FlatSQL vs. MySQL
Queries using Unix tools vs. a DBMS

Konrad Lorincz
Kevin Redwine
Jesse Tov

Harvard University
December 17, 2002
Dilemma: How to Perform Queries?

Options
- Use a simple search tool (grep)
- Use FlatSQL
- Use MySQL

Research Question
- How do you decide which option to choose?
Criteria

- **Ease of Use**
  - Text files – portable, easy to edit, existing Unix tools
  - Language – learn new language, use standard language

- **Performance**
  - Timing
What is FlatSQL

- **FlatSQL** is like SQL
  - Queries have the form "SELECT ___ FROM ____ WHERE ___"
  - There is added syntax for dealing with flat files
  - A lot of complexity has been removed

- Sample Query: `/etc/mtab`

```bash
$ flatsql SELECT m.1, m.2 \
  FROM   /etc/mtab AS m \
  WHERE  m.3 = "ext3"
```
Sample Query:
/etc/group and /etc/passwd

```
SELECT user
FROM   /etc/passwd WITH /:/
    AS p (user x uid gid gecos home shell),
    /etc/group WITH /:/
    AS g (group x gid memb)
WHERE  p.gid = g.gid
    AND group = "cs265"
```

Output
konrad
redwine
tov
**Query Translation**

```
SELECT user
FROM   /etc/passwd WITH /:/
    AS p (user x uid gid gecos home shell),
    /etc/group WITH /:/
    AS g (group x gid memb)
WHERE p.gid = g.gid AND group = "cs265"
```

```
awk "\{system("awk ""\$4 \"== \"cs265\" {print \"$1\"}BEGIN{FS=":\"}" /etc/group")\}BEGIN{FS=":\"}" /etc/passwd
```
Experiments

- **Selects** (One Relation)
  - Grep vs. FlatSQL vs. MySQL
  - indexed vs. non-indexed

- **Joins** (Multiple Relations)
  - FlatSQL vs. MySQL
  - indexed vs. non-indexed

- Start with **cold cache** each time
Select – 1 Relation
Indexed Data

Grep vs. FlatSQL vs. MySQL

<table>
<thead>
<tr>
<th>Nbr. of Lines</th>
<th>Time (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>1e+06</td>
<td>180.0</td>
</tr>
<tr>
<td>2e+06</td>
<td>160.0</td>
</tr>
<tr>
<td>3e+06</td>
<td>140.0</td>
</tr>
<tr>
<td>4e+06</td>
<td>120.0</td>
</tr>
<tr>
<td>5e+06</td>
<td>100.0</td>
</tr>
<tr>
<td>6e+06</td>
<td>80.0</td>
</tr>
<tr>
<td>7e+06</td>
<td>60.0</td>
</tr>
<tr>
<td>8e+06</td>
<td>40.0</td>
</tr>
<tr>
<td>9e+06</td>
<td>20.0</td>
</tr>
</tbody>
</table>

- FlatSQL
- Grep
- MySQL
Select – 1 Relation
Non-Indexed Data

Grep vs. FlatSQL vs. MySQL

Time (sec) vs. Nbr. of Lines

FlatSQL
Grep
MySQL
Join – 2 Relation
Indexed Data

FlatSQL vs. MySQL

![Graph showing the comparison between FlatSQL and MySQL in terms of time (sec) vs. Nbr. of Lines. The graph illustrates the performance difference between the two systems, with FlatSQL generally outperforming MySQL as the number of lines increases.]
'Join – 2 Relations
Non-Indexed Data'

The diagram compares FlatSQL vs. MySQL for join operations on non-indexed data. The x-axis represents the number of lines, while the y-axis shows time in seconds. The graph illustrates how the time increases with the number of lines, with FlatSQL consistently outperforming MySQL.
Conclusion

- Simple search tool (**grep**)  
  - 1 Relation and just regular expression matching  
  - Non Indexed – could be very large

- **FlatSQL**  
  - Comparison Operators  
  - Small joins – performance not critical  
  - Prefer to keep data in flat files

- **MySQL**  
  - Performance is Critical  
    - Data has to be indexed  
  - Multiple Joins