

Stream Processing Systems

stream = database

- filter and transform events
- join event streams
- aggregate event streams
- replication of event streams

database -> state -> updated

stream -> state modification -> appended

Micro-batching - aggregate data in batches of configurable duration

Latency:

- computation triggered after batch times out
- each batch scheduled, loaded libraries, open connections

Programming model:

- no separation of mechanism and business logic
- changing batch size = different results

Event-based streaming - process events one by one

event-time system + aligned delay trigger = micro-batching

Flink

data-flow graph

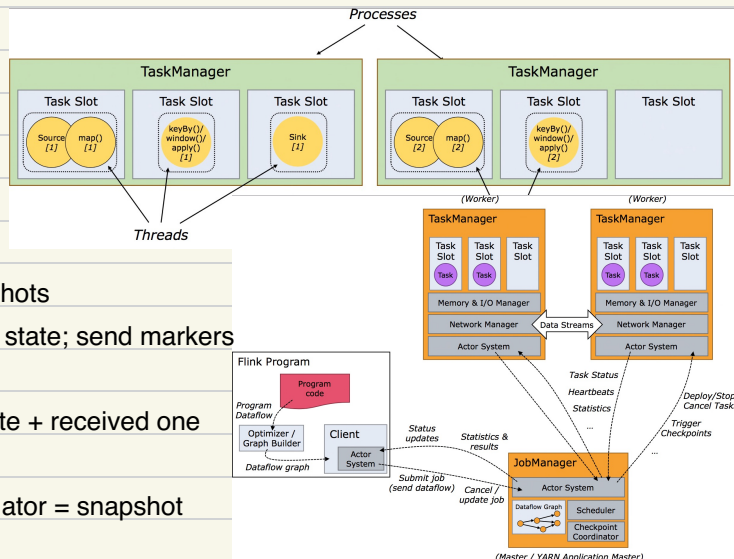
node in DFG = task

schedule with manager

Chandy-Lamport

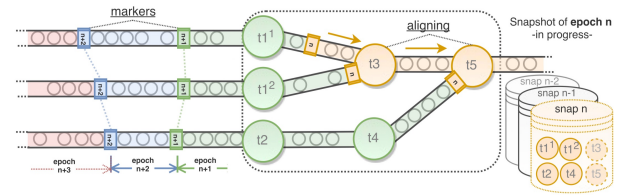
consistent global snapshots

- initiator - save local state; send markers
- receiving nodes:
 - save own local state + received one
 - forward marker
- marker reaches initiator = snapshot



Flink snapshots

- incremental snapshots
- markers + messages
- stream - repeatable



Event processing guarantees:

- at most once event will be processed once
- at least once event might flow twice
- just once event flow through operators once

