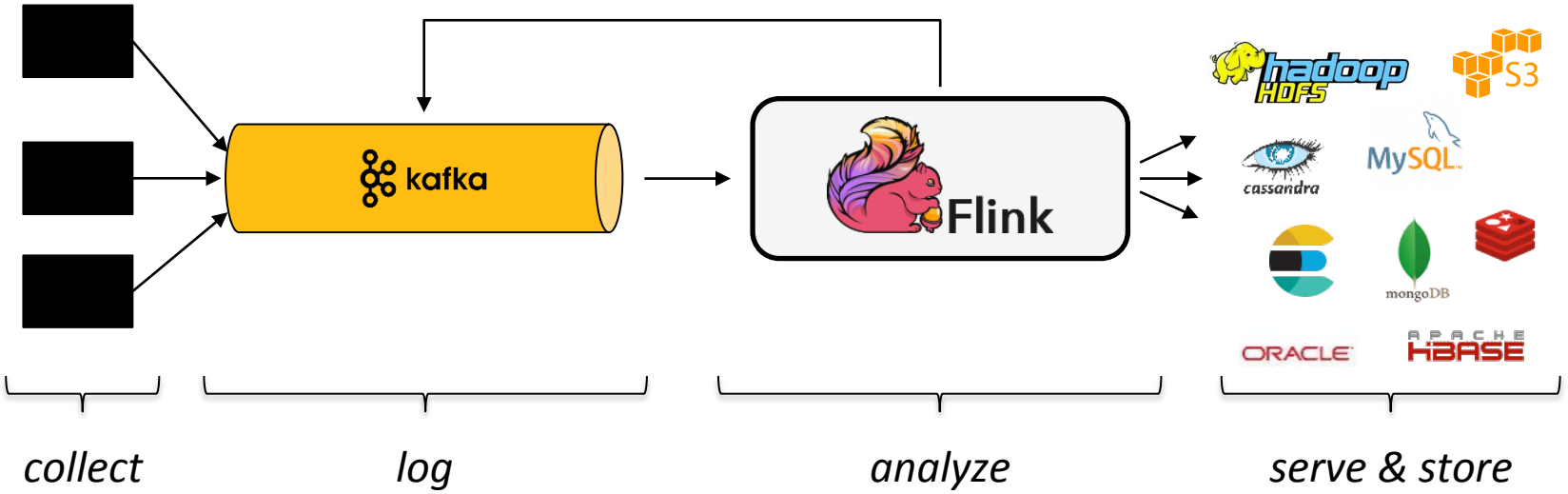
Apache Flink London Meetup

Stephan Ewen
@stephanewen

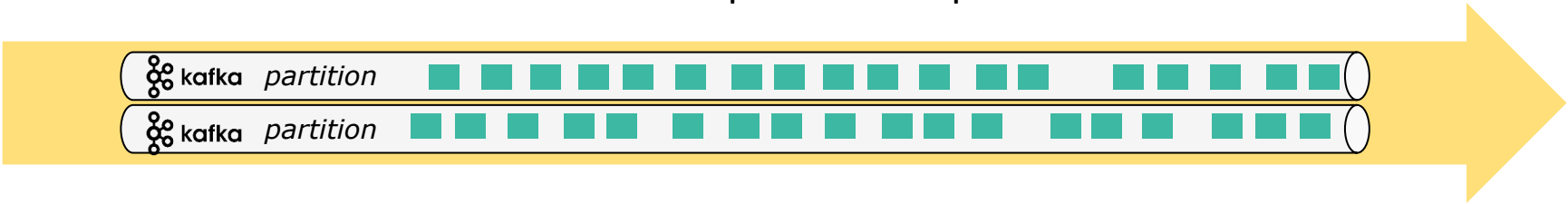
dataArtisans

A Stream Processing Pipeline

