PERFORMANCE OPTIMIZATION FOR LARGE SCALE LOGISTICS ERP SYSTEM

Santosh Kangane
Persistent Systems Ltd.
Pune, India

September 2013
Logistic System Overview

0.5 millions shipment / day
11000 Packages / day
183 millions shipments / year

24 Operations / shipment
6 Transit hops / shipment
2-4 day’s journey
System Deployment

- **Citrix Access Gateway**
- **Citrix Presentation Server**
- **Application Server**
- **WCF 8 CPU & 16 GB RAM**
- **SAN Storage**
- **Node 1**
- **Node 2**
- **Database Server (Oracle 11g R2 - RAC) Solaris 10**
  - 8 CPU & 32 GB RAM / Node
### Workload

<table>
<thead>
<tr>
<th>0.5 Million /Day</th>
<th>6 x ShipmentMaster</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ShipmentMaster</strong></td>
<td></td>
</tr>
<tr>
<td>PK</td>
<td>ShipmentID</td>
</tr>
<tr>
<td></td>
<td>ShipmentType</td>
</tr>
<tr>
<td></td>
<td>DataEntryOffice</td>
</tr>
<tr>
<td></td>
<td>ShipmentDate</td>
</tr>
<tr>
<td></td>
<td>ShipFrom</td>
</tr>
<tr>
<td></td>
<td>ShipToDestination</td>
</tr>
<tr>
<td></td>
<td>PresentLoc</td>
</tr>
<tr>
<td></td>
<td>...</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>TransitShipment</strong></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>PK</td>
<td>ShipmentID</td>
</tr>
<tr>
<td>PK</td>
<td>TransitDate</td>
</tr>
<tr>
<td>PK</td>
<td>TransitFrom</td>
</tr>
<tr>
<td>PK</td>
<td>TransitDestination</td>
</tr>
<tr>
<td>IsReceived</td>
<td>Varchar2</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>DataExportTracker</strong></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>PK</td>
<td>ShipmentID</td>
</tr>
<tr>
<td></td>
<td>DataEntryOffice</td>
</tr>
<tr>
<td></td>
<td>ShipmentDate</td>
</tr>
<tr>
<td></td>
<td>Region</td>
</tr>
<tr>
<td></td>
<td>IsDataSent</td>
</tr>
<tr>
<td></td>
<td>...</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>ShipmentHistory</strong></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>PK</td>
<td>EventID</td>
</tr>
<tr>
<td>PK</td>
<td>ShipmentID</td>
</tr>
<tr>
<td></td>
<td>Location</td>
</tr>
<tr>
<td></td>
<td>ShipmentStatus</td>
</tr>
<tr>
<td></td>
<td>EventDateTime</td>
</tr>
<tr>
<td></td>
<td>Operator</td>
</tr>
<tr>
<td></td>
<td>...</td>
</tr>
</tbody>
</table>

*Same as ShipmentMaster* | *24 x ShipmentMaster*
Performance Challenges

• System would hang for more than 5 min at unpredictable intervals
• Shipment tracking report, started taking up to 10 minutes to load
• Weekly and Monthly Analysis reports would runs for more than an hour
• Scheduled Billing Data Extraction job took 6-15 min to complete 1 iteration, Sometimes the job would take 45 to 60 minutes.
• Oracle Database Archive logs was abnormally high; 200-250 GB/day for total database size of 400GB
• All service points started taking extra 3-4 hour to complete their operations.

Further System rollout was stalled
Managing Data Growth

Problem:
• 500,000 shipments x 24 shipment operations per day
• 11,000 packages x 12 package operations per day
• Managing per day data growth was major concern
• Data was not archived since system went live (more than 1 year) and excessive data volume was appearing to be main cause for degraded system performance

Restriction:
• Oracle Table Partition is not allowed (Customer not willing to invest on Oracle Table Partitioning feature license)

2 Step Approach:
1. Remove excessive (old than 90 Days) data, One time activity
2. Setup data archive process, to keep data volume in control
Data Archive Strategy

Remove Excessive Data: One Time Activity done in planned 8 Hr System downtime. Mainly intended to remove excessive data and solve current performance problem

How we did it?

1. Export Data
2. Drop constraint & Indexes
3. Truncate data from all table
4. Import Latest Shipment data into OLTP tables
5. Create New schema, with exactly same structure as OLTP for Archived data
6. Import Older data in Archive schema
7. Create indexes & constraints
8. Update Database statistics
9. System Up in Action

46% of data was archived and has given some relief on system performance
Data Archive Strategy

Data Archival Job

- Table partitioning was not allowed
- No System down should be taken for data archival
- Archival process should not affect OLTP system performance
- Remove 500,000 shipments x 24 shipment operations and 11,000 packages x 12 package operations per day

Hence, we did it using Oracle PL-SQL.

