

# CS 744: FLINK

Shivaram Venkataraman

Fall 2022

# ADMINISTRIVIA

## In Progress

- Course Project Proposal feedback
- Midterm, Assignment 2 grading

## Resources for Course Projects

- Cloumlab reservations (Check Piazza)
- GCP credits (Email Roger and me)

# Applications

Machine Learning

SQL

Streaming

Graph

Computational Engines

Scalable Storage Systems

Resource Management



Datacenter Architecture



# DASHBOARDS

## Sales Dashboard

Total Sales

**\$3,256.8M**

Number of Deals

**17,164**

Avg Deal Size

**\$189,545**

Rev. per Salesperson

**\$20.5M**

Week of Date Closed

December 6, 200 - December 25, 20



Region

(All)

Country

(All)

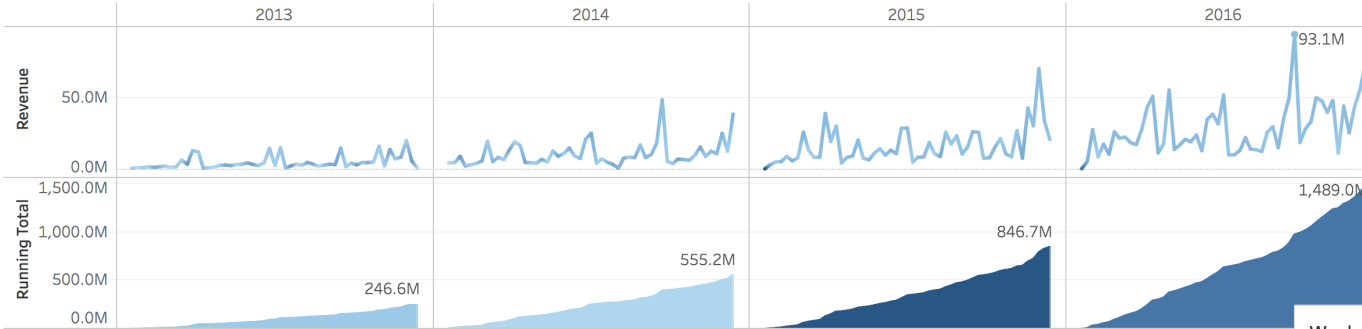
Sales Team

- (All)
- Small and Midmarket
- Enterprise

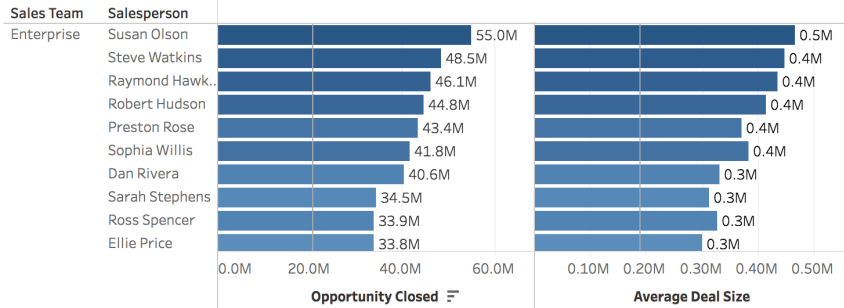
Avg Deal Size/Salesperson



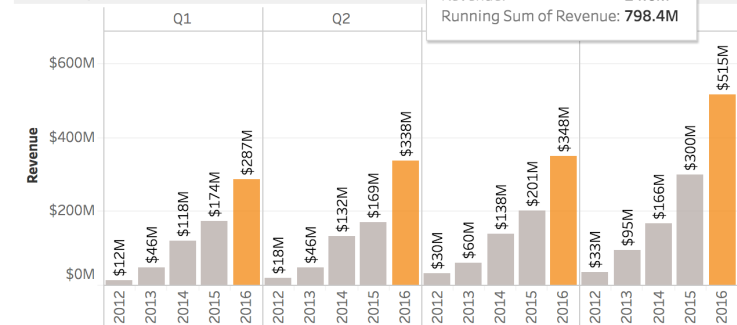
Revenue Over Time



Sales Team Performance

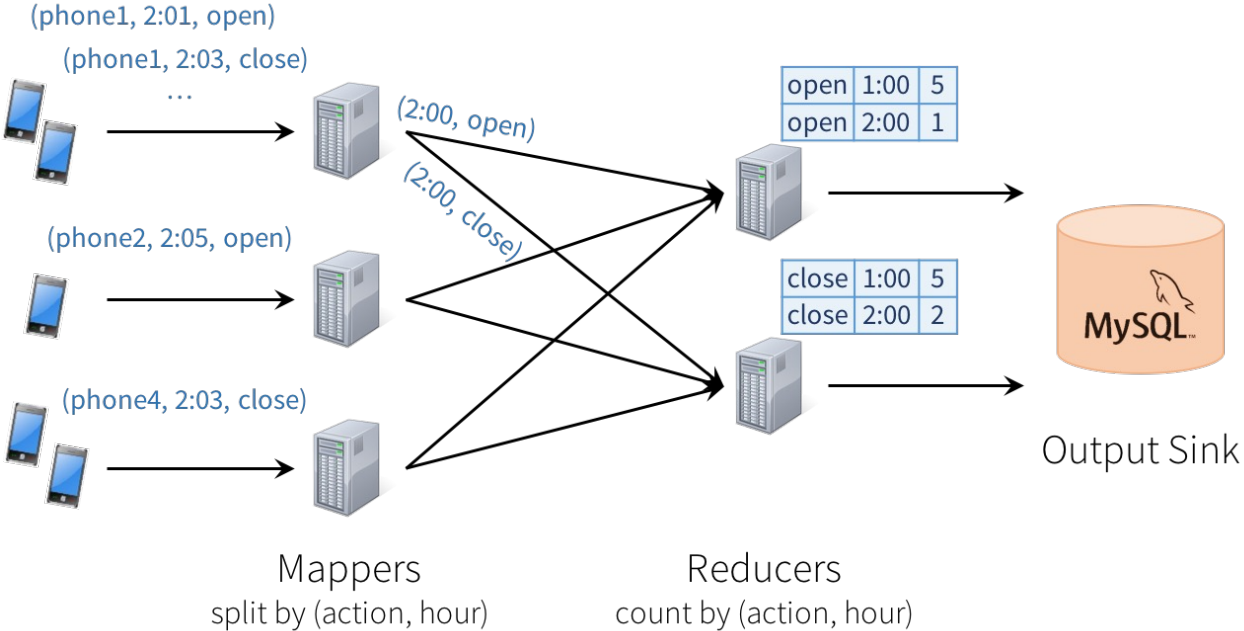


Revenue by Quarter

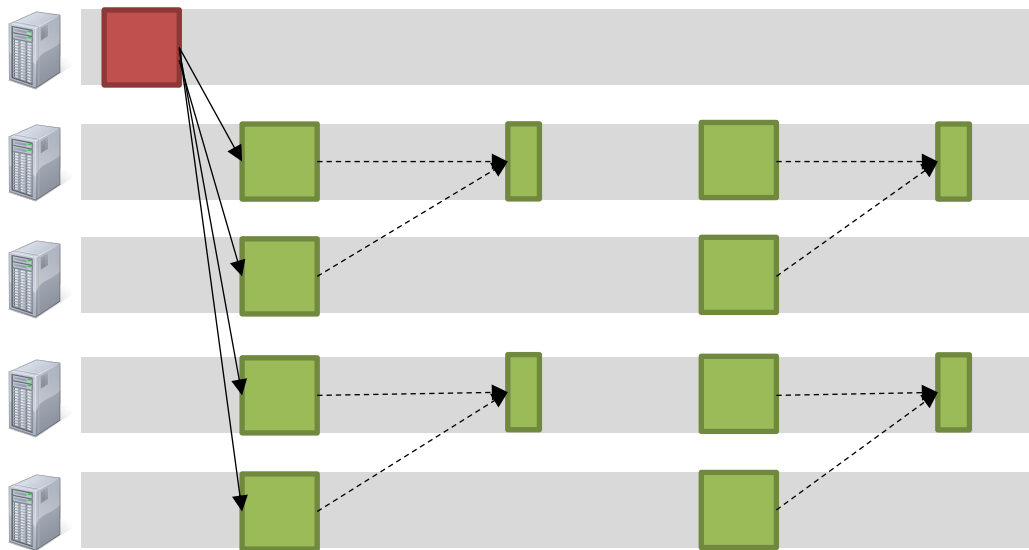


