**Beam Documentation**

Apache beam is for streaming (unbounded data, not fixed size) and batch (bounded data, fixed size) pipeline. Apache beam is a streaming system that is able to handle batch processing as a special case of streaming.

Apache beam run on runner 🡪 direct runner, spark runner, GCP dataflow runner, flink runner, etc

**Main features of Apache Beam:**

* Main features
  + Windowing 🡪 determines where in event time data are grouped together for processing. For streaming, there must be a window so the data become bounded
    - Fixed window 🡪 fixed time window
    - Sliding window 🡪 static size with slide period, may overlap between windows
    - Session window 🡪 timeout gap
  + Triggering 🡪 determines when in processing time the results of groupings are emitted as panes. allow developers to specify when to emit the output result for a given window 🡪 minimizing latency, we desire to make event time == processing time
* When event time != processing time 🡪 latency
* Event time 🡪 when the event itself actually happened
* Processing time 🡪 when the event arrives to our system for processing

**Core transformation in Apache Beam**

* ParDo (Map)
  + General parallel processing
  + Expects user defined logic (Do Fn in function definition, ParDo in pipeline)
  + Element wise operation
  + Can be applied to unbounded data
* GroupByKey / CombinePerKey (Reduce)
* Aggregate data
* Group elements by key
* Collect all data for a given key in a window
* Better to use CombinePerKey for faster operation

**Installing Apache Beam**

1. Optional: create a python virtual environment, so all libraries will be installed here.
2. Install Airflow:

pip install apache-beam

to include GCP Dataflow runner:

pip install apache-beam[gcp]

**Executing a pipeline**

1. Direct:

python -m apache\_beam.examples.wordcount \

--input /path/to/inputfile \

--output /path/to/write/counts

1. Spark:

python -m apache\_beam.examples.wordcount \

--input /path/to/inputfile \

--output /path/to/write/counts \

--runner SparkRunner

1. GCP:

python -m apache\_beam.examples.wordcount \

--input gs://dataflow-samples/shakespeare/kinglear.txt \

--output gs://<your-gcs-bucket>/counts \

--runner DataflowRunner \

--project your-gcp-project \

--region your-gcp-region \

--temp\_location gs://<your-gcs-bucket>/tmp/

**Make a data pipeline**

1. Create the pipeline in python

worcount.py 🡪 counting each word at a text file

my\_pipline.py 🡪 for counting number of click for each user, no text file input

1. Create the input file, input.txt

Place the input in home directory ~/input.txt

1. Run the beam job:

For wordcount:

python -m apache\_beam.examples.wordcount --input ~/input.txt --output ~/counts

There will be an counts-00000-of-00001 file as an output at home directory

For my\_pipeline:

python my\_pipeline.py

There will be an output-00000-of-00001 file as an output at home directory