"New" Astro Server Benchmarking

Background

The old astroserver was hosted by MySQL on a very modest server - 64-bit linux with 4Gb Memory, local hard-disks, and 4 cores.

The new server machine is 64-bit linux with 64Gb of Memory, a hefty Raid-5 array, and 8 cores.

With the aquisition of a new machine to host the astro-server we thought it would be wise to test new ways of arranging the event-data for faster querying.

We decided on three major 'improvements' to test:

1. Indexing data based on a spatial tesselation using Hierarchical Triangular Mesh (new) vs (old) indexing by Ra, Dec
2. Comparing performance of Oracle (new) vs MySQL
3. Partitioning of data

HTM Indexing

Gorey Details: http://skyserver.org/htm/HtmPrimer/tut_htm.html

We've used a degree of 6 which gives 2^15^ regions running from 2^15^ through 2^16^-1 (and fits in a 16-bit integer.)

Oracle vs MySQL

Oracle is freely available and supported at SLAC and has convenient developer tools for monitoring query load and optimizing queries and indexes.

MySQL is also freely available and has SLAC support but has less in the way of interactive developer tools for monitoring and optimization.

Partitioning

Data partitioning allows data to be physically separated so that a query requiring data in a subset of partitions can immediately disregard unneeded partitions. This can greatly improve query performance. Both Oracle and MySQL support sub-partitioning, which allows a further degree of data segmentation.

MySQL supports a maximum of 1024 (sub)-partitions on a single table, and has quite restrictive rules on the combination of partition / subpartition type. All partitions must be created manually.

Oracle supports an unbounded number of (sub)-partitions and allows for automatic allocation of new partitions as required. It has more partition types than MySQL and is more generous with the possible combinations of partitions and sub-partitions.

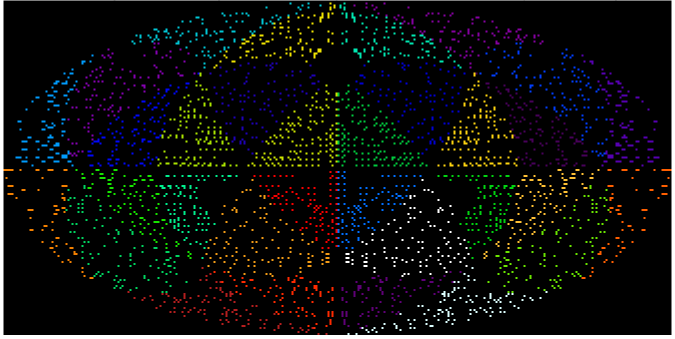
Partitioning Strategy - MySQL:

Because of the limit of 1024 partitions, we settled on partitioning by Time (2 weeks, per) and then by Event Class (1, 2, 3.) This will allow for 15 years of operation. Only the first 2 years of partitions (156 total) were initially created for testing.

Partitioning Strategy - Oracle:

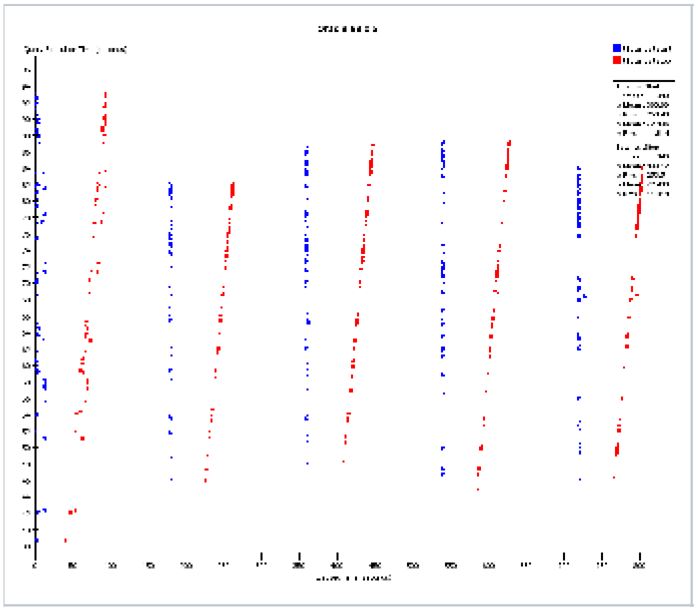
Data is partitioned by Time (1 week, per) and then by HTMID in 32 sub-partitions (1024 HTM regions per partition.) A single partition is intially allocated with a template for Oracle to create future partitions.

