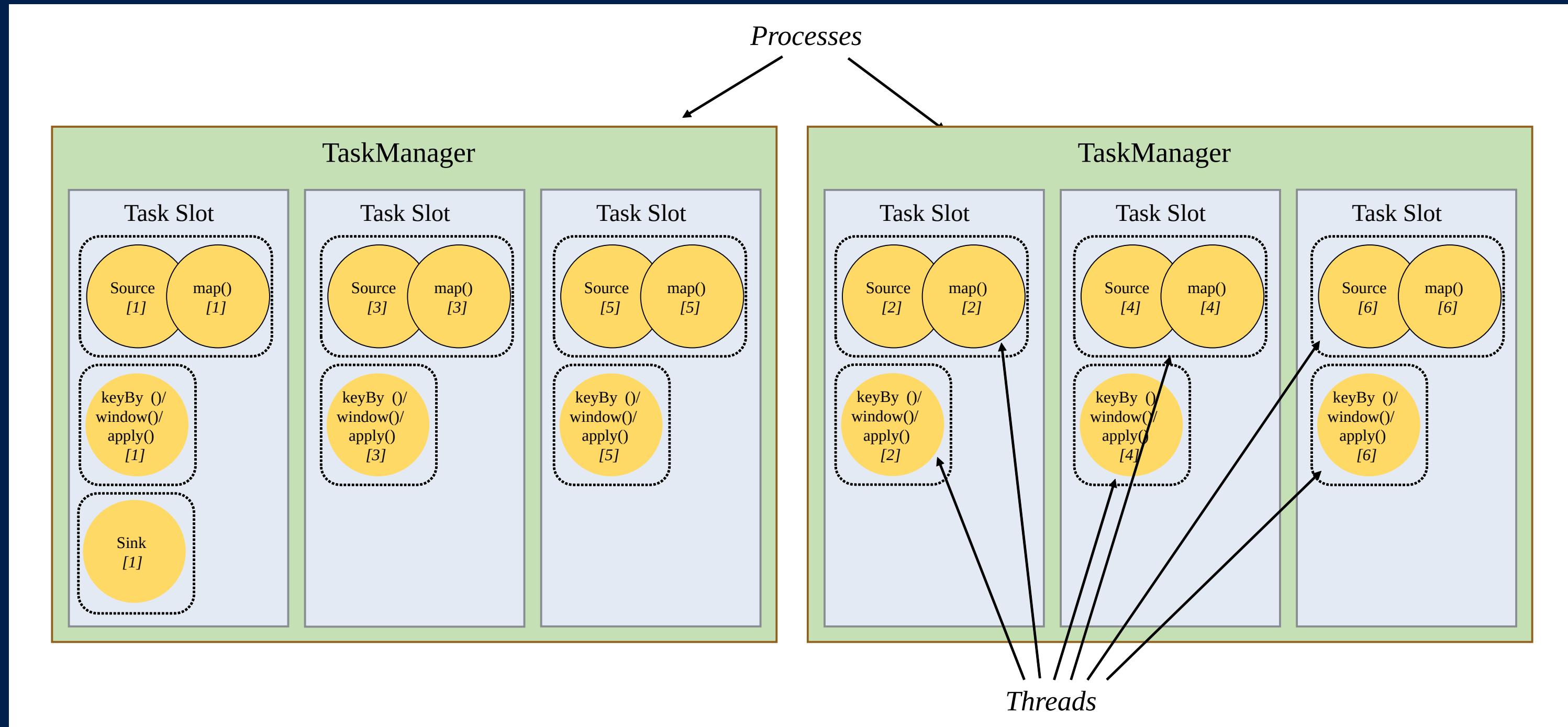


Flink applications respond to events in a stateful manner and have specific time-based semantics

