Extreme Analytics at eBay

Tom Fastner

XLDB
10/18/2011
eBay Analytics

>50 TB/day new data

>100 PB/day Processed

>100 Trillion pairs of information

>50k chains of logic

>100k data elements

>7500 business users & analysts

Active/Active

24x7x365 Always online

Turning over a TB every second

Millions of queries/day

Near-Real-time
Data Platforms

- **Structured SQL**
  - Production Data Warehousing
  - Large Concurrent User-base
  - Data Warehouse
  - Enterprise-class System
  - 6+PB

- **Semi-Structured SQL++**
  - Contextual-Complex Analytics
  - Deep, Seasonal, Consumable Data Sets
  - Data Warehouse + Behavioral
  - Low End Enterprise-class System
  - 40+PB

- **Unstructured Java/C**
  - Structure the Unstructured
  - Detect Patterns
  - Commodity Hardware System
  - 20+PB

- 500+ concurrent users
- 150+ concurrent users
- 5-10 concurrent users

Analyze & Report
Discover & Explore

Hadoop
Behavioral Data Flow

eBay Visitors > Application Server > CAL > Analytics Platform & Delivery > Analysts

- Application servers
- Central Application Logging (Tibco)
- Listeners
- Ingest Servers
- Singularity
- Hadoop
- DB
### Semi-Structured Data

<table>
<thead>
<tr>
<th>Start_dt</th>
<th>Guid</th>
<th>Sess_id</th>
<th>Page_id</th>
<th>Soj</th>
</tr>
</thead>
<tbody>
<tr>
<td>2011-10-18</td>
<td>1234</td>
<td>1</td>
<td>15</td>
<td>Language=en&amp; source=hp&amp; itm=i1,i2,i3,i4,i5</td>
</tr>
</tbody>
</table>

```sql
SELECT start_dt, guid, sess_id, page_id, 
NVL(e.soj, ‘itm’) AS item_list
FROM event e
WHERE e.start_dt = '2011-10-18'
AND e.page_id = 3286
/* Search Results */
```

<table>
<thead>
<tr>
<th>Start_dt</th>
<th>Guid</th>
<th>Sess_id</th>
<th>Page_id</th>
<th>Item_list</th>
</tr>
</thead>
<tbody>
<tr>
<td>2011-10-18</td>
<td>1234</td>
<td>1</td>
<td>15</td>
<td>i1,i2,i3,i4,i5</td>
</tr>
</tbody>
</table>
WITH event (start_dt, item_list) AS (previous SQL)
SELECT
    start_dt,
    item_id,
    /* Individual Item */
    count(*)
FROM
    TABLE ( /* Normalize comma delimited list */
        normalize_list( start_dt, item_list, ‘,’)
    ) RETURNS(start_dt, idx, item_id)
GROUP BY 1, 2
ORDER BY 3 DESC

* syntax simplified

<table>
<thead>
<tr>
<th>Start_dt</th>
<th>Item_id</th>
<th>Count(*)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2011-10-18</td>
<td>i1</td>
<td>555</td>
</tr>
<tr>
<td>2011-10-18</td>
<td>i2</td>
<td>444</td>
</tr>
<tr>
<td>2011-10-18</td>
<td>i3</td>
<td>333</td>
</tr>
<tr>
<td>2011-10-18</td>
<td>i4</td>
<td>222</td>
</tr>
<tr>
<td>2011-10-18</td>
<td>i5</td>
<td>111</td>
</tr>
</tbody>
</table>
Semi-Structured Data

Event Table

~ 4 Billion rows per Day
~ 2 Trillion rows in ~640 partitions (day)
~ 10,000 Tags

~ 40 Billion Search impressions per day
~ 1.2 PB compressed database space
~ 6 PB raw, uncompressed data