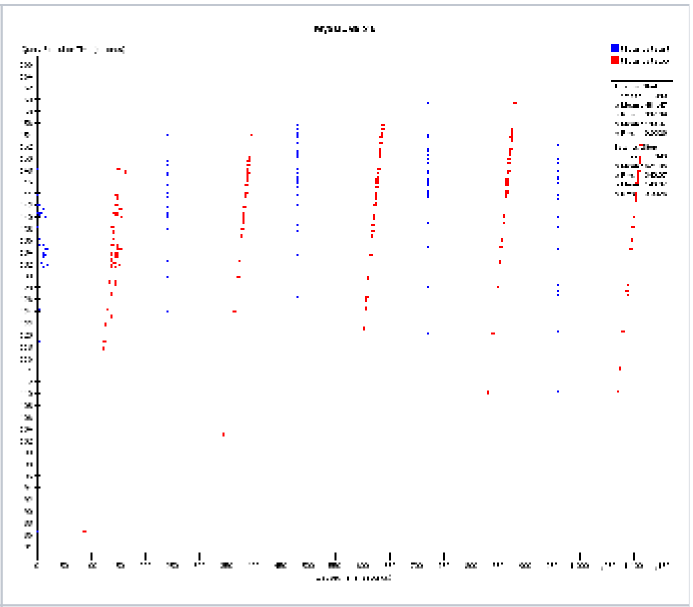
The HTMID partitions represent contiguous areas of the sky so that spatial queries can potentially be satisfied by a few partitions. Seth Digel was kind enough to generate the following plot:

