Metrics and Cassandra

Tyler Hobbs
tyler@datastax.com
Why store metrics in Cassandra?

• Problems with alternatives
• Particular advantages of Cassandra
RDBMS Issues

• Random writes, random reads
  • Write amplification on SSDs
• Lock contention
• Availability
• Difficulty of horizontal scaling
RRDTool

• Same problems as RDBMS, lower throughput
OpenTSDB

- Same data model as Cassandra
- Master/Slave vs Fully Distributed
About Apache Cassandra

• Open Source
• Fully Distributed
• Non-Relational
• Log-structured Merge-Tree
Cassandra does Time-Series Data Very Well

- Sequential writes
- Mostly sequential reads
- Supports high parallelism
- Partitions data automatically, no distributed joins
- Block-based compression
Cassandra does Time-Series Data Very Well

- ~30k writes/sec per node
- Linear scalability
  - (see Netflix's 1,000,000 writes/sec benchmark)
CREATE TABLE metrics (  
  metric_id text,  
  time timestamp,  
  value float,  
  PRIMARY KEY (metric_id, time)  
) WITH CLUSTERING ORDER BY (time DESC);
CREATE TABLE metrics (  
  metric_id text,  
  time timestamp,  
  value float,  
  PRIMARY KEY (metric_id, time)  
) WITH CLUSTERING ORDER BY (time DESC);
Cassandra Schema

```
INSERT INTO metrics (metric_id, time, value)
VALUES ('node12-load', now(), 1.2);
```
Cassandra Schema

```
SELECT time, value FROM metrics
WHERE metric_id = 'node12-load'
AND time > '2012-11-28';
```
Metric Aggregation

- Write Time
- Read Time
Write-Time Aggregation

• Entirely Optional
• Good for reducing total volume of data stored
Write-Time Aggregation

• Primarily useful for rolling up a single metric at different time granularities
  • 1 min avg, 3 hour max, etc
• Use a strategy similar to RRDTool in memory
Read-Time Aggregation

- Complex analysis of data
- Read individual metrics, combine client-side
- Potentially store the results
Benchmarking Cassandra

- Simple API, few hidden costs
- Primarily benchmarking reads
  - Writes always have the same cost
  - Reads depend on data model quality, caching, and disk seek times
Benchmarked Cassandra

- Primary benchmarking mistake?
- Not enough client-side threads/processes.
Questions?

Tyler Hobbs
tyler@datastax.com