The job had 2 parts,
1. Logic to find eligible shipment to be archived, based on complex business rules and save shipmentID’s in a table
2. Move to Archive
   - For all eligible shipment, select batch of 15,000 shipmentID’s
     - Append those shipment data to the archived database tables
     - Delete rows from the OLTP tables
     - Commit
   - Update Database statistics
Data Archive Strategy
Result:

• 500,000 shipments x 24 operations and 11,000 packages x 12 operations archived in 6 hours per day with minimal impact on OLTP performance
• No need to take system downtime for archival
• No need to pay substantially for advanced Oracle Table partition feature
Data Extraction Process

Shipment billing data was exported and sent to external MIS billing systems. Process was expected to run 24x7 at a 15 minute interval.

Problem:
- Takes 6-15 min to complete 1 iteration, Sometimes the job would take 45 to 60 minutes.
- Spill over has became a major drag in the business operations.
- High system resource utilization.

Risk in Changing:
Any error in calculating the results or incorrectly dropping data would have messed up the all India customer billing.
Data Extraction Process

- **For all Data Entry Offices in given region**
  - Select ShipmentID from DataExportTracker where DataEntryOffice, date and IsDataSent has value “NotProcessed”
  - Update all above rows to set IsDataSent = “InProcess” and commit.
    //logical lock on selected rows
  - Select data from different tables do calculation and transform it in format compatible to billing system.
  - Save selected data in Oracle global temporary tables.

- **Copy data from temporary tables to flat files**
- **Update all rows showing “InProcess” to “Processed”**
- **Delete data from Global temporary tables.**
Data Extraction Process

• For all Data Entry Offices in given region
  - Select ShipmentID from DataExportTracker where DataEntryOffice, date and IsDataSent has value “NotProcessed”
  - Update all above rows to set IsDataSent = “InProcess” and commit.
    //logical lock on selected rows
  - Select data from different tables do calculation and transform it in format compatible to billing system.
  - Save selected data in Oracle global temporary tables.

• Copy data from temporary tables to flat files
• Update all rows showing “InProcess” to “Processed”
• Delete data from Global temporary tables.
Data Extraction Process

Process actually took snapshot of data between last processed file till current time

---

**DataExportTracker**

<table>
<thead>
<tr>
<th>PK</th>
<th>ShipmentID</th>
<th>Varchar2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DataEntryOffice</td>
<td>Varchar2</td>
</tr>
<tr>
<td></td>
<td>ShipmentDate</td>
<td>Date</td>
</tr>
<tr>
<td></td>
<td>Region</td>
<td>Varchar2</td>
</tr>
<tr>
<td>IsDataSent</td>
<td>Varchar2</td>
<td></td>
</tr>
<tr>
<td>DataEntryTimestamp</td>
<td>Timestamp</td>
<td></td>
</tr>
</tbody>
</table>

**LastFileProcessed**

<table>
<thead>
<tr>
<th>PK</th>
<th>FileID</th>
<th>Number</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>lastTimestamp</td>
<td>Timestamp</td>
</tr>
<tr>
<td></td>
<td>CurrentTimestamp</td>
<td>Timestamp</td>
</tr>
<tr>
<td></td>
<td>Region</td>
<td>Timestamp</td>
</tr>
<tr>
<td></td>
<td>Varchar</td>
<td>Varchar</td>
</tr>
</tbody>
</table>
Data Extraction Process

- Find lastTimestamp from LastFileProcessed for given region
- Select ShipmentID from DataExportTracker with **criteria region code and timestamp between lastTimestamp and CurrentTimestamp**
- Select data from different tables do calculation and transform it in format compatible to billing system.
- Save selected data in Oracle global temporary tables.
- Save file information and CurrentTimestamp as lastTimestamp in LastFileProcessed
- Copy data from temporary tables to flat files
- Delete data from Global temporary tables.

<table>
<thead>
<tr>
<th>FileID</th>
<th>LastTimestamp</th>
<th>CurrentTimestamp</th>
<th>Region</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>PK</th>
<th>ShipmentID</th>
<th>DataEntryOffice</th>
<th>ShipmentDate</th>
<th>Region</th>
<th>IsDataSent</th>
<th>DataEntryTimestamp</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Varchar2</td>
<td>Varchar2</td>
<td>Date</td>
<td>Varchar2</td>
<td>Varchar2</td>
<td>Timestamp</td>
<td></td>
</tr>
</tbody>
</table>
Data Extraction Process

Optimization Result

- Data Extraction process completes in 3-6 seconds.
- Oracle Database Archive logs creation reduced by 25%.

Process completion time:

<table>
<thead>
<tr>
<th>Shipment Count</th>
<th>Region</th>
<th>Processes Start Timestamp</th>
<th>Processes End Timestamp</th>
</tr>
</thead>
<tbody>
<tr>
<td>28000</td>
<td>WEST</td>
<td>02:35:33:22</td>
<td>02:35:36:45</td>
</tr>
<tr>
<td>51000</td>
<td>EAST</td>
<td>04:36:13:14</td>
<td>04:36:17:45</td>
</tr>
<tr>
<td>14000</td>
<td>NORTH</td>
<td>04:38:02:69</td>
<td>04:38:06:93</td>
</tr>
</tbody>
</table>
Data Extraction Process

Timestamp Drift in Oracle RAC

27/9/2013 04:54:13
27/9/2013 04:53:03
27/9/2013 04:51:12
Data Extraction Process

Now, use *LastTimestamp* and calculated *CurrentTimestamp* to select data from OLTP tables.