Week of September 4, 2016  
 Revenue: 14.6M  
 Running Sum of Revenue: 798.4M

# STREAMING COMPUTATION

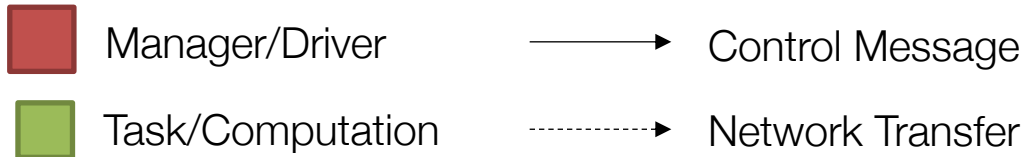


# FLINK: COMPUTATION MODEL



Long-lived operators

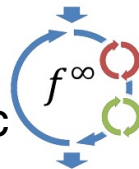
Mutable State



Google  
MillWheel

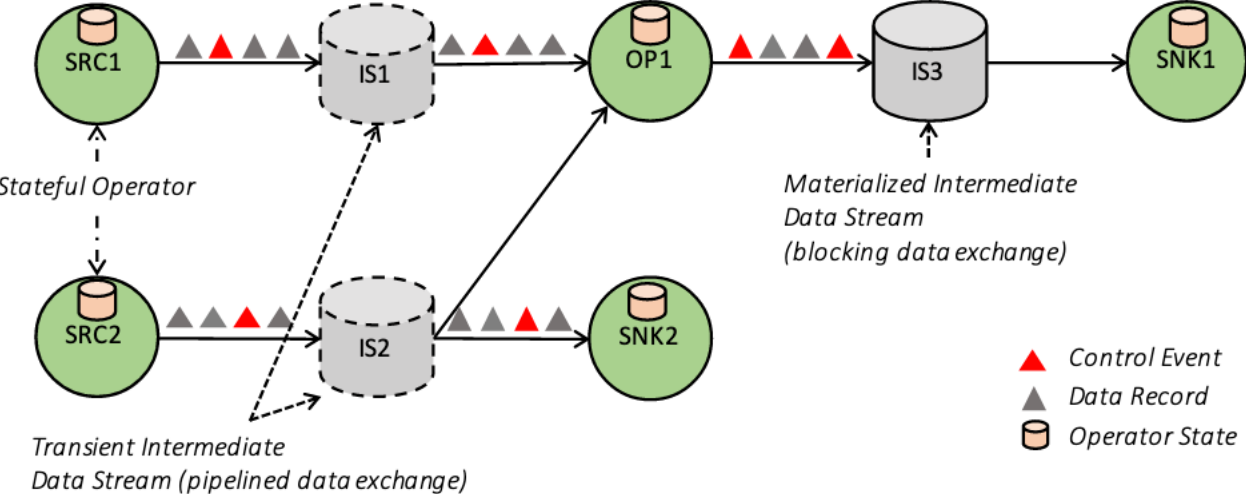


Streaming DBs:  
Borealis, Flux etc



Naiad

# INTERMEDIATE DATA STREAMS



# STATEFUL OPERATORS

Examples?

Challenge

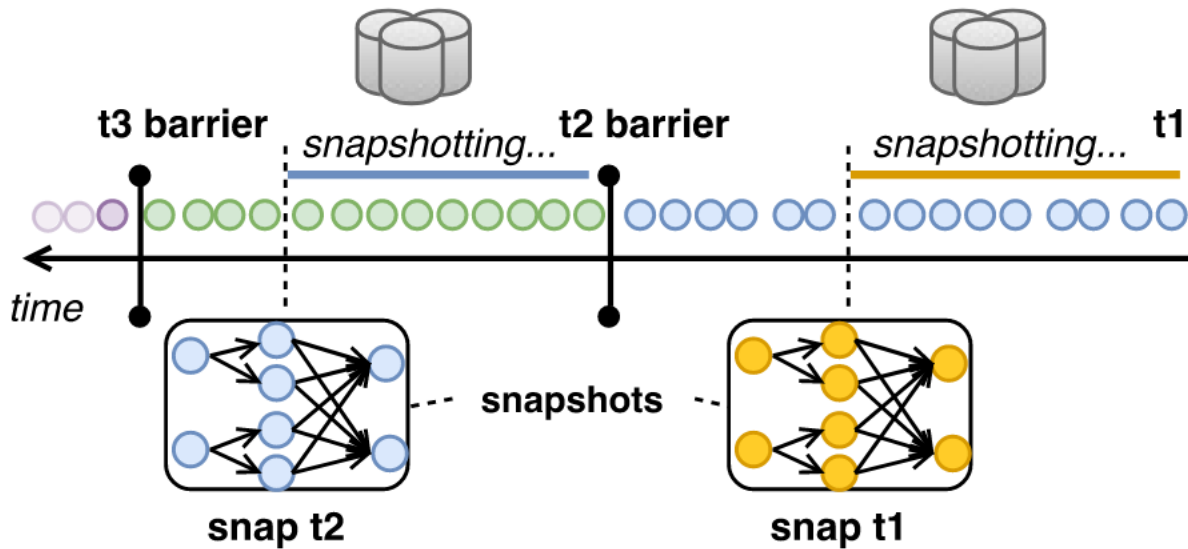
How to ensure fault tolerance?