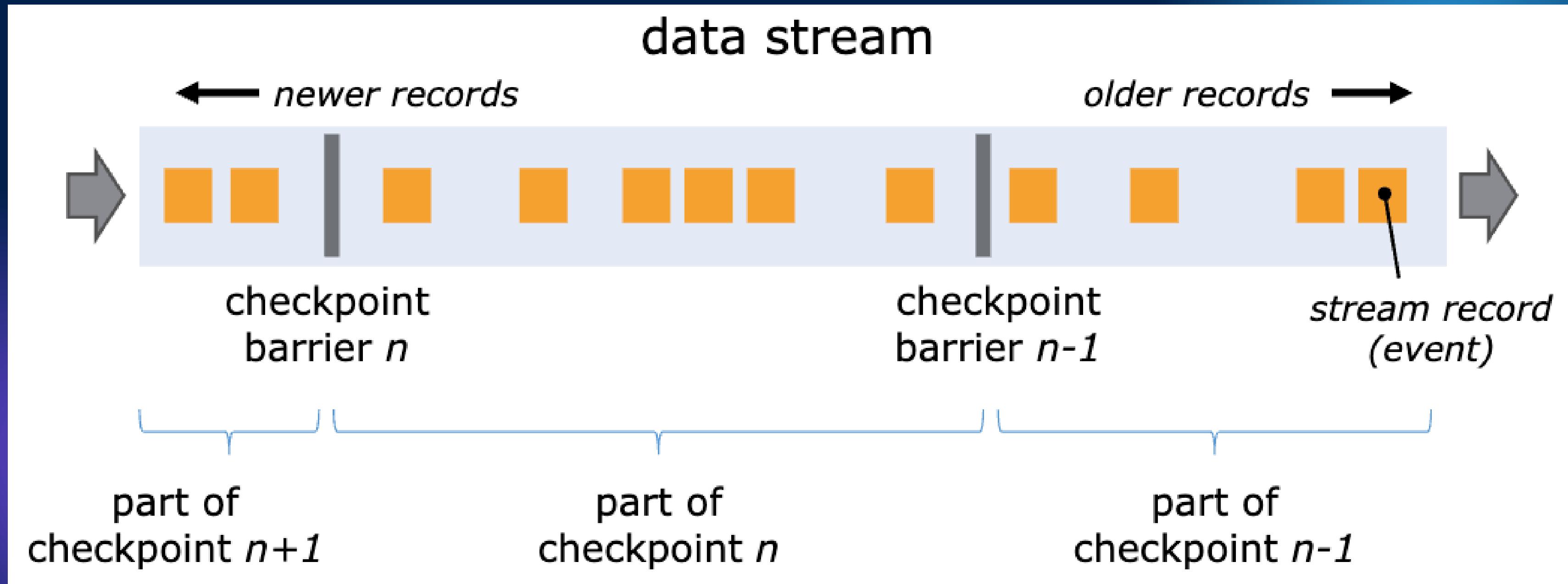


Concurrency, Responsibility Analysis, & Quality Attributes