<table>
<thead>
<tr>
<th>Start_dt</th>
<th>Item_id</th>
<th>Count(*)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2011-10-18</td>
<td>1</td>
<td>~70,000</td>
</tr>
<tr>
<td>2011-10-18</td>
<td>~135</td>
<td>Million Items</td>
</tr>
<tr>
<td>2011-10-18</td>
<td></td>
<td>???</td>
</tr>
<tr>
<td>2011-10-18</td>
<td></td>
<td>???</td>
</tr>
<tr>
<td>2011-10-18</td>
<td></td>
<td>???</td>
</tr>
</tbody>
</table>
Benefits of Semi-Structure

• Data Modeling
  > No need to build out a complex model
  > Less vulnerable to changes in the model

• ETL
  > Simplified coding; less maintenance
  > Less vulnerable to changes
  > Less transformations
  > Faster loads
  > Better compression

• Processing
  > Eliminating joins (NVP = pre-joined!)
  > Having the context (rows) of “denormalized table” in one row available for processing
  > Potentially reduce the number of path over the data using a UDF vs. like Ordered Analytics
Path Analysis
Path Analysis

Diagram:
- Home → Search → View Item → View Item → Watch Item → Bid
- Home → MyeBay → Bid → Bid

Paths:
- H S V V W B
- H S V W W B
- H S V (2) W B
- H M B B
- H M B
- H M B (2)
### XPath Function

**TABLE (xpath(**

<table>
<thead>
<tr>
<th>new variant_type ( date, sessionid )</th>
<th>/* key columns */</th>
</tr>
</thead>
<tbody>
<tr>
<td>‘H*B’</td>
<td>/* regex pattern */</td>
</tr>
<tr>
<td>‘H: pageid=1,2,3,4&amp;B: pageid=5,6’</td>
<td>/* symbol definitions */</td>
</tr>
<tr>
<td>new variant_type ( pageid )</td>
<td>/* metric columns */</td>
</tr>
<tr>
<td></td>
<td>/* metric definitions */</td>
</tr>
<tr>
<td>‘list_count(pageid of *), list_collaps(pageid of *)’</td>
<td></td>
</tr>
<tr>
<td>RETURNS( date, sessionid</td>
<td>/* key columns */</td>
</tr>
<tr>
<td></td>
<td>/* metric definition columns */</td>
</tr>
<tr>
<td>, ‘m1=&lt;count&gt;&amp;m2=&lt;collapsed list&gt;’</td>
<td></td>
</tr>
<tr>
<td>)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Date</th>
<th>Sessionid</th>
<th>Metric definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>2011-10-02</td>
<td>1234</td>
<td>m1=HSV(2)WB&amp;m2=HSVVWB</td>
</tr>
<tr>
<td>2011-10-02</td>
<td>5678</td>
<td>m1=HMB(2)&amp;m2=HMBB</td>
</tr>
</tbody>
</table>
Compression (Block Level)

Compression Rate Histogram

IO Traffic by Day

Day 1 through Day 30

- Without Compression Read/Write MB
- With Compression Read/Write MB
SELECT Userid, Sessionid, Fraudscore
FROM TABLE (ExecMR('Athena',
'getfraudcandidates '2011-07-07'));

SELECT key, value
FROM mytable

Teradata.Reader reader = new Teradata.Reader(
FileSystem.get(prodTD, mySQL)
);
While (reader.next(key, value)) {
...
}
Platform Metrics for query 5

Table scan and summation

- **system unit cost (norm)**
  - EDW
  - Singularity
  - Hadoop

- **units consumed**
  - EDW
  - Singularity
  - Hadoop

- **job/query cost (norm)**
  - EDW
  - Singularity
  - Hadoop

- **latency**
  - EDW
  - Singularity
  - Hadoop

- **parallel efficiency**
  - EDW
  - Singularity
  - Hadoop
Analytics at eBay

Fueled by data, powered by people.
Tons of potential at your fingertips.

Predefined Early Binding Structured

Columnar
Cache/DataMart

Relational
Data Warehouse

Semi-Structured
Singularity

Flat file
Hadoop

Undefined Late Binding Unstructured