Explicitly register local variables

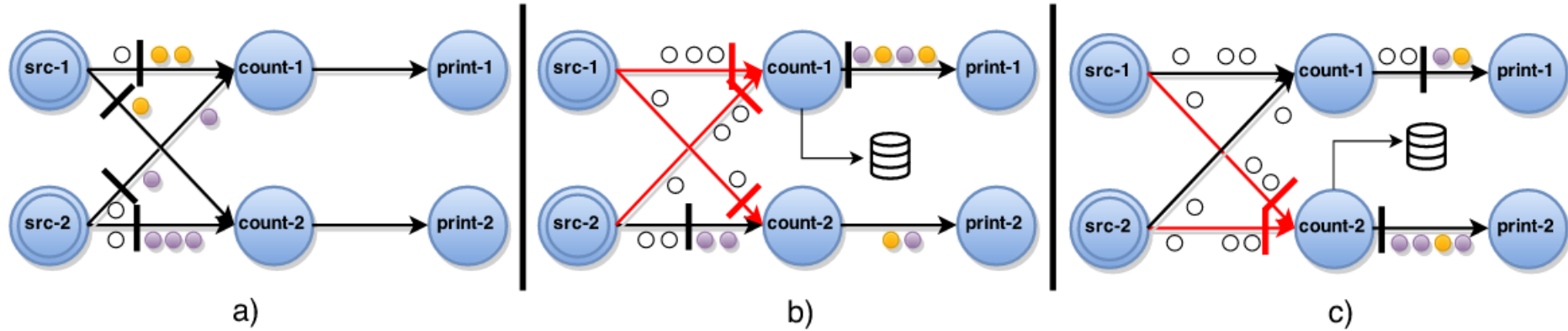
StateBackends that are automatically saved/recovered



# FAULT TOLERANCE: CHECKPOINTING



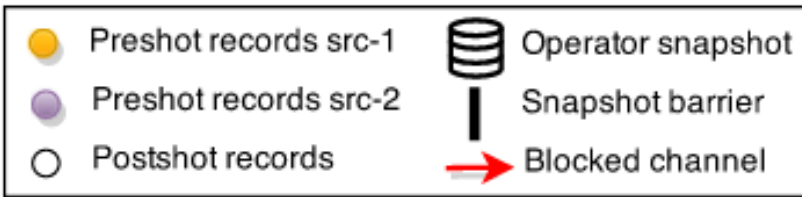
# ASYNCHRONOUS BARRIER SNAPSHOTTING



a)

b)

c)



# WATERMARKS, WINDOWS

Implements similar model as Dataflow

“Watermarks originate at the sources of a topology”

Propagate through the other operators of dataflow

Windows based on event-time, processing time, ingest time(?)

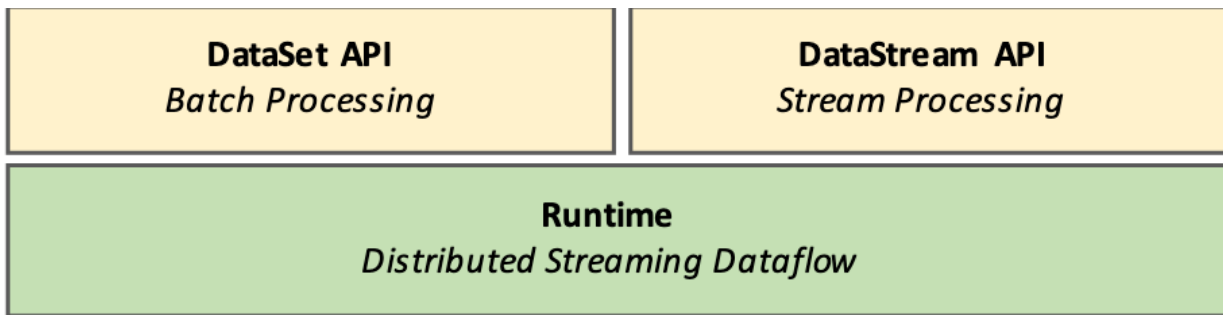
```
stream
  .window(SlidingTimeWindows.of(
    Time.of(6, SECONDS), Time.of(2, SECONDS))
  .trigger(EventTimeTrigger.create()))
```