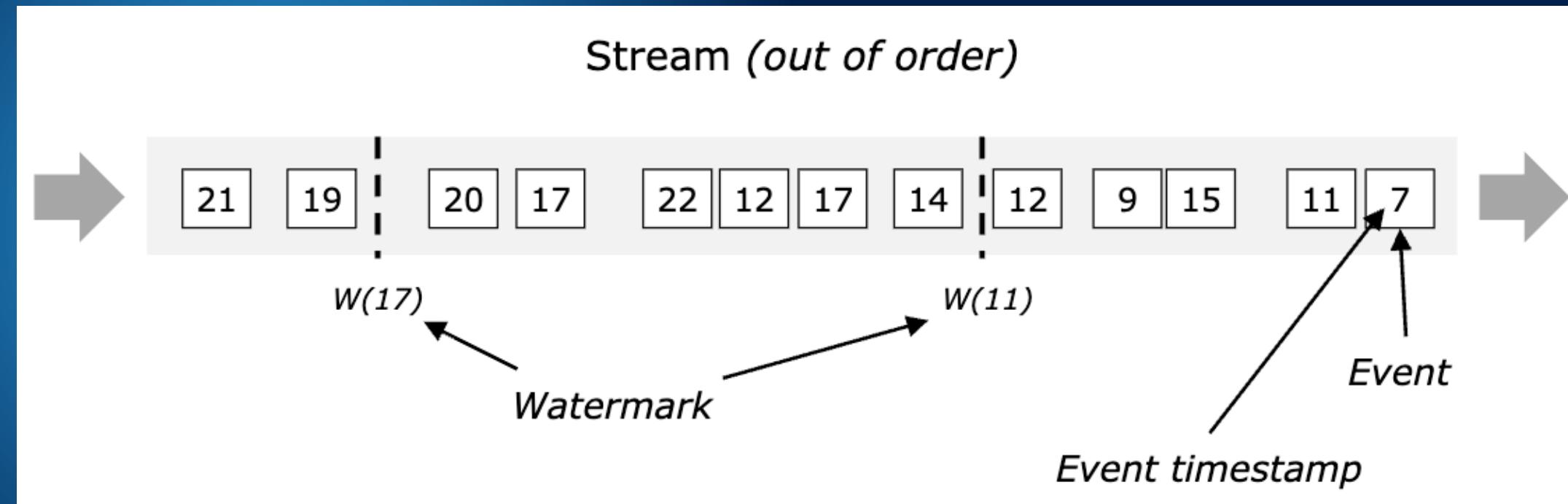
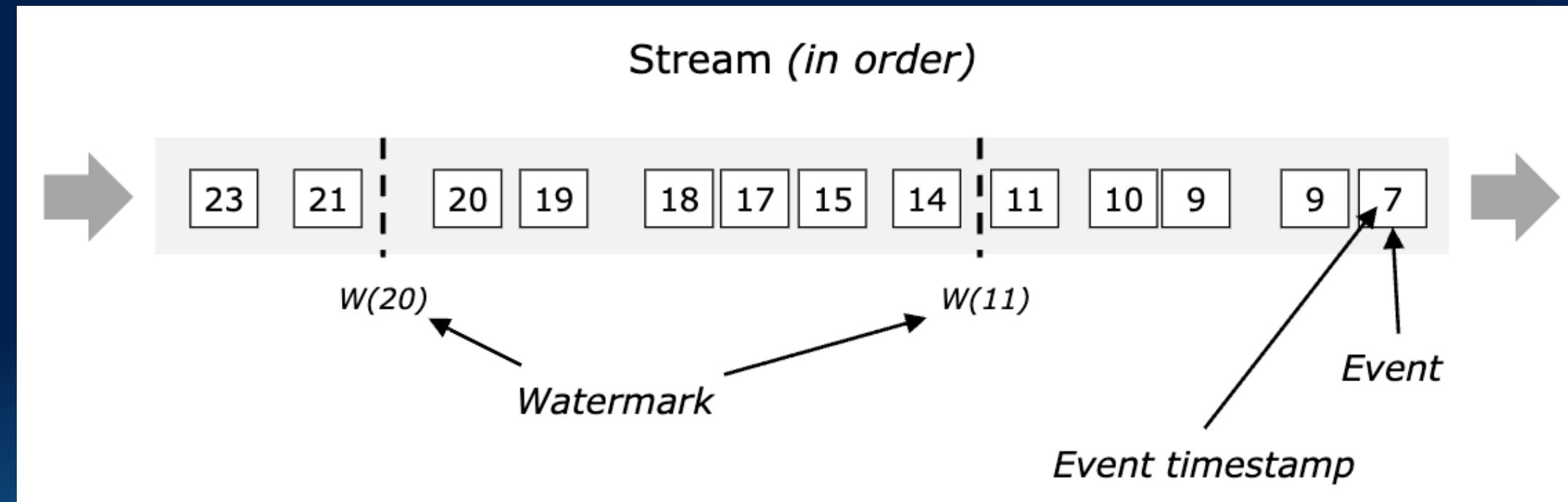
Concurrency Aspects - Operator Parallelism



