Abstract
Understanding and improving the performance and efficiency of HPC centers requires detailed analysis of running systems. To this end, modern HPC facilities provide extensive capabilities for collecting performance-related data for analysis. However, these data sources are often disparate from one another, measuring different components in different domains. It is not clear, for example, how to correlate per-rack temperature readings with mesh input sizes recorded for a particular physics simulation.

HPC Performance Data Sources
...in the hardware domain:

Network Data
- Rack x Time x Heat x Nodelist (sorted by heat)
- Rack, Node, temperature

Node Data
- Rack x Time x Heat x Joblist (sorted by heat)
- Rack id, heat, list of running jobs

Facility Data
- Facility temperature (9 sensors per rack, 23 racks)
- Facility temperature (9 sensors per rack, 23 racks)

...in the software domain:

Performance Data
- AMG generated the most heat
- AMG generated the most heat

Merging Disparate Data
Disparate data sources often require more advanced merging than a simple SQL JOIN operation.

Case: No one-to-one mapping
<table>
<thead>
<tr>
<th>Time</th>
<th>FLOP Counter</th>
<th>Temperature</th>
</tr>
</thead>
<tbody>
<tr>
<td>10:00</td>
<td>6453</td>
<td>55.6</td>
</tr>
<tr>
<td>10:01</td>
<td>55.6</td>
<td>58.2</td>
</tr>
</tbody>
</table>

Better merge
<table>
<thead>
<tr>
<th>Time</th>
<th>Temperature</th>
<th>FLOP Counter</th>
</tr>
</thead>
<tbody>
<tr>
<td>10:00</td>
<td>55.6</td>
<td>6453</td>
</tr>
<tr>
<td>10:01</td>
<td>58.2</td>
<td>786</td>
</tr>
</tbody>
</table>

Case: Same domain, different units
<table>
<thead>
<tr>
<th>Time</th>
<th>FLOP Counter</th>
<th>Temperature</th>
</tr>
</thead>
<tbody>
<tr>
<td>10:00</td>
<td>6453</td>
<td>55.6</td>
</tr>
<tr>
<td>10:01</td>
<td>55.6</td>
<td>58.2</td>
</tr>
</tbody>
</table>

Solution: Semantic Table

<table>
<thead>
<tr>
<th>Column</th>
<th>Units</th>
<th>Aggregator</th>
<th>Conversions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time</td>
<td>HH:MM</td>
<td>N/A</td>
<td>T =&gt; HH:MM</td>
</tr>
<tr>
<td>FLOP Counter</td>
<td>count</td>
<td>Sum</td>
<td>none</td>
</tr>
<tr>
<td>Temperature</td>
<td>Celsius</td>
<td>Average</td>
<td>C =&gt; F</td>
</tr>
</tbody>
</table>

This tells us:
1) If two data sources may be merged, and
2) how to merge them
And in turn:
3) possible datasets that may be produced by different sequences of merges (see below)

Preliminary Results
Dedicated Access Time (DAT) for 2 days on Cab (1296 nodes)

Data collected:
1. Job queue information (slurm)
2. Facility temperature (9 sensors per rack, 23 racks)
3. Facility layout (assignment of nodes to racks)

How much heat is generated by different jobs?

The SONAR Data Cluster
Above results used >8GB data (only 2 days worth)

Soon will be collecting continuous HPC performance data
- Power
- Temperature
- LDMS (counters on cores, uncore, and motherboard)

Need long-term massive storage, large-scale data processing

SONAR: newly deployed data cluster
- 13 nodes, SSDs, data software stack
- Apache Cassandra distributed database
- Apache Spark distributed data-local processing (used here)