Data Stream Processing and Analytics

Spring Semester 2019

Course Info

Vasiliki (Vasia) Kalavri
kalavriv@inf.ethz.ch
What is this course about?

Low-latency analysis of continuous, distributed, rapid data events.

Stream processor:
- In-memory storage
- Standing queries
- External stable storage
- Output streams

Ad-hoc queries
Example streams and applications

Sensor measurements
  • anomaly detection, incident risk calculation

Financial transactions
  • fraud detection, stock trading

Location and traffic data
  • report train system status, find optimal routes

Web logs
  • online recommendations, personalization

Network packets
  • intrusion detection, load balancing

Online social interactions
  • trending topics, sentiment analysis
Topics

- Systems
  - Architecture and design
  - Scalability and elasticity
  - Fault-tolerance and processing guarantees
  - State management

- Algorithms
  - Windowing semantics and optimizations
  - Basic data stream mining
  - Complex event processing

- Streaming applications and use-cases
Tools

Apache Flink: flink.apache.org

Apache Kafka: kafka.apache.org

Apache Beam: beam.apache.org

Google Cloud Platform: cloud.google.com

Timely Dataflow: www.frankmcsherry.org/timely-dataflow
Course Structure

1. Lectures
   • Mondays 10-12
   • ~30’ discussion on topics of previous week
   • Introduction of new topics

2. Exercise Sessions
   • Mondays 13-15
   • Seminar-style: review and discuss research papers
   • Hands-on: analyze streaming data, use and compare streaming tools

3. Semester Project
   • In teams of 2 students
   • Implement, test, and evaluate a new feature in a stream processor
Semester Project

1. Choose your teammate

2. Choose your system
   1. Apache Flink: Java, high-level API, component-heavy, fast
   2. Timely Dataflow: Rust, low-level API, lightweight, super-fast

3. Choose your topic

4. Send me an e-mail by 25-Feb-2019
Semester Project

- Development on gitlab (https://gitlab.inf.ethz.ch)
  - Create a project and give me access

- Deliverables
  - code with comments
  - documentation
  - tests
  - written reports

- Milestones
  - Midterm progress report [15-Apr-2019]
  - Final report [1-June-2019]
Grading Scheme

• No Exam
• Participation in class: 10%
• Weekly assignments (reviews and hands-on): 50%
• Semester project (code and reports): 40%