

A Quick Look at Metamarkets' Druid



Jay Wang
CS 394: Research Independent Study
21 March 2013

Background



- ❖ “Druid is the distributed, in-memory OLAP data store” Metamarkets built to serve their ad analytics needs
- ❖ “*Keeping everything in memory provides fast scans, but it does introduce a new problem: machine memory is limited.*”
– Metamarkets
- ❖ Solution: distribute data over multiple machines

Requirements

- ❖ The initial requirements for Druid were:
- ❖ (1) ability to load up, store, and query data sets in memory
- ❖ (2) parallelized architecture to allow more machines
- ❖ (3) parallelized queries to speed full scan processing
- ❖ (4) no dimensional tables to manage

Architecture



- ❖ Druid stores *beta* data, a more potent and compressed version of *alpha* data (raw, un-aggregated event logs), in memory
- ❖ Sample *beta* data set:

timestamp	publisher	advertiser	gender	country	impressions	clicks	revenue
2011-01-01T01:00:00Z	ultratrimfast.com	google.com	Male	USA	1800	25	15.70
2011-01-01T01:00:00Z	bieberfever.com	google.com	Male	USA	2912	42	29.18
2011-01-01T02:00:00Z	ultratrimfast.com	google.com	Male	UK	1953	17	17.31
2011-01-01T02:00:00Z	bieberfever.com	google.com	Male	UK	3194	170	34.01

