SkyServer Traffic Report: The SQL!

10 Years of SkyServer Web and SQL Logs

Ani Thakar
Jordan Raddick
Alex Szalay
(JHU)

and

Jim Gray
First Traffic Report

- **5-year report, 2006 MSTR**
  - Vik Singh, Jim Gray et al., covered 2001-2006

- **Highlights:**
  - Web & SQL traffic doubled every year
  - Hundreds of astronomers “graduated” from using canned and sample queries to free-form SQL
  - Flurry of activity after each release
  - Hard to separate bots & mortals reliably

<table>
<thead>
<tr>
<th>Users</th>
<th>Web Hits</th>
<th>SQL Queries</th>
<th>Web Sessions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total</td>
<td>65M</td>
<td>16M</td>
<td>3M</td>
</tr>
<tr>
<td>Mortals</td>
<td>27M</td>
<td>16M</td>
<td>1.5M</td>
</tr>
<tr>
<td>Spiders</td>
<td>14M</td>
<td>3M</td>
<td>1.4M</td>
</tr>
<tr>
<td>Bots</td>
<td>24M</td>
<td>14M</td>
<td>1M</td>
</tr>
</tbody>
</table>
10 Years of SkyServer Logs

- Sequel to the 5-year report
- In preparation
- Extend original analysis to new, larger dataset
- Separate the Web hits and SQL query analysis
- Focus more on the 10-year SQL usage data
- Unique dataset
- Best record of how new paradigm of data intensive science is being embraced by scientists
Research Questions for SQL Usage

- Who’s using SkyServer & CasJobs SQL?
- How often are they using it?
- How are they using it?
- Are they getting better at it?
  - How complex are their queries?
  - How do users learn SQL?
- What type of science is being done?
- Is it meeting the requirements?
- How can we improve the system?
- How effective is our online Help?
SQL Usage by the Numbers

<table>
<thead>
<tr>
<th>Total</th>
<th>Unique</th>
<th>Succeeded (error=0)</th>
<th>Failed (error != 0)</th>
</tr>
</thead>
<tbody>
<tr>
<td>194,023,591</td>
<td>67,946,073</td>
<td>145,002,755</td>
<td>49,020,836</td>
</tr>
</tbody>
</table>

- **Top 5 SQL users:**
  - All bots/programs
  - Big Brother (JHU monitor) is 2nd

<table>
<thead>
<tr>
<th>Client IP</th>
<th>Queries</th>
</tr>
</thead>
<tbody>
<tr>
<td>UVic/CADC</td>
<td>44.3 M</td>
</tr>
<tr>
<td>JHU (BB Monitor)</td>
<td>14.1 M</td>
</tr>
<tr>
<td>Berkeley</td>
<td>12.4 M</td>
</tr>
<tr>
<td>Japan</td>
<td>6.0 M</td>
</tr>
<tr>
<td>NRC (Canada)</td>
<td>5.4 M</td>
</tr>
</tbody>
</table>

- **Biggest single day: 37,097,351 queries!**
  - October 23, 2008 (close to DR7 release)
  - Nearly all from one IP (UVic/CADC)
  - Vast majority *failed*, only 1% actually succeeded (362k)
Query Complexity Index

• Length of Query (in bytes)
  – Naïve, doesn’t necessarily indicate an intelligent or sophisticated query
• JOINs: number and types of JOINs
• GROUP BY / ORDER BY
• CROSS JOINs, CROSS APPLYs, cursors
• Function calls (UDFs)
  – Depends on type(s) of function called
  – Good way to detect science use cases
• Combinations of some/all of the above
SQL Templates

• **Divide queries into templates:**
  - Focus on successful queries only (69M)
  - Regexp replace of all numbers with “#”
  - SELECT DISTINCT SQL statements
  - Assign each template a number (templateID)

• **Just under a million query templates**
  - From ~ 69 M unique queries
  - Derive a crude “complexity index”
  - Based on presence of SQL elements
    • JOINs (implicit/explicit/CROSS/OUTER,multiple)
    • GROUP BY, ORDER BY, cursors

• **Faster to query based on templates**
  - Avoids expensive text search through entire DB
INSERT SqlTemplate
SELECT
dbo.RegExReplace(
dbo.RegExReplace(
  dbo.RegExReplace(
    dbo.RegExReplace(
      dbo.RegExReplace(
        SUBSTRING(statement,
          PATINDEX(‘%select%’,
            statement), 9999),
          ‘(?<char>\W)(0x[0-9A-Fa-f]+|[+|-%][0-9.]+)’,
          ‘${char}#’
        )
      )
    ,’\s+’,’ ’
  )
  ,’\^[.]*\n\v/g’,’
  )
  ,’/\[-\w]\s*--\s.*\s\n/gm’,’
  )
  ,’/--\s*[^\r\n]\*/g’,’
  )
  ,’/\-{[^\r\n]}+/g’,’
  )
  ,0 AS hits,
COUNT(*) AS queries
FROM sqlstatement
WHERE
  statement LIKE ‘%select%from%’ AND NOT
(  statement LIKE ‘%delete%’
    OR statement LIKE ‘%drop%’
    OR statement LIKE ‘%create%’
    OR statement LIKE ‘%parseonly%’
    OR statement LIKE ‘%<a%’
    OR statement LIKE ‘%up_name%’
    OR statement LIKE ‘%batch%’
)
1) SELECT COUNT(*) FROM SqlTemplate WHERE template LIKE ‘%join%’
2) SELECT COUNT(*) FROM SqlStatement WHERE TemplateID IN
   (SELECT TemplateID FROM SqlTemplate WHERE template LIKE ‘%join%’)
Length of Query Templates

- Most queries: O(100) bytes in length
- \(\sim 35k \geq 1000\) bytes
  Less than 5%
- Bot and prog queries are usually small
- Limit on query length \(\sim 4k\) in the SkyServer
  (larger in CasJobs)
Fig. 11. Typical search queries from a German institute who search SDSS mapped areas only for high redshift objects.
Provenance and SQL Workflows

• Analysis of CasJobs queries by N.Li (2010)
• Study how users do data-driven analysis
  – Complex queries using MyDB system
• Number of MyDB objects per query
  – How many tables a workflow uses
  – Only ~ 20% of users with > 1 objects/query
• Number of MyDB object (table) dependencies
  – Objects created from queries on other MyDB objects
  – Better measure of workflow/complexity (?)
  – ~38% of users with > 1 dependency
  – These users responsible for 76% of query WFs
Next Steps

• Get templates for sample queries
  – How useful/effective are sample queries?
  – How many queries are just resubmitted samples?

• Refine complexity index further
  – Nested queries, use of UDFs and SPs
  – Track complexity as function of time

• Track SQL “sessions”
  – More relevant to CasJobs users
  – Multiple queries, use of variables etc.

• Use of built-in indices and HTM
  – Indexed columns, nearby functions, htmlID

• More detailed user demographics
SkyServer Traffic Page