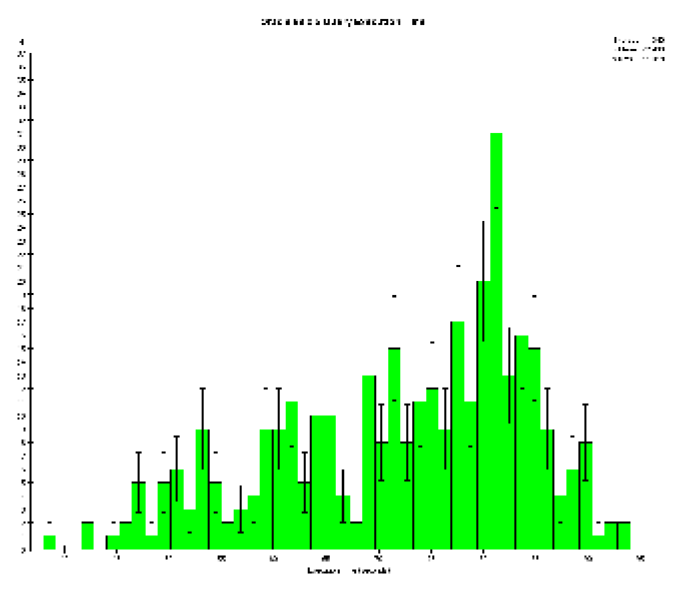


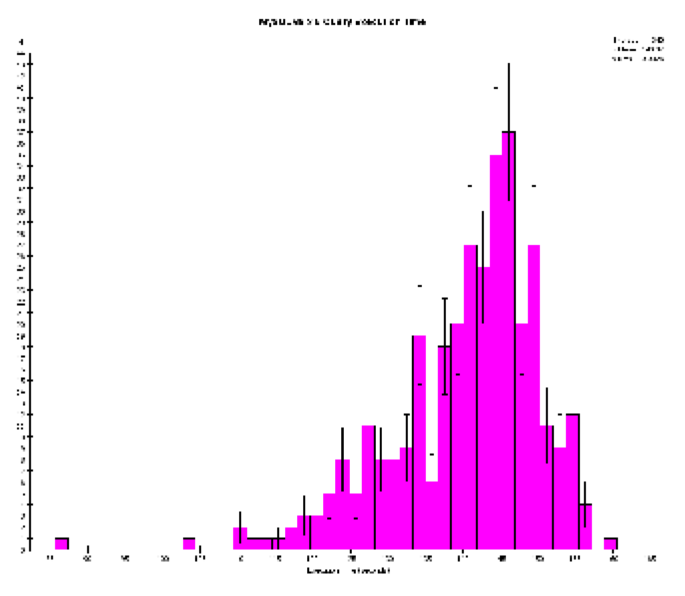
Results

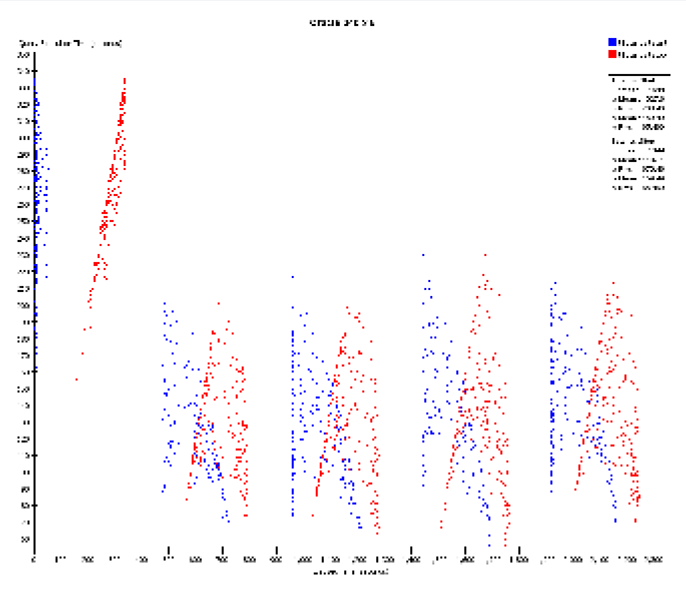
Oracle	MySQL

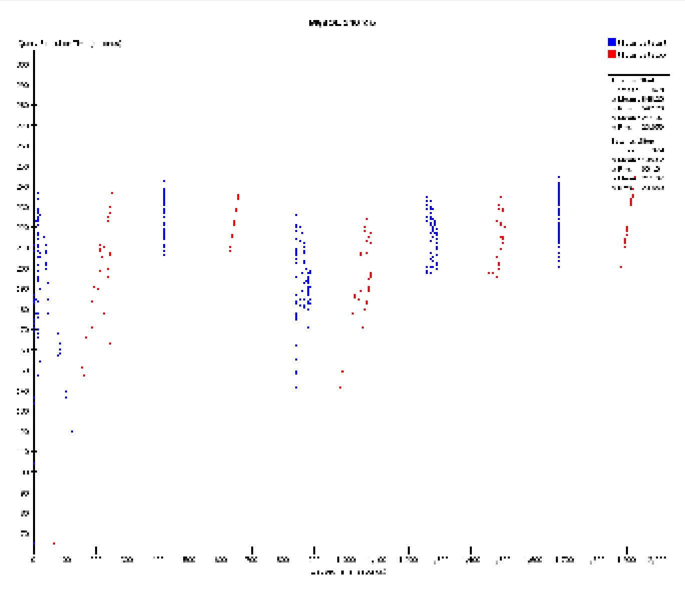


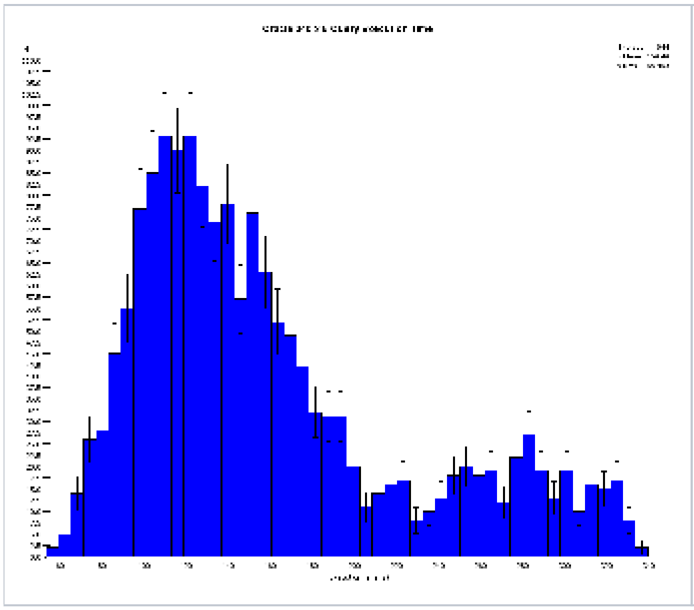


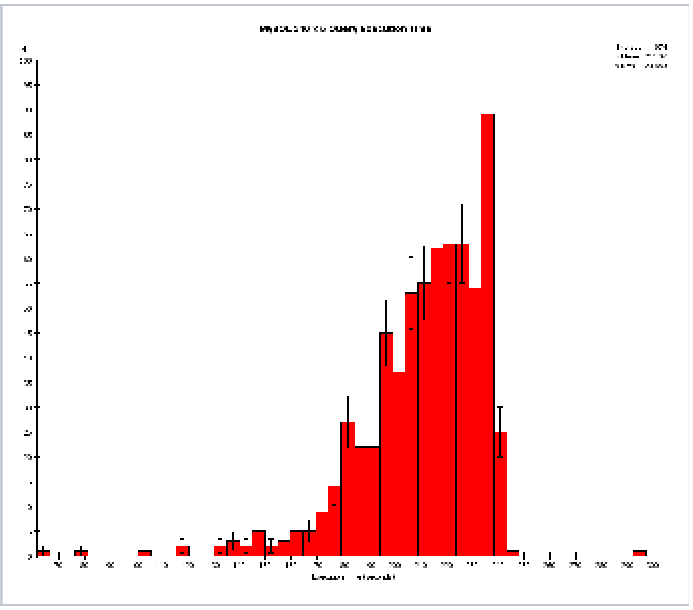









Oracle query execution time


MySQL query execution time