- > 7
- 27/9/2013 04:48:03
- 27/9/2013 04:53:03
- 3 <
- 27/9/2013 04:51:12
- 27/9/2013 04:54:13
- 5 min
Application Dragging and Hang issue

System would hang for more than 5 min at unpredictable intervals

What's happening on Database ??
Application Dragging and Hang issue

What about Web Server??

Does Network shows packet drop??

• No packet loss on network
• Users on Local LAN too faces hang issue
However, for very short duration IIS process was showing quite high CPU utilization.

Application Dragging and Hang issue
Application Dragging and Hang issue

Web Garden
• By default, each Application pool in IIS contains a single worker process.
• Single Worker Process wasn’t able to scale.
• Hence Web Garden was enabled by changing IIS worker process count to 2.
Design optimization for reports

<table>
<thead>
<tr>
<th>ShipmentMaster</th>
<th>ShipmentHistory</th>
</tr>
</thead>
<tbody>
<tr>
<td>PK</td>
<td>PK</td>
</tr>
<tr>
<td>ShipmentID</td>
<td>EventID</td>
</tr>
<tr>
<td>ShipmentType</td>
<td>ShipmentID</td>
</tr>
<tr>
<td>DataEntryOffice</td>
<td>Location</td>
</tr>
<tr>
<td>ShipmentDate</td>
<td>ShipmentStatus</td>
</tr>
<tr>
<td>ShipFrom</td>
<td>EventDateTime</td>
</tr>
<tr>
<td>ShipToDestination</td>
<td>Oprator</td>
</tr>
<tr>
<td>PresentLoc</td>
<td>Number</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>Varchar2</td>
<td>Varchar2</td>
</tr>
<tr>
<td>Varchar2</td>
<td>Varchar2</td>
</tr>
<tr>
<td>Date</td>
<td>Date</td>
</tr>
<tr>
<td>Varchar2</td>
<td>Varchar2</td>
</tr>
<tr>
<td>Varchar2</td>
<td>Date</td>
</tr>
<tr>
<td>Varchar2</td>
<td>Varchar2</td>
</tr>
</tbody>
</table>

Computer Measurement Group, India
Design optimization for reports

1. Design Changes: How many shipments delivered Today?

~ 90 x 100,000

SELECT ShipmentID,
    ShipmentStatus,
    EventDateTime
FROM (SELECT hst.ShipmentID,
    hst.ShipmentStatus,
    hst. EventDateTime,
    ROWNUM as Rank,
    hst.Location as PresentLoc
FROM ShipmentHistory hst,
    ShipmentMaster mst
WHERE hst.ShipmentID =
    mst.ShipmentID
ORDER BY hst.EventDateTime DESC
)
WHERE Rank = 1
    AND trunc(EventDateTime) = trunc(sysdate)
    AND ShipmentStatus = "Delivered"
    AND PresentLoc = :v_userlocation

~ 24 x 900,000
Design optimization for reports

1. Design Changes: How many shipments delivered Today?

```sql
SELECT ShipmentID, ShipmentStatus, EventDateTime
FROM ShipmentMaster
WHERE trunc(EventDateTime) = trunc(sysdate)
  AND PresentLoc = :v_userlocation
  AND ShipmentStatus = "Delivered"
```
Row Lock Contention

Up to 65,000 shipment in a transit

Hub (Station) -> Hub (Station)

ShipmentsMaster
- PK: ShipmentID
- ShipmentType
- DataEntryOffice
- ShipmentDate
- ShipFrom
- ShipToDestination
- PresentLoc
- EventDateTime
- ShipmentStatus

ShipmentsHistory
- PK: ShipmentID
- Location
- ShipmentStatus
- EventDateTime
- Operator

TransitShipments
- PK: ShipmentID
- TransitDate
- TransitFrom
- TransitDestination
- IsRecived
Design optimization for reports

2. Code Optimization:
   - Use Metalized views instead of calculating results every time, refresh views in non-peak hours
   - For Complex business rules saved results in table and refresh data using PL-SQL jobs

3. Database Server Configuration:
   - Many of the master tables in ERP system are frequently read but are seldom updated. So enabled in memory Table cache in Oracle
   - To support cache memory requirement Oracle SGA size was increased from 8 to 16 GB
   - The IO activity on disk arrays it was found that, not all disks were equally loaded. Hence separated large volumes, frequently accessed tables and REDO logs onto a different disk array.
## Improvement in Reports

<table>
<thead>
<tr>
<th>Optimized process</th>
<th>Old Time</th>
<th>Improved Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shipment tracking reports</td>
<td>10 – 15 minutes</td>
<td>Less than 2 seconds</td>
</tr>
<tr>
<td>Analytical Monthly/Weekly reports</td