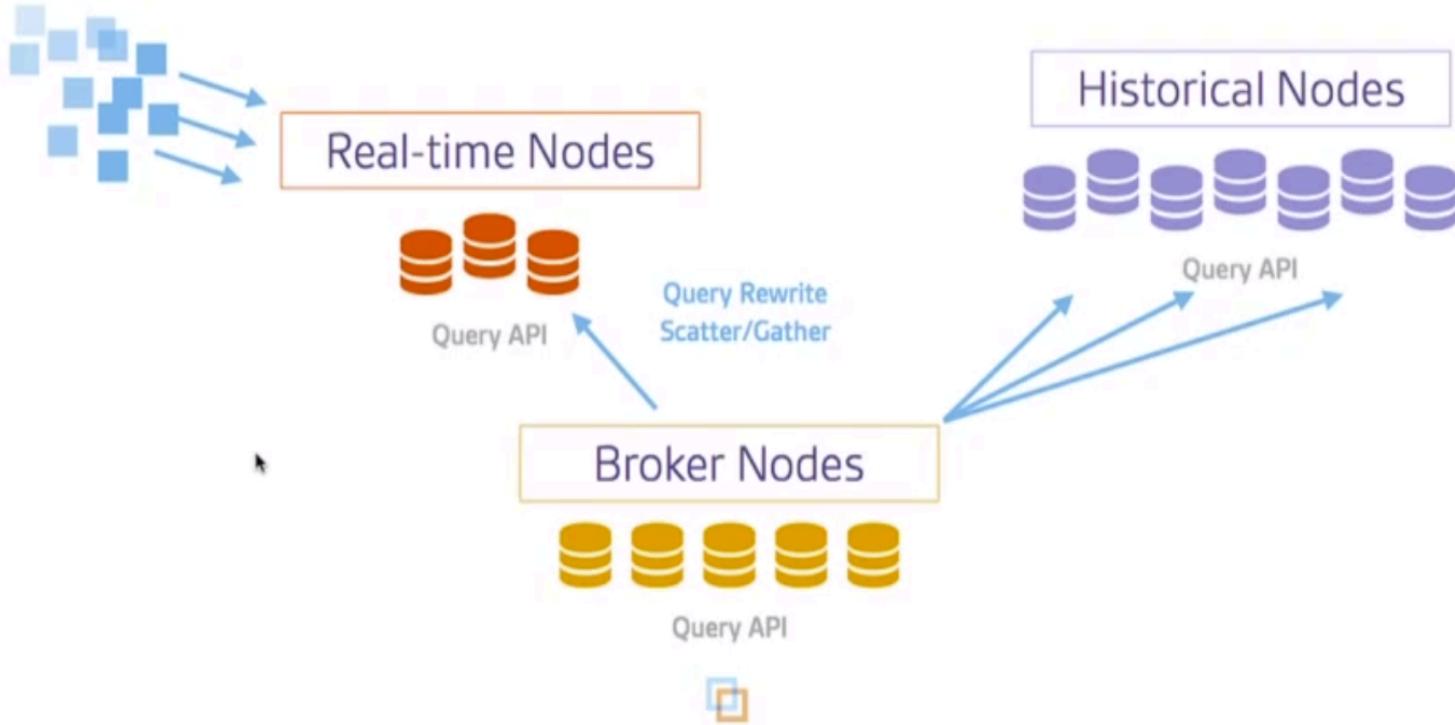
Realtime Ingestion



- ❖ Realtime Ingestion is done through Historical and Realtime nodes
- ❖ Both nodes have data for a time period they are responsible for (i.e. one hour)
- ❖ For example, a Realtime node would have data for one hour and that hour includes right now
- ❖ On some time bound, the Realtime nodes will push their data segments to the Historical nodes

Diagram

FEATURE 2 – REAL-TIME INGESTION



Node Types



- ❖ **Compute** nodes are the workhorses that handle storage and querying on historical data
- ❖ **Realtime** nodes ingest data in real-time and focus on making the stream of incoming data immediately queryable by the system
- ❖ **Master** nodes serve as the main coordinators, making sure that data is available, replicated, and in an “optimal” configuration
- ❖ **Broker** nodes understand the data layout across all of the other nodes in the cluster and re-write and route queries accordingly
- ❖ **Indexer** nodes load batch and real-time data into the system as well as allow for alterations to the data stored in the system (*as of Nov 2012, these nodes are still a work in progress. Realtime and HadoopDruidIndexer are the two entities handling indexing right now*)

Competing Products



- ❖ Druid sits between PowerDrill and Dremel on the spectrum of functionality
- ❖ Druid can do almost anything Dremel does (except that Dremel handles arbitrary nested structures whereas Druid only allows for single level of array nesting)
- ❖ Druid utilizes some data layout and compression methods from PowerDrill

Diagram



DRUID VERSUS OTHERS

- vs Google Dremel
 - No indexing structure
- vs Google PowerDrill
 - Close analog, all in-memory
- vs Hadoop+Avro+Hive(+Yarn)
 - Closer to Tenzig
 - Back-office use case

Use Cases



- ❖ Druid is *ideal* for products that require real-time ingestion of a single, large data stream
- ❖ Also a good fit if you are dealing with a no-downtime operation and are building your product on top of a time-oriented summarization of the incoming data stream
- ❖ If you care about query flexibility and raw data access, Druid is *not right* for you
- ❖ Druid is fast – queries that run in single-digit seconds over a 6 TB data set, but this is limited to certain types of queries and specific types of data sets (inflexible)

Metamarkets' Use Case



- ❖ Metamarkets' online advertising customers had data volumes upwards billions of events per month
- ❖ Thus, they needed two main functions:
 - ❖ perform highly interactive queries on the latest data
 - ❖ arbitrarily filter across any dimension (with data sets upwards thirty dimensions)
- ❖ Sample query – “find me how many advertisements were seen by female executives, aged 35 to 44, from the US, UK, and Canada reading sports blogs on the weekends.”

Netflix's Use Case



- ❖ Netflix has been testing Druid for operational monitoring of real-time metrics across their streaming business
- ❖ *“Netflix manages billions of streaming events each day, so we need a highly scalable data store for operational reporting. We are so far impressed with the speed and scalability of Druid, and are continuing to evaluate it for providing critical real-time transparency into our operational metrics”*

– Sudhir Tonse of Netflix

Quote from Eric Tschetter



- ❖ When asked about the best use cases for Druid...
- ❖ “*It’s very suited for slice-n-dice exploration of event streams. It is also hopefully potentially useful in other areas as well, but the thing we use it for is largely human-driven data exploration.*”

– Eric Tschetter, creator of Druid

- ❖ Ultimately, Druid’s functionality is that of any OLAP data store