Concurrency Aspects - Checkpointing

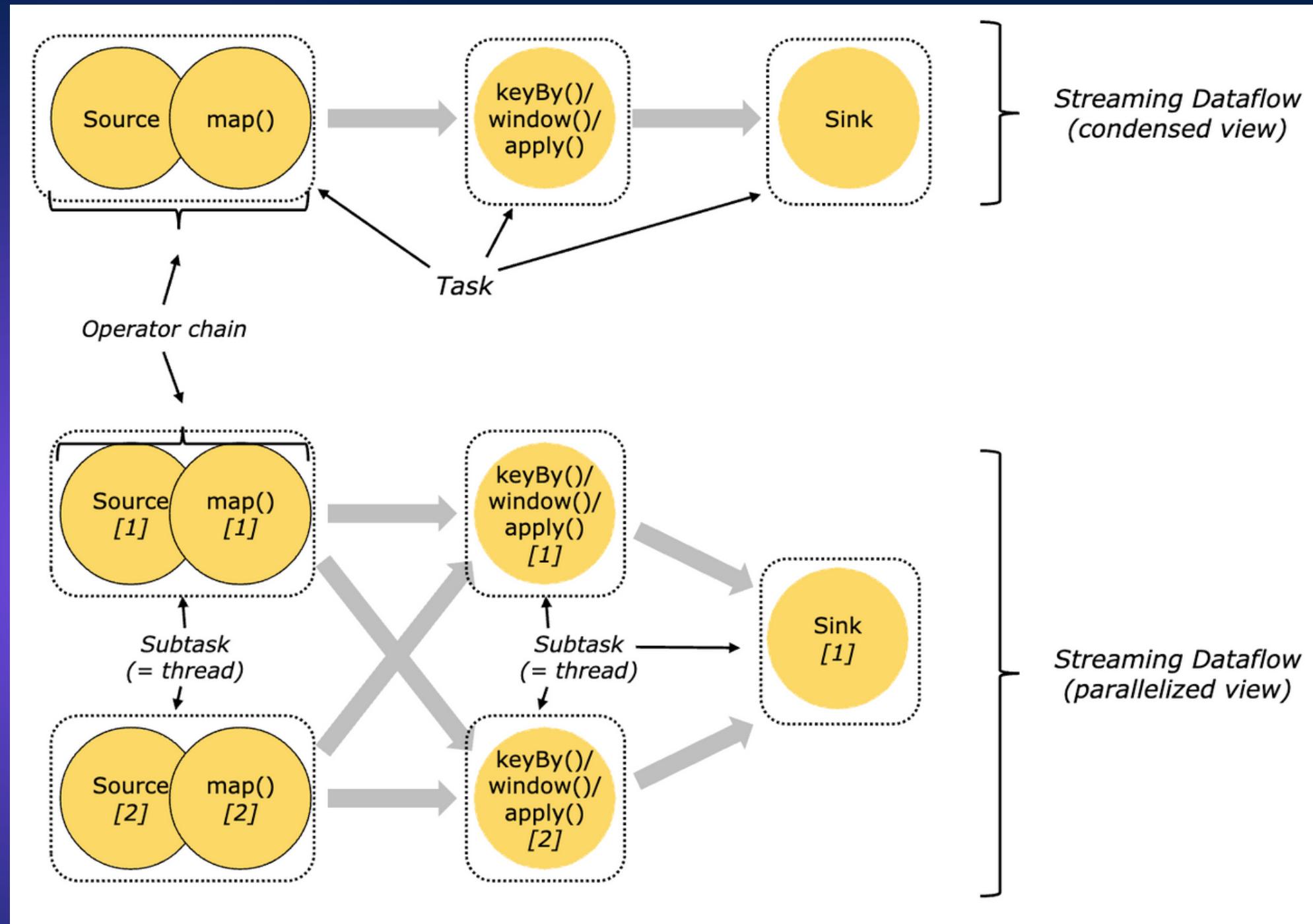


Watermarks



Job and Task Managers

Task and Operator Chaining



Division of Responsibilities

- Open-source project
- Committers
- PMC (Project Manager Committee)
- Release Manager



Security Measures

- Security features
 - Authentication
 - Authorization mechanisms
 - Data encryption
- Apache Software Foundation's licensing provisions
- Individual Contributor License Agreement (ICLA)