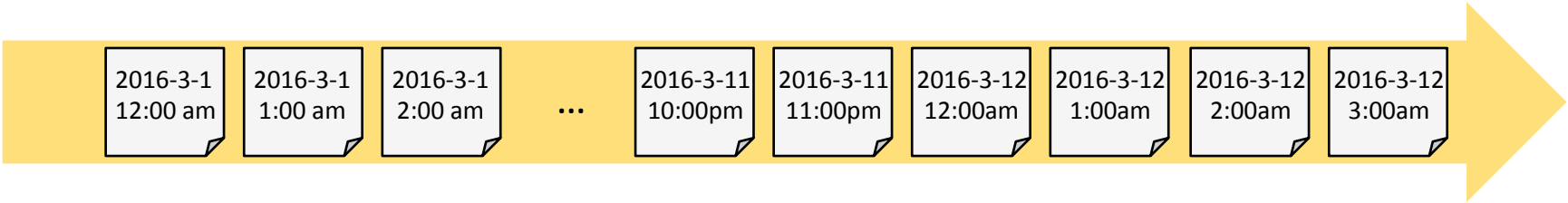


Continuous Data Sources

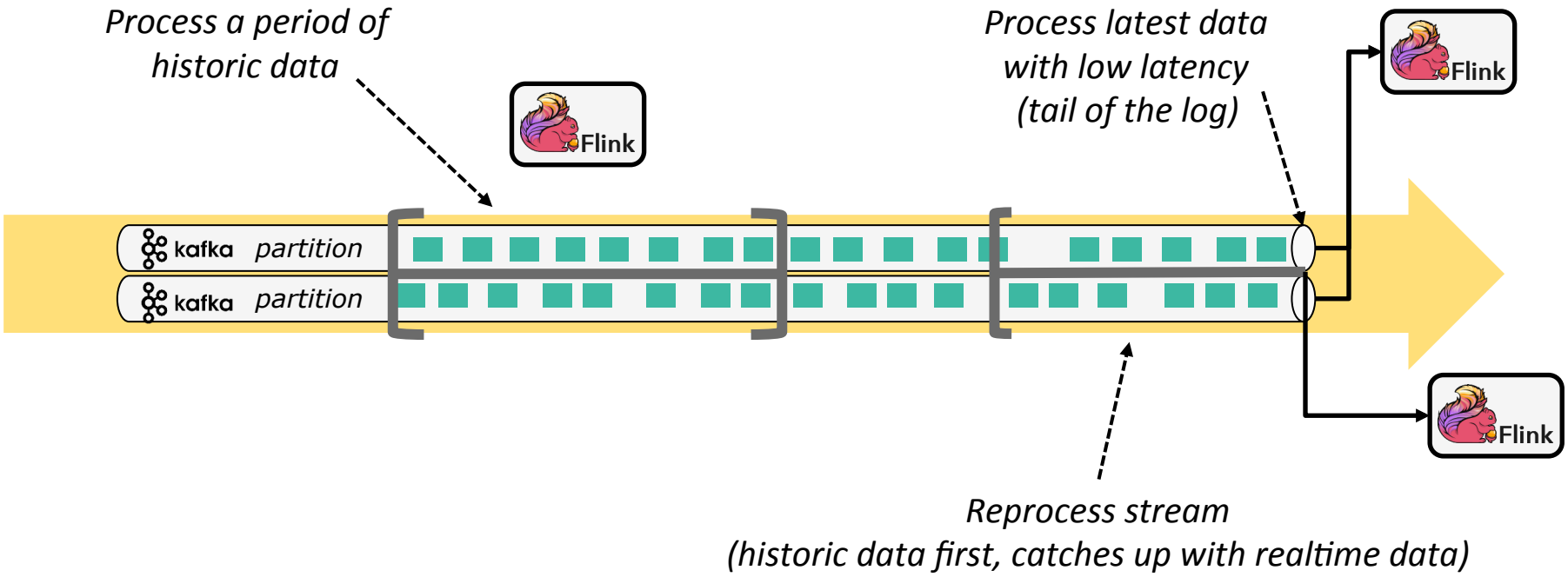
Stream of events in Apache Kafka partitions



