Events CSV (events-csv) TODO

* Then go to Events CSV, and:

cdt  
  
csv -d from: '09/22/17 12:03:03' -o 45, 540 ./metrics.csv

* **Time Queries (from:, to:)**
  + For the time being I am using from:<timestamp> to:<timestamp> form. This is the simplest, and consistent with other types of query, but it will conflict with fields named “from” and “to”. Think of a way to resove the conflict: more semantics? Identity the fact that there’s a field with this name and … or identify timestamps after the keywords and …
  + Header handling – should be keep track of the last header, so we have more info about the timed events, or the self-contained information is sufficient.
  + The timestamp should be handled in a central location, probably in nu. Hook up with “Timestamp”. Do we want to implement heuristics on Timestamp so it recognizes a wide range of formats? Or we keep the format outside Timestamp? For the time being we we loaded a bunch of “supported” formats in TimeQuery and let it deal with it, but we can have this logic in Timestamp and make our live easier.
  + A time query should be pushed into the parser’s API, because we can optimize by not parsing complex events if they do not fall within the time window.
* **Workflow**

csv headers --last 'jhq0[1,4].\*mp-grams.\*onsumerCount' ./metrics.csv

csv --from <timestamp handled in a central location> --to <> -o <id-corresponding-to-mpgrams-before>, <id-corresponding-to-mpgrams-after>, …. <file>

With this I get the differences between active and standby and back during a failure event, and I can decide whether HA Franklin works.

* Refactor SimpleDateFormat everywhere, not thread safe.
* CSV vs Output  
  + CVSFormat, CVSFormatter –how those match with what’s now in Output.
* **CSV output in “processing” and “CSV”.** There is code that deals with CSV output both in events-processing/output and in the events-csv. Unify, one set of tests.
* **Reconcile ‘processing’ OutputFormat and ‘CSV’ CSVFormat,** when they are used together.
* **Databot has AsynchronousCsvLineWriter**, which also uses CSVFormatter. Reconcile.
* in-line graph