Quality Attributes

Performance



real-time data processing, complex event processing and Batch data processing

Scalability

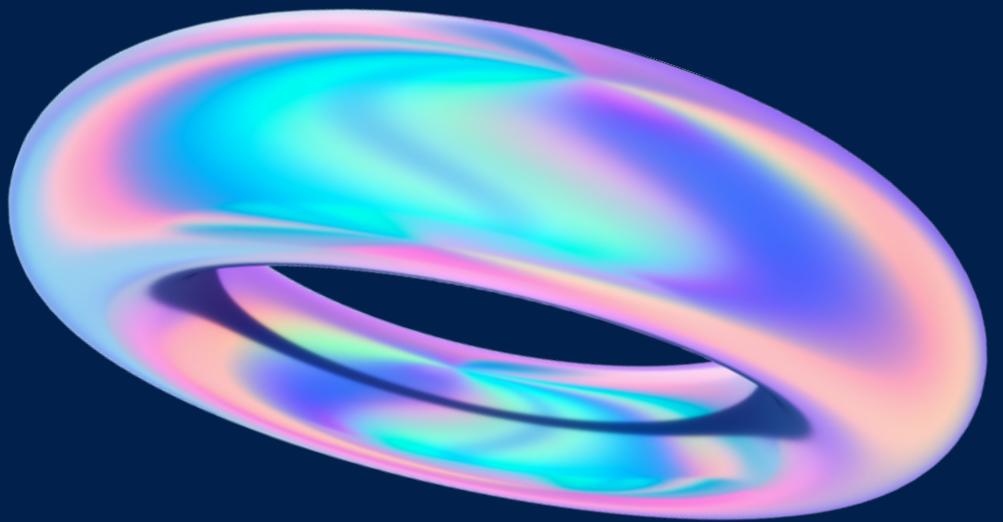


clustering and operational in any-scale system

Reliability

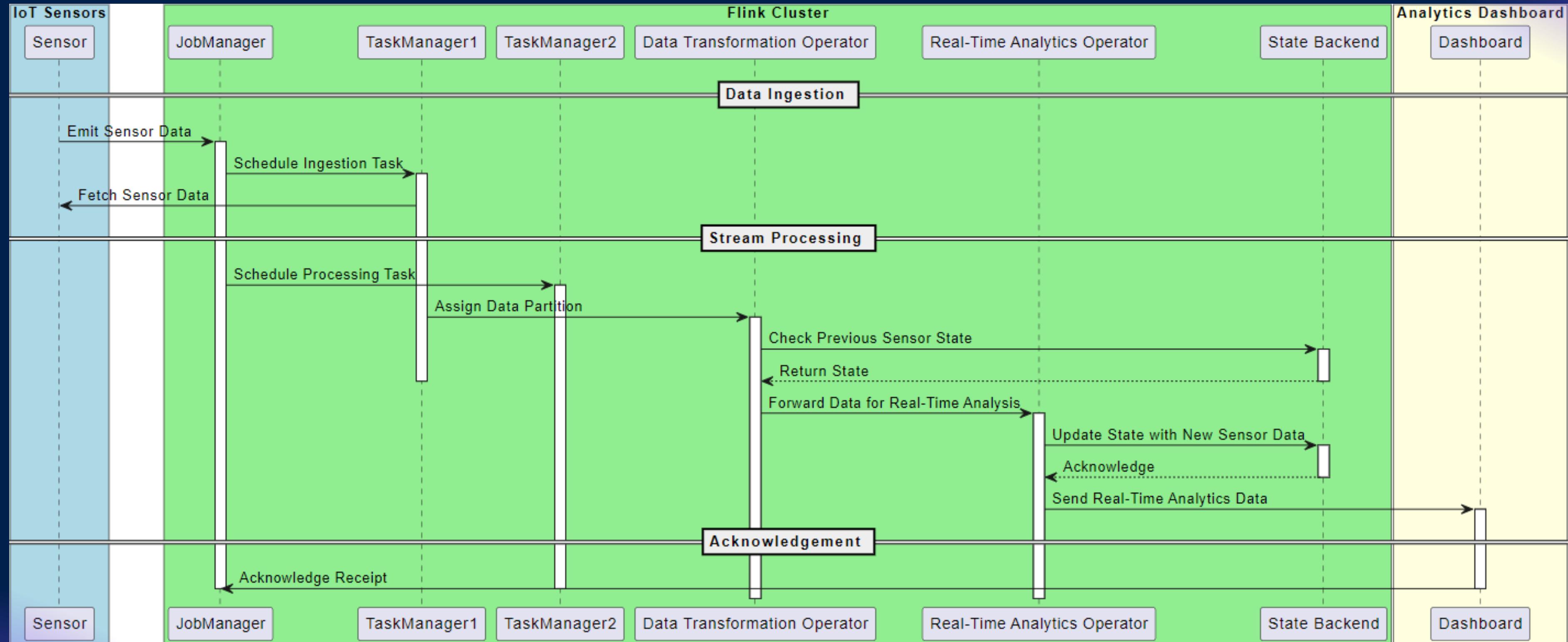


Fault tolerance and monitoring

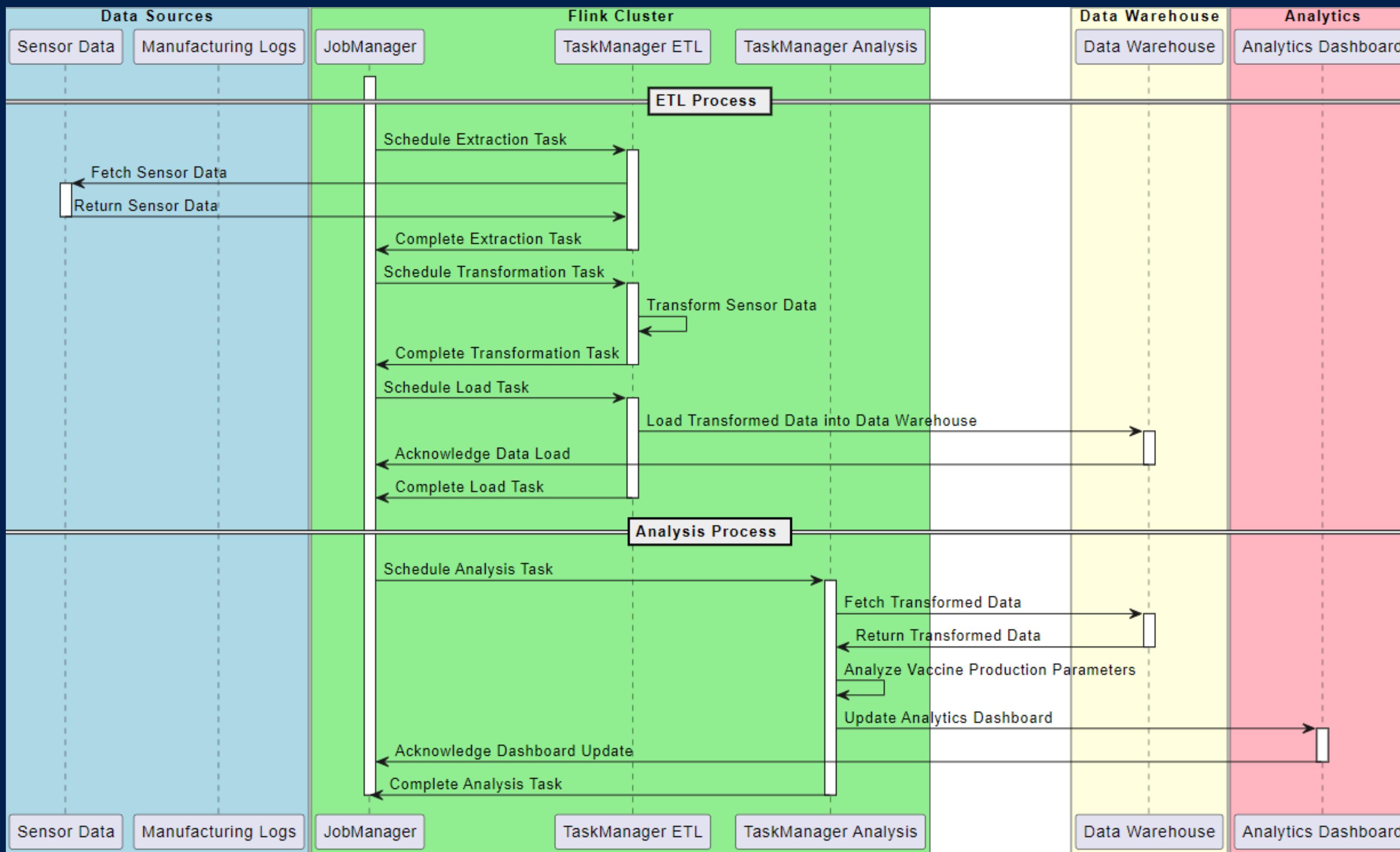


Use Cases

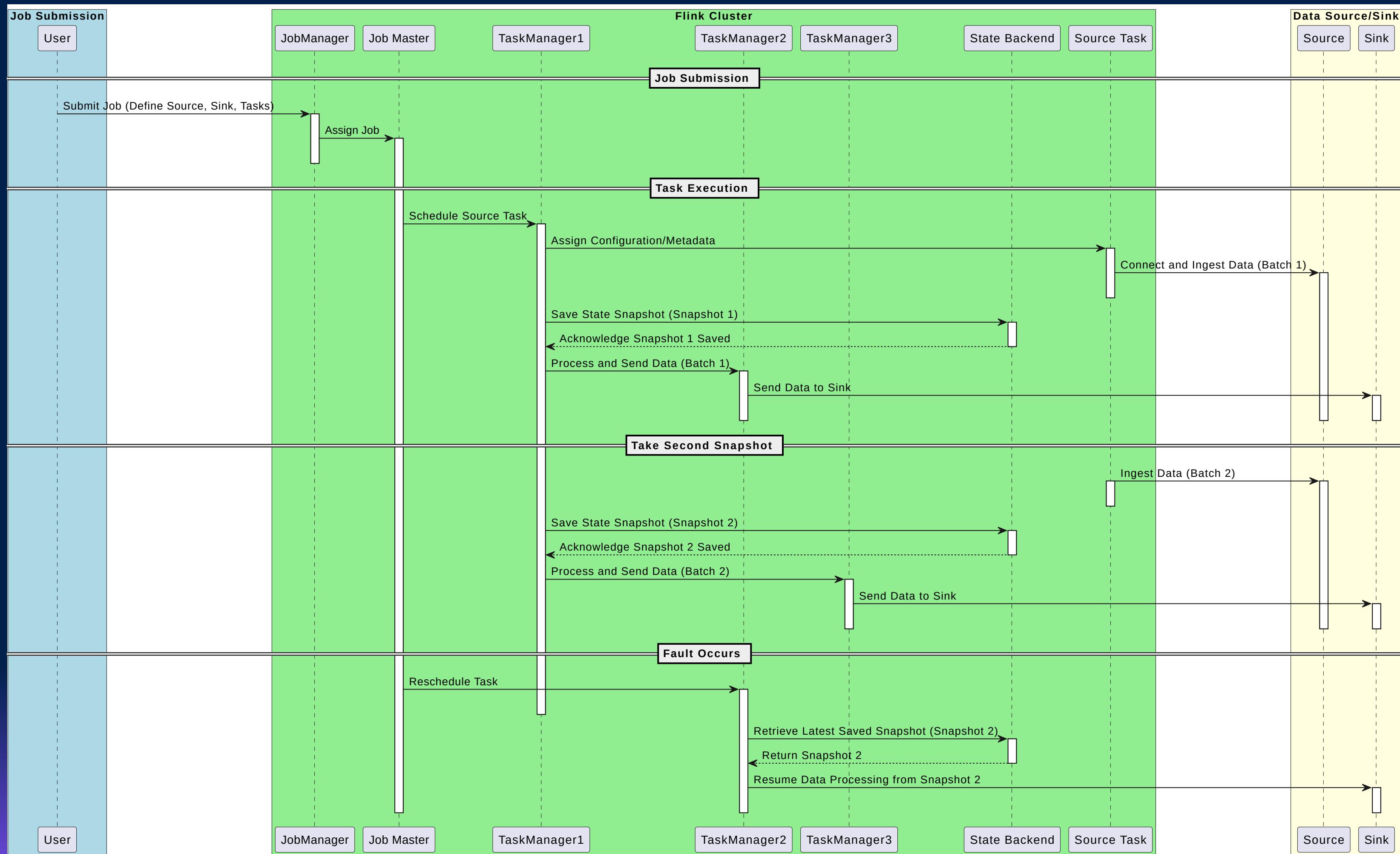
Stream Processing



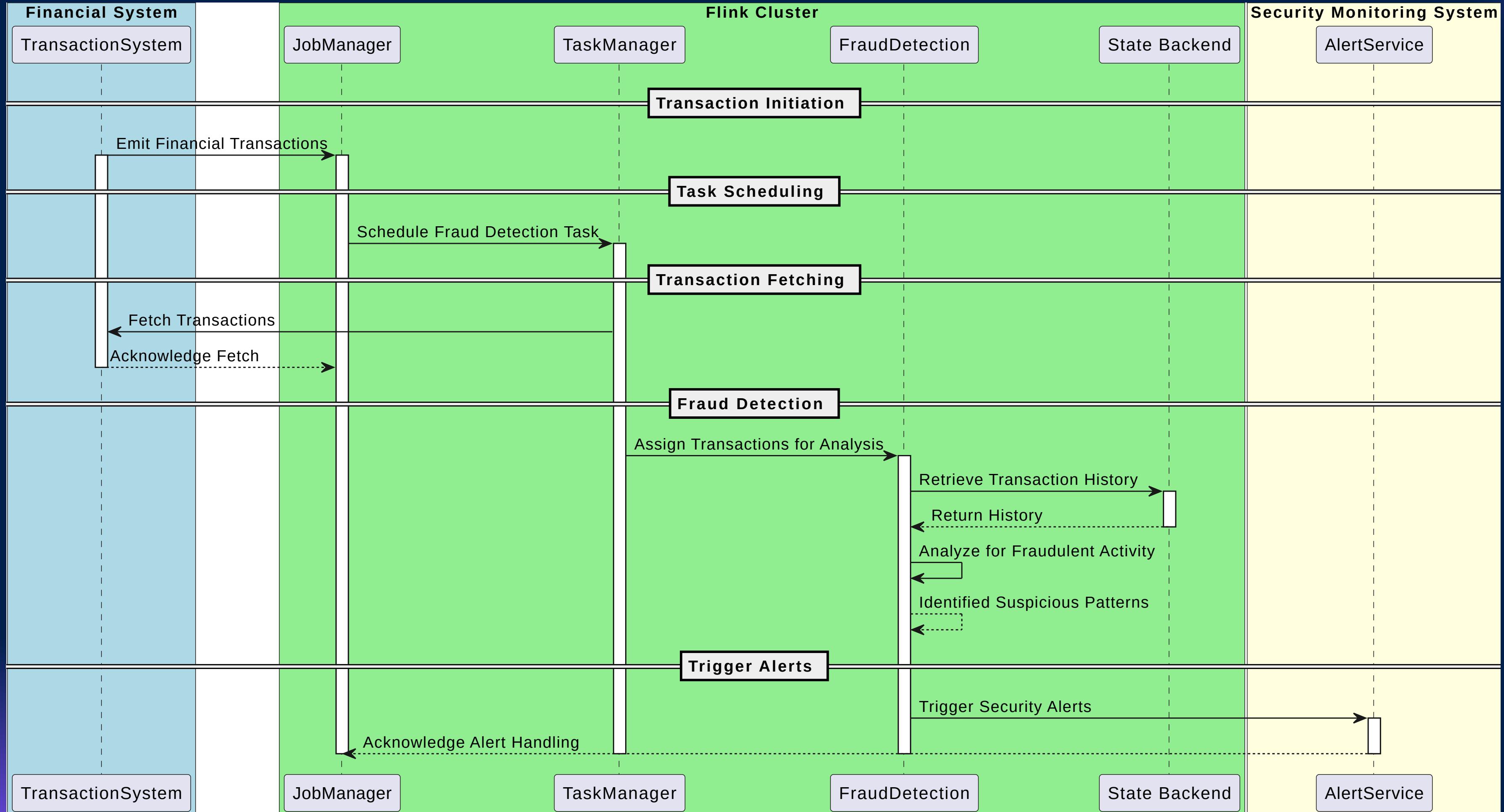
Batch Processing



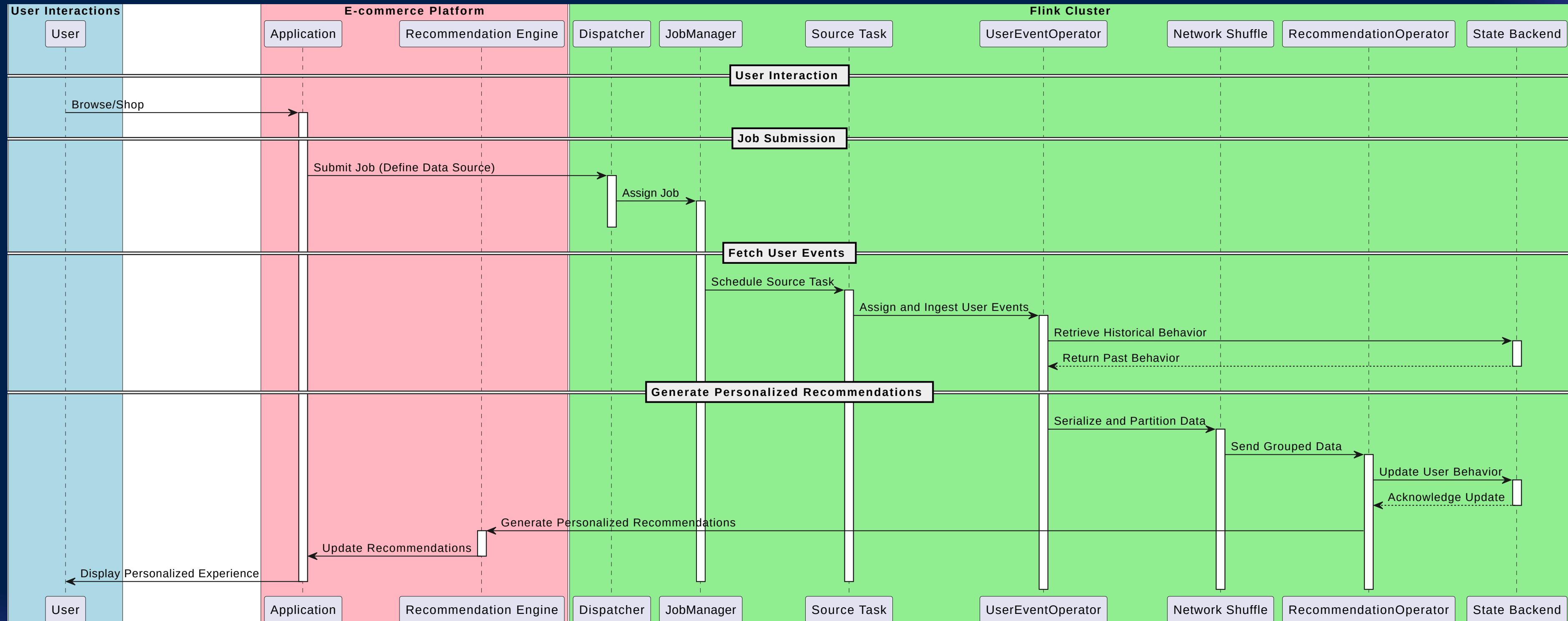
Fault Tolerance



Fraud Detection



