Exascale Data Analytics @Facebook

8/12/2014
ATPESC
Avery Ching
Make the world more open & connected
How are analytics used?
Men are talking about the economy; women are talking about inequality.
Recommended Pages

Get updates from your favorite businesses and brands.

Search for Pages...

Oakland International Airport
Airport Terminal
Michael Kong and 8 other friends were here.

Facebook Analog Research Laboratory
Organization
Carol Pai was here.

Sawa Sushi
Sushi Restaurant
Agata Zielinska and 3 other friends were here.

Fuki Sushi at Facebook
Sushi Restaurant
Alex Feinberg and 10 other friends were here.
>1.23B monthly active users
4.75B content items shared daily
350M photos uploaded daily
10B messages daily
Evolution of the Warehouse
2007

Relational Database =⇒ hadoop
2011

Sophisticated Hadoop Ecosystem

Log(Category, Event, Parameters)

Social
User content ➔ Transform ➔ Graph Search index
2014

600 TB/day → Warehouse → 10PB/day

300PB
Project 1 EB “soon”
Today
Data Ingestion

Scribe

Log line generated: <user_id, photo_id>

Log Storage
Log line reaches central storage (10s)

Copier-Loader
Log line reaches warehouse (1hr)

MySQL

Scrapes

Warehouse
Core Analytics

Hive
- Query Compiler
- Execution Engine

Corona
- Map-Reduce
- Job Scheduler
- Resource Manager

Data Pipelines

Social Graph
User Content
Transform
Graph Search
Real-Time Stream Processing

Log Storage → Puma

Stream processed (<1min)

HBase

Warehouse
Application Insights

Active Users

- **Daily Active People**
- **Weekly Active People**
- **Monthly Active People**
Real-Time Analysis

ODS

Scuba
Graph Analytics

Graph & ML Apps
Giraph Platform
Corona
Job Scheduler & Resource Manager
HDFS

https://giraph.apache.org/
Putting it all together
Scaling the Platform
Storage
Not all Data is Equal

Number of accesses/day

Partition Age
Data Lifecycle

Format: Encoding, Compression, Sorting

Replication: Erasure Codes

Hardware: Storage Hierarchy
Storage Format

Application Input Stream
Load/Transform

Hive Storage Format
RcFile

HDFS Files
High-availability

Table

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>101</td>
<td>111</td>
<td>121</td>
<td>131</td>
<td></td>
</tr>
<tr>
<td>102</td>
<td>112</td>
<td>122</td>
<td>132</td>
<td></td>
</tr>
<tr>
<td>103</td>
<td>113</td>
<td>123</td>
<td>133</td>
<td></td>
</tr>
<tr>
<td>104</td>
<td>114</td>
<td>124</td>
<td>134</td>
<td></td>
</tr>
<tr>
<td>105</td>
<td>115</td>
<td>105</td>
<td>135</td>
<td></td>
</tr>
</tbody>
</table>

RCFile

<table>
<thead>
<tr>
<th>Sync</th>
<th>Header</th>
</tr>
</thead>
<tbody>
<tr>
<td>101</td>
<td>102</td>
</tr>
<tr>
<td>111</td>
<td>112</td>
</tr>
<tr>
<td>121</td>
<td>122</td>
</tr>
<tr>
<td>131</td>
<td>132</td>
</tr>
</tbody>
</table>
## Column Encodings

<table>
<thead>
<tr>
<th>User</th>
<th>Action</th>
<th>Country</th>
<th>Index Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>18</td>
<td>like</td>
<td>2</td>
<td>comment</td>
</tr>
<tr>
<td>17</td>
<td>post</td>
<td>4</td>
<td>friend</td>
</tr>
<tr>
<td>90</td>
<td>like</td>
<td>2</td>
<td>like</td>
</tr>
<tr>
<td>38</td>
<td>comment</td>
<td>0</td>
<td>photo</td>
</tr>
<tr>
<td>23</td>
<td>comment</td>
<td>0</td>
<td>photo</td>
</tr>
<tr>
<td>42</td>
<td>like</td>
<td>2</td>
<td>post</td>
</tr>
</tbody>
</table>

*ORCFile developed in collaboration with HortonWorks*
In practice

- Type-agnostic Encodings
- Adaptive Encoding
  \[5x \rightarrow 8x\]
- CPU/Space tradeoff
Reader and Writer Performance

Reads and writes need to be fast

- Amortize Decisions
- Memory Management
- Lazy Operations
Experimental Results

https://github.com/facebook/hive-dwrf
Data Lifecycle

- **Hot Data**
- **Warm Data**
- **Cold Data**

**Replication:** Erasure Codes

**Format:** Encoding, Compression, Sorting

**Hardware:** Storage Hierarchy
Tying it all together

Simulated: 25 PB raw input data, 8% MOM growth
Base physical = RCFile + 3x replication
Optimized physical = ORCFile + Sorting + Erasure codes (applied gradually)
Compute
MapReduce == Easy parallel programming

Data partitioning / parallel processing
Scheduling with locality
Failure handling
MapReduce - Word count example

Map (load text lines from TextInputFormat)

```java
map(Object key, Text value) {
    StringTokenizer itr =
        new StringTokenizer(value.toString());
    while (itr.hasMoreTokens()) {
        context.write(new Text(itr.nextToken()),
                       new IntWritable(1));
    }
}
```

Input: “An elephant is an elephant”
Output:
{an, 1}
{elephant, 1}
{is, 1}
{an, 1}
{elephant, 1}

Reduce (Store text lines in TextOutputFormat)

```java
reduce(Text key, Iterable<IntWritable> values) {
    int sum = 0;
    for (IntWritable val : values) {
        sum += values.get();
    }
    context.write(key, new IntWritable(sum));
}
```

Input: Map output pairs
Output:
{an, 2}
{elephant, 2}
{is, 1}
Hadoop (v1)

Job Client → Job Tracker

Task Tracker 0
- M₀
- R₀
- M₁
- R₁

Task Tracker 1
- M₀
- R₀
- M₁
- R₁

Task Tracker 2
- M₀
- R₀
- M₁
- R₁

Resource Management

Job Management
Temporary Solution: Queue up Jobs

Please take a Number
Corona

Job Client → Cluster Mgr → Resource Management

Task Tracker 0
M₀  R₀
M₁  R₁

Task Tracker 1
M₀  R₀
M₁  R₁

Task Tracker 2
M₀  R₀
M₁  R₁
Corona

Job Client → Cluster Mgr → Resource Management

Task Tracker 0
1. Task Tracker
2. Task Tracker

J

Task Tracker 1

J

Task Tracker 2

J

Job Management
Multi-Tenancy Challenges

800+ users, 10’s of teams
Ads, BI, Data Science, Feed, Growth, Infrastructure etc.

bi.critical  bi.sla  bi.nonsla  bi.adhoc
Multi-Tenancy Challenges

What happens to resources when pool is under-utilized?

Pool Groups

Pool Priorities
Many other Improvements

- Resource Sandboxing
- Online Upgrades
- Restartable Job Trackers
Corona can scale to ...

- 4000+ node Clusters
- 120K+ Jobs/Day
- 20m+ Tasks Completed/Day
What’s next?
Multi-Temperate Data
Seamless Multi-Namespace Query
Distributed Machine Learning
Questions?