# COMBINING BATCH, STREAMING

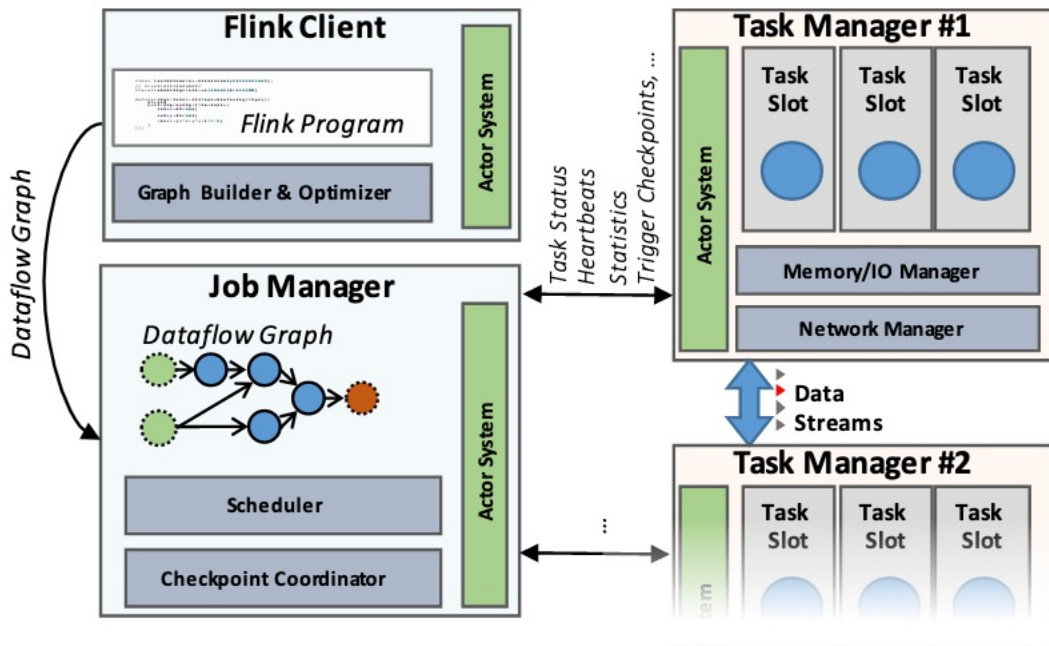
Blocked DataStreams

Turn off periodic snapshots

Blocking operators (e.g., sort)



# OVERALL ARCHITECTURE



# SUMMARY

Stream processing → Increasingly important workload trend

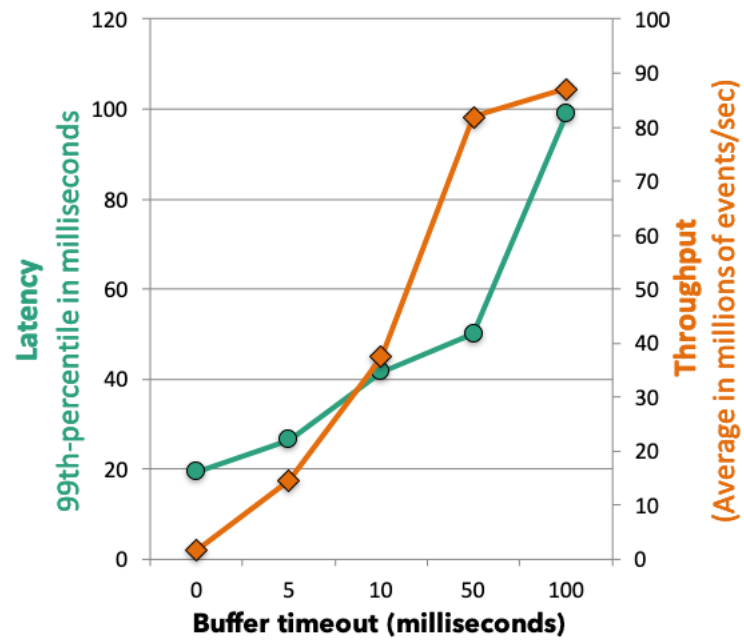
Flink: Distributed streaming dataflow to run streaming, batch, iterative

Distributed streaming dataflow

- Stateful operators
- Checkpointing based FT

# DISCUSSION

<https://forms.gle/idnKJWoBFHpRNC2j7>





Consider you are implementing a micro-batch streaming API on top of Apache Spark. What are some of the bottlenecks/challenges you might have in building such a system?

# SUMMARY

Next class: Spark Streaming