Stream view over sequence of files



Continuous Data Sources



Dimensions of Continuous Apps

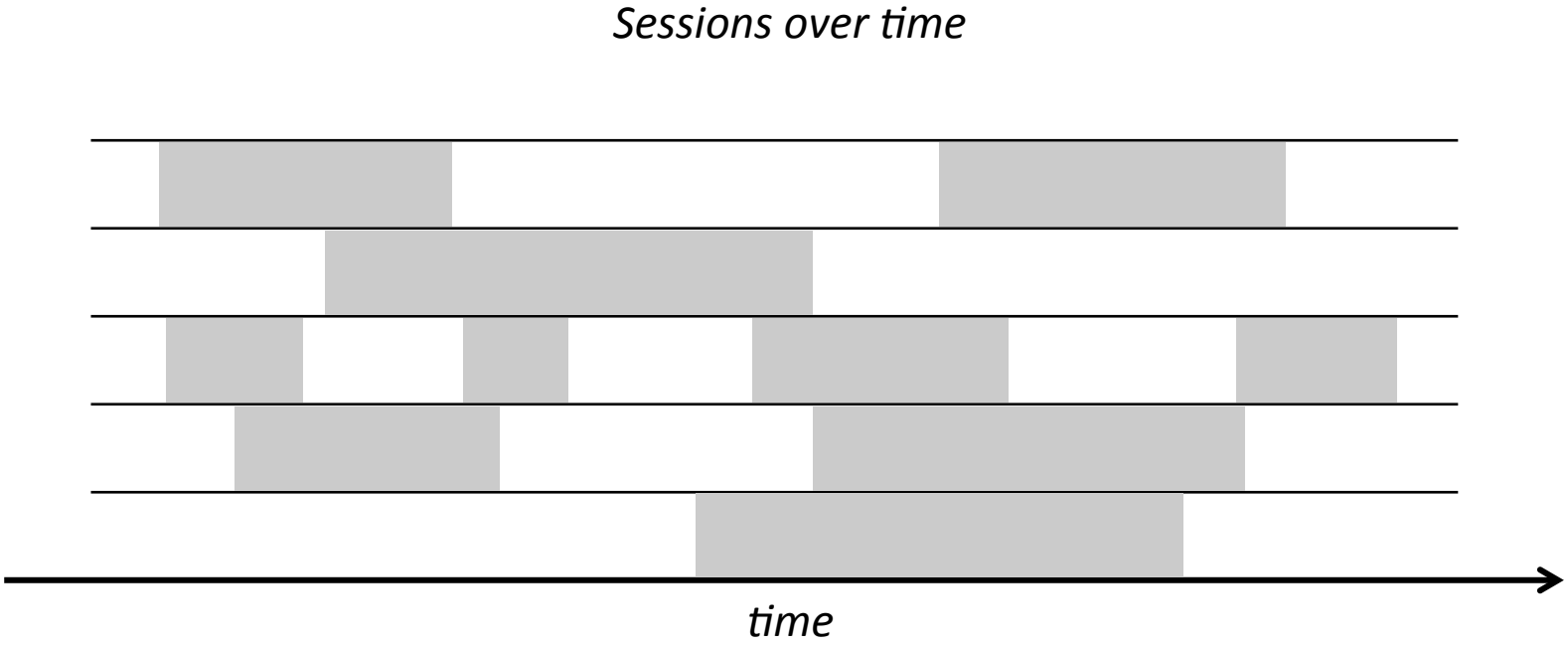
Time

- Grouping / Windowing
- Out-of-order Events
- Low Latency Results
- Late data

State

- State across Batch Boundaries
- Fault Tolerance
- Historic data / re-processing

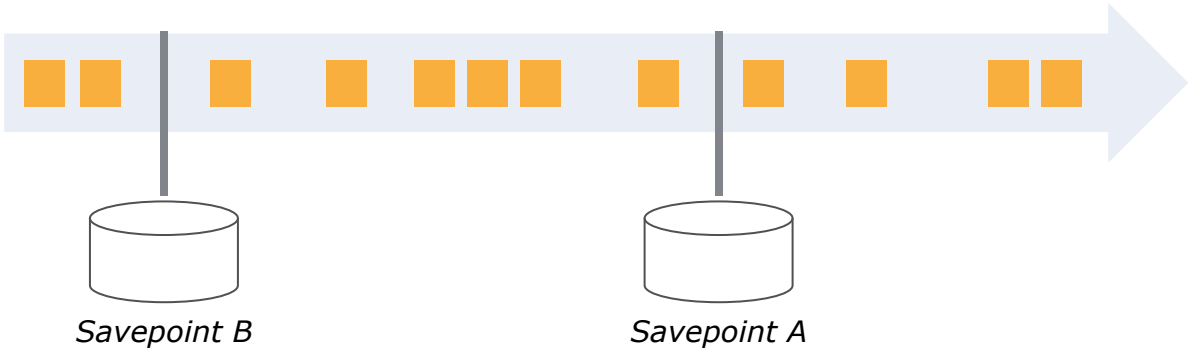
Continuous State



No stateless point in time

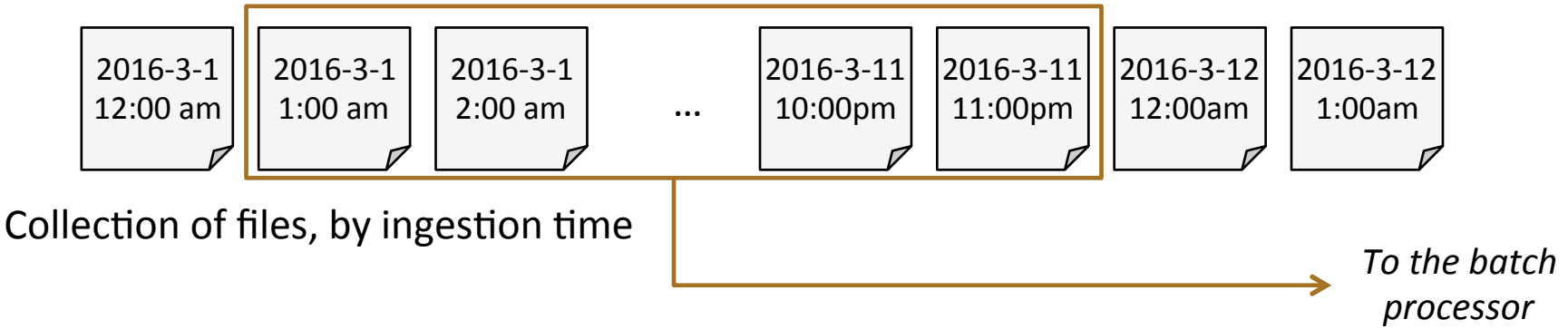
Savepoints

- A "Checkpoint" is a globally consistent point-in-time snapshot of the streaming program *(point in stream, state)*
- A "Savepoint" is a user-triggered retained checkpoint
- Streaming programs can start from a savepoint

