cab.csv

(1) csvimport

Generate an Accelerator dataset from input file.

One row per taxi trip.

$slice_{S-1}$, $slice_2$, $slice_1$, $slice_0$

A dataset of C columns, partitioned into S parallel slices.

pickup_datetime, latitude, longitude, ... vendor

(2) quant_timestamps

Use existing columns to compute a new cleaned date column with coarser time resolution.

pickup_datetime ... startdate

(3) filter_columns

Create a new dataset containing only the columns needed for further processing.

This is done in a fraction of a second, since the operation just involves creating a set of links, or references, to existing data.

startdate, latitude, longitude

(4) hashpart

hash(startdate) % S

Partition data according to column "startdate". Each row is written to slice hash(startdate) % S

As a consequence, all entries for a certain startdate will be located in one slice only.

Then slices can be processed independently in parallel.

startdate, latitude, longitude

(5) heatmap

For each startdate, create a heatmap using a Python Counter-object mapping position -> occurances.

Write new dataset with columns "timestamp" and "heatmap".

(Counter objects, or, in fact, any pickle-able data type can be stored in a dataset!)

Due to the previous hash partitioning, this is a **parallel** operation!

timestamp, heatmap

(6) sort

Sort dataset in increasing timestamp order

This dataset is only one row per 20 minutes of data.

timestamp, heatmap

(7) movingsum

Compute a moving average of the heatmap column and store in a new column.

This is a **parallel** operation!

timestamp, heatmap, heatmap_filtered

(8) paintframes

Create graphical frames for all slices in **parallel.**

frame_xxxxx.png

(9) frames_to_mp4

Call ffmpeg to convert frames into a movie.

cab.mp4