Stateful Computations - User Recommendations



Thank You!

Any Questions?

References:

- [1] <https://flink.apache.org>
- [2] <https://github.com/apache/flink>
- [3] <https://www.apache.org>
- [4] <https://cwiki.apache.org/confluence/display/FLINK/Flink+Bylaws>
- [5] <https://docs.cloudera.com/csa/1.11.0/security/topics/csa-flink-security-overview.html>
- [6] <https://flink.apache.org/powerd-by/>
- [7] <https://thenewstack.io/3-reasons-why-you-need-apache-flink-for-stream-processing/>
- [8] <https://www.uber.com/blog/building-scalable-streaming-pipelines/>
- [9] <https://aws.amazon.com/managed-service-apache-flink/>
- [10] <https://flink.apache.org/how-to-contribute/overview/>
- [11] <https://nightlies.apache.org/flink/flink-docs-release-1.17/docs/dev/dataset/operators/>
- [12] <https://nightlies.apache.org/flink/flink-docs-release-1.17/docs/dev/dataset/execution/parallel/>
- [13] <https://nightlies.apache.org/flink/flink-docs-release-1.17/docs/concepts/stateful-stream-processing/>
- [14] <https://nightlies.apache.org/flink/flink-docs-release-1.17/docs/concepts/time/>
- [15] <https://nightlies.apache.org/flink/flink-docs-release-1.17/docs/concepts/flink-architecture/>
- [16] <https://medium.com/@BitrockIT/apache-flink-and-kafka-stream-a-comparative-analysis-f8cb5b946ec3#:~:text=Kafka%20Streams%20is%20a%20partially,data%20processing%2C%20and%20data%20analytics.>
- [17] <https://nightlies.apache.org/flink/flink-docs-release-1.17/>
- [18] <https://apache.googlesource.com/flink/+/release-1.0.2/flink-runtime-web/README.md>
- [19] <https://www.cloudduggu.com/flink/architecture/>

Contributing Members:

Rafael Dolores

Walid AlDari

Alex Arnold

Zachary Ross

Hashir Jamil

Nabaa Gaziy

Maaz Sidiqqi