**Final Project Topic: Google Cloud Dataflow Real-time Service for Batch and Stream Processing**

**Hanjiao Qiu**

**Abstract**

Google Cloud Dataflow is a fully-managed cloud service and programming model for batch and streaming big data processing. It can be used to process massive-scale, unbounded and unordered data. It also supports the unaligned, event-time-ordered windows modern data consumers require. This project will demonstrate how to build your streaming and batch processing pipeline using Cloud SDK and show how does Cloud Dataflow run and optimize your pipeline. Moreover, it will integrate with Cloud Storage, Cloud Pub/Sub and BigQuery for seamless data processing.

**Problem Statement**

Process massive HTTP Archive dataset using Google Cloud Dataflow. Analyze the trends in web technology and measure the website performance, such as: 1) Search and sort the popular CDN (Content Delivery Network) providers by building Dataflow pipeline. 2) Compute the distribution of requests for different image format. 3) Calculate the average downloaded content length based on content type. 4) Use Dataflow pipeline derived data as a side input to filter image object whose size is larger than global mean value. 5) Use Dataflow time-based windowing model to aggregate total number of requests per page.

**Dataset**

HTTP Archive data (<http://httparchive.org/)>, also available through BigQuery. (40+ GB, csv)

**Technology Used**

Google Cloud Dataflow SDK, Eclipse IDE version 4.5.1, Java version 1.8.0.

**Overview of steps**

1. Environment setup: Configure Eclipse, install Dataflow SDK, run example.
2. Prepare the dataset, load in data from Cloud Storage or BigQuery.
3. Use the Dataflow Java SDK to build a linear pipeline to process HTTP Archive data.
4. Apply more sophisticated models to do real-time stream processing.
   1. Branching a pipeline: use side outputs, process input once, multiple outputs.
   2. Window model for processing bounded / unbounded streaming data.
5. Validate the results using BigQuery. Analyze and visualize the results in D3.

**Issues & Lessons learned**

With the practice of windowing model, we can implement more complicated models. Cloud Dataflow data transforms also supports triggering, incremental processing model and so on. Highly recommend reading the official research paper and try using them to solve real world problems.

**Video URL**

Short video: <https://youtu.be/Jt5J4SMQNko>

Full video: <https://youtu.be/kjFoB6ZM9Iw>