Moving and storing Big Data repeatedly for analytics

Old Way

Applications **retrieve** full (large) objects to their own (small) clusters for processing

Objects are internally partitioned for storage in parallel

Data loaded into warehouse is a fraction of data in high-tier, with high overhead for indexing to accelerate “expected” queries… NOT the unexpected ones
How storage protects data

Erasure Coding:

Data striped across multiple disks as data disks and parity disks

Designed way back... Then:
(logic was expensive, interconnect was relatively cheap
What that means for data reliably placed in storage: First 4 devices shown...

**Simple Table:**

<table>
<thead>
<tr>
<th>#1</th>
<th>#2</th>
<th>#3</th>
<th>#4</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSD</td>
<td>SSD</td>
<td>SSD</td>
<td>SSD</td>
</tr>
</tbody>
</table>

Bytes of data divided evenly across SSDs!

Data protection and streaming performance!

Supports data protection algorithms designed for HDD!
Simple Table:

<table>
<thead>
<tr>
<th>#1</th>
<th>#2</th>
<th>#3</th>
<th>#4</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSD</td>
<td>SSD</td>
<td>SSD</td>
<td>SSD</td>
</tr>
</tbody>
</table>

Bytes of data divided evenly across SSDs!

Data protection and streaming performance!

HDD-centric RAID/Erasure Coding prevent in-storage analytics

What that means for data reliably placed in storage: First 4 devices shown...

Simple Table:

<table>
<thead>
<tr>
<th>Device ID</th>
<th>SSD</th>
<th>HDD</th>
<th>Bytes</th>
<th>Type</th>
<th>Erasure Coding</th>
<th>RAID</th>
<th>Storage Analytics</th>
</tr>
</thead>
<tbody>
<tr>
<td>155190</td>
<td>1</td>
<td>1</td>
<td>21168.23</td>
<td>O</td>
<td>0.04</td>
<td>TRUCK</td>
<td></td>
</tr>
<tr>
<td>67310</td>
<td>1</td>
<td>2</td>
<td>45983.16</td>
<td>N</td>
<td>0.09</td>
<td>TRUCK</td>
<td></td>
</tr>
<tr>
<td>63700</td>
<td>3</td>
<td>3</td>
<td>13599.6</td>
<td>O</td>
<td>0.1</td>
<td>TRUCK</td>
<td></td>
</tr>
<tr>
<td>2132</td>
<td>4</td>
<td>4</td>
<td>28955.64</td>
<td>O</td>
<td>0.09</td>
<td>TRUCK</td>
<td></td>
</tr>
<tr>
<td>24027</td>
<td>5</td>
<td>5</td>
<td>22824.48</td>
<td>N</td>
<td>0.01</td>
<td>TRUCK</td>
<td></td>
</tr>
<tr>
<td>15635</td>
<td>6</td>
<td>6</td>
<td>49620.16</td>
<td>O</td>
<td>0.07</td>
<td>TRUCK</td>
<td></td>
</tr>
<tr>
<td>106170</td>
<td>1</td>
<td>7</td>
<td>44694.46</td>
<td>O</td>
<td>0.05</td>
<td>TRUCK</td>
<td></td>
</tr>
</tbody>
</table>

Copyright © 2023 AirMettle, Inc.
We divide (transparently) so we can conquer common analytical needs

- Data is unchanged for client
- Each internal component can be processed in parallel

AirMettle internal metadata enables parallel in-storage analytics

FIG. 3A

FIG. 3B

FIG. 4A

FIG. 4B
AirMettle Accelerates Multi-dimensional data

- Complex data format!
- Even the coordinates are multi-dimensional

- Scientific analysis also requires:
  - Sampling of different subsets for each query
  - Gathering statistical properties
AirMettle Accelerates Multi-dimensional data

- Complex data format!
- Even the coordinates are multi-dimensional

- Scientific analysis also requires:
  - Sampling of different subsets for each query
  - Gathering statistical properties

- Data is stored in partitions, based on semi-structural boundaries & size of segment
  - Enables distributed in-place parallel processing

  - Validated in Phase I

  - Queries are more complex than events w/ SQL
    - 2-3 Stage queries typical, but each stage can be executed in parallel
Selection & Re-scaling of weather data

Trials ~ mid’24

This work was supported with government support under

- 2135007 awarded by the National Science Foundation
- NA22OAR0210591 awarded by the National Oceanic and Atmospheric Administration
- NA23OAR0210342 awarded by the National Oceanic and Atmospheric Administration