Big Data Cloud Advisory System

Vladimir Suvorov, EMC
Vladimir.Suvorov@emc.com
Choosing between different Big Data technologies

• Different Big Data processing pipelines

• Different Technologies for same business tasks
Common way of benchmarking

- Compare similar technologies
  - Cloud and key-value stores
    - Yahoo Cloud serving benchmark
  - SQL-like technologies
    - Berkley Big Data Benchmarking
- Compare read/write rates (filesystem)
  - Teragen/Terasort/Teravalidate
- Compare real business cases
  - Impala vs Hive vs HAWQ
Our approach

Develop the unified architecture & prototype to perform any test against any technology and compare the results:

• Do not focus on benchmarks themselves
• Unified interface regardless of programming language
• Utilize virtualization to max extent
• Common and special performance
• Pipelines
• Scalability
VmWare Vsphere Layer

Virtualization helps to deal with all technologies from the resource allocation point

- VSphere, VDirector, VCenter
- Resource pools of the same size
- Vsphere API to get performance into DB
- Cloud cost model
- Dynamic resource allocation
- Other virtualization technologies
Technologies

- Manually/Automatically configured (Serengeti)
- CPU, OpenMPI, Hadoop, GraphChi, Cassandra, MongoDB
- Configuration obtained by connector and stored in the DB
- Provide Client API/ Interfaces/Libraries
- Separate CUDA cluster
Connector

- Utilizes SOAP protocol on one end and Technology API on the other
- No API = SSH + FTP
- Configure itself given Technology IP's
- Load Data and return link
- Execute task for some preloaded data
- Return Technology Health Status upon request
- Return results of the task, pathes & specific performance values – async mode
Control Server Layer

- Data Generator – generates custom data
- Task = prewritten executable file or code with parameters (data in/out directory required)
- Configurable Task Database
- Data pool
- Configuration & Performance DB
- Health monitor module
- CLI
Web service Layer

• Web-based cluster discovery
• Task configuration/ running/ monitoring
• Graph reports
• Pipelines
• Performance/cost reports
• Comparison tables
### Big Data Benchmark

<table>
<thead>
<tr>
<th>ID</th>
<th>Connector</th>
<th>Task</th>
<th>Files Count</th>
<th>File Size</th>
<th>Start at</th>
<th>Status</th>
<th>Finish at</th>
</tr>
</thead>
<tbody>
<tr>
<td>14</td>
<td>cpu</td>
<td>Substring search</td>
<td>5</td>
<td>60</td>
<td>2013-07-08 05:05:48</td>
<td>finished</td>
<td>2013-07-08 05:05:49</td>
</tr>
<tr>
<td>18</td>
<td>GraphChi</td>
<td>Substring search</td>
<td>3</td>
<td>1000</td>
<td>2013-07-08 05:16:10</td>
<td>finished</td>
<td>2013-07-08 05:16:12</td>
</tr>
<tr>
<td>20</td>
<td>cpu</td>
<td>Substring search</td>
<td>3</td>
<td>1000</td>
<td>2013-07-08 05:10:24</td>
<td>finished</td>
<td>2013-07-08 05:10:27</td>
</tr>
<tr>
<td>21</td>
<td>cassandra</td>
<td>Substring search</td>
<td>3</td>
<td>1000</td>
<td>2013-07-08 05:21:22</td>
<td>finished</td>
<td>2013-07-08 05:22:02</td>
</tr>
<tr>
<td>16</td>
<td>GraphChi</td>
<td>Substring search</td>
<td>3</td>
<td>900</td>
<td>2013-07-08 05:00:44</td>
<td>finished</td>
<td>2013-07-08 05:06:46</td>
</tr>
<tr>
<td>23</td>
<td>GraphChi</td>
<td>Substring search</td>
<td>3</td>
<td>500</td>
<td>2013-07-08 13:00:33</td>
<td>finished</td>
<td>2013-07-08 13:56:35</td>
</tr>
<tr>
<td>36</td>
<td>GraphChi</td>
<td>Substring search</td>
<td>3</td>
<td>500</td>
<td>2013-07-08 15:10:24</td>
<td>finished</td>
<td>2013-07-08 15:16:38</td>
</tr>
<tr>
<td>33</td>
<td>cassandra</td>
<td>Substring search</td>
<td>3</td>
<td>200</td>
<td>2013-07-08 14:05:24</td>
<td>failed</td>
<td>2013-07-08 15:18:01</td>
</tr>
<tr>
<td>34</td>
<td>cpu</td>
<td>Substring search</td>
<td>3</td>
<td>200</td>
<td>2013-07-08 14:05:50</td>
<td>finished</td>
<td>2013-07-08 14:55:52</td>
</tr>
<tr>
<td>35</td>
<td>cpu</td>
<td>Substring search</td>
<td>3</td>
<td>200</td>
<td>2013-07-08 15:07:37</td>
<td>finished</td>
<td>2013-07-08 15:07:40</td>
</tr>
</tbody>
</table>

Showing 1 to 10 of 34 entries