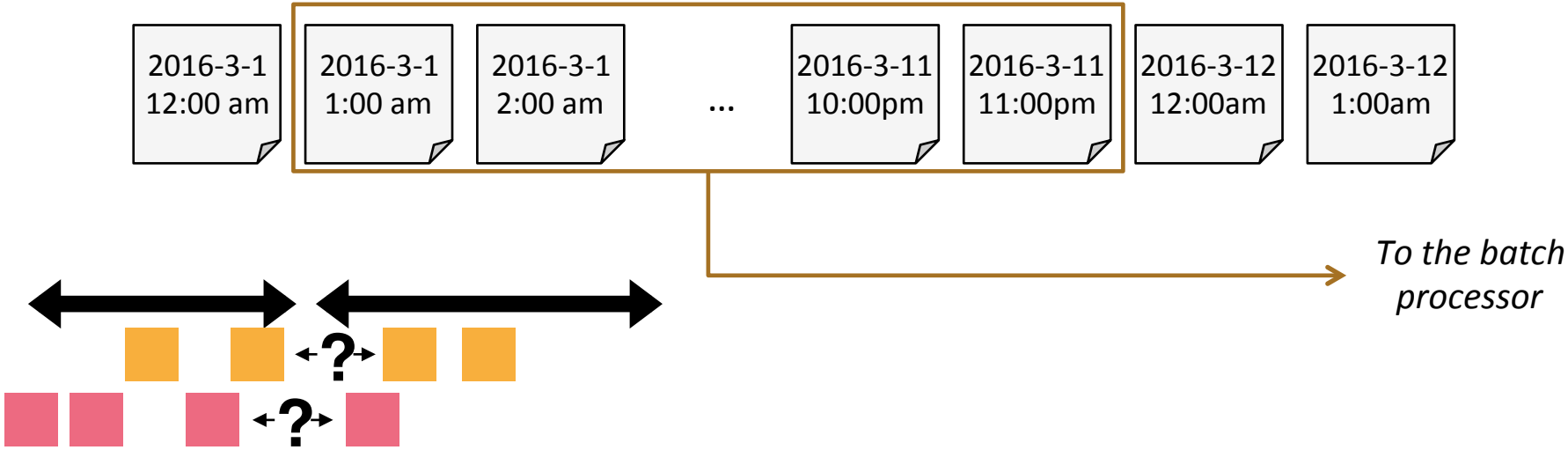


Re-processing data (in batch)

- Re-processing data (what-if exploration, to correct bugs, etc.)
- Usually by running a batch job with a set of old files
- Tools that map files to times



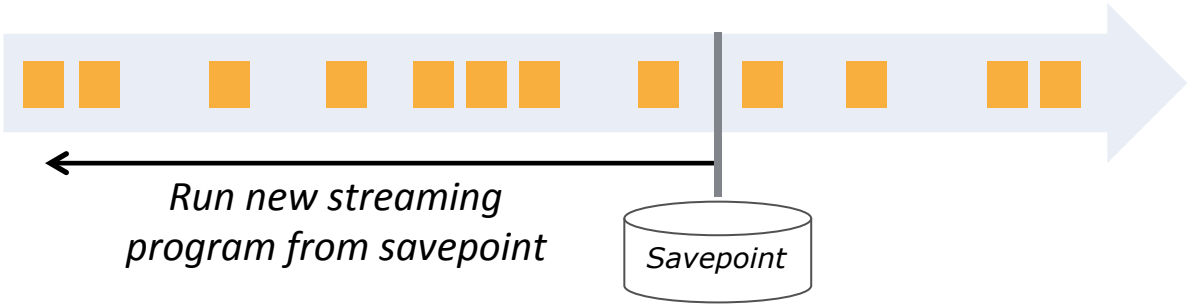
(Re)processing data (in batch)



Loses state across batches

Re-processing data (continuous)

- Draw savepoints at times that you will want to start new jobs from (daily, hourly, …)
- Reprocess by starting a new job from a savepoint
  - Defines start position in stream (for example Kafka offsets)
  - Initializes pending state (like partial sessions)



Demo Scenario

Pattern validation & violation detection:

- Events should follow a certain pattern,
  or an alert should be raised
- Think cybersecurity, process monitoring, etc