Real-World Performance Training
Team Exercise

Real-World Performance Team
Team Exercise

Briefing

• Tax authority of some government visits another country and hears about the potential value of running BI on their transactional records to help identify fraud and error, hence reducing evasion and improving collection. The tax authority is told that the system might pay for itself in a few months.

• Current system runs on mainframe and spends all its time processing the records in batch using a non-relational database; there is no ability to handle BI.

• The tax authority is interested in BI but still has to be able to perform the basic processing. Based on the supplied test data, the tax authority believes their mainframe system would take 1 hour to process the data. The tax authority converted the application to run on Oracle, and were disappointed to find it runs in 2 hours on a small development server. The tax authority believes the data should be processed in under 10 minutes to ensure the data is ready for use at the start of the day for BI. The customer has heard about Exadata for BI and will be willing to investigate Exadata for the entire system if Exadata can also handle the batch processing.
Team Exercise

Briefing

• “VAT clearing/matching”
  – Company A sells to company B
  – Company B buys from company A
  – Each report VAT

• “Buy” and “sell” side must be matched

• Batches received – always in sets of two (sell/buy)

• Possible errors
  – Duplicates
  – Out of sync records (buy and sell in different batches)”
Application workflow

Prematch done on done for both Buy and Sell

Input data

Matched earlier?

Yes

Duplicate error

No

Duplicate row?

Yes

Duplicate error

No

Prematch will contain leftover rows from last run
Application workflow

Matching

Prematch Buy

- Rows matching?
  - Yes: Merge rows
    - Move matched rows to matched table
    - Keep non-matched rows in prematch tables
  - No: Insert into MATCHED table

Prematch Sell

- Keep rows (do nothing)
Team Exercise

Application

• Application is written in PL/SQL
• 5 Pairs of Buy/Sell files provided
Team Exercise

 Tasks

1. Setup Schema
2. Load first pair of files and get a baseline
3. Quick fixes
Team Exercise

Setup Schema

• Database setup
  – A schema with the name teamXX has been created in the database for this exercise.
Team Exercise

Step 1: Setup Schema

• Load your schema:
  – cd to the `match` directory
  – Login as `teamXX/teamXX` and run the following scripts in this order.
    Make sure you do not run these as system!
    ```
    create_tables.sql
    create_external.sql
    create_views.sql
    matching.sql
    match_body.sql
    ```
Team Exercise

Step 2: Run application and get a baseline

• The `match.sh` script runs the application
  – It has two parameters
    • The first lists the batch to run
    • The second parameter is optional. If ‘init’ is specified, it truncates the tables at the beginning of the batch. Use this for only the first set of files.

• Execute `./match.sh "2001a_et" init`
  – Loads the first batch set
Team Exercise

Step 2: Run application and get a baseline

• Get a baseline for the match process to record how long it currently takes
• Use various tools to gather performance figures
  – Operating system commands
  – Database tools like EM, AWR, EMExpress
  – Tracing if you find it appropriate
Team Exercise
Step 3: Quick Fixes

• Implement quick fixes — keep these to only a few lines of code changes in the original PL/SQL
• Concentrate on loading the first set of flat files as fast as possible
• Record your performance investigations
  – Include time to load, changes made, issues, next steps
  – For the purposes of this exercise, include only the time to load files in the execution time. Do not include time to create tables, gather stats, etc.
# Team Exercise

**Five sets of Buy & Sell Records to Load**

<table>
<thead>
<tr>
<th>Set Name</th>
<th>Buy File</th>
<th>Sell File</th>
</tr>
</thead>
<tbody>
<tr>
<td>2001a_et</td>
<td>t1_2001a.dat</td>
<td>t2_2001a.dat</td>
</tr>
<tr>
<td>2001b_et</td>
<td>t1_2001b.dat</td>
<td>t2_2001b.dat</td>
</tr>
<tr>
<td>2001c_et</td>
<td>t1_2001c.dat</td>
<td>t2_2001c.dat</td>
</tr>
<tr>
<td>2001d_et</td>
<td>t1_2001d.dat</td>
<td>t2_2001d.dat</td>
</tr>
<tr>
<td>2001e_et</td>
<td>t1_2001e.dat</td>
<td>t2_2001e.dat</td>
</tr>
</tbody>
</table>
Team Exercise

Rules

- Must load each pair of files individually
  - The customer receives the files in pairs at different times and that’s how they are processed. We have 5 pairs of files for testing.
  - Cannot load all fives sets of files concurrently

- Cannot change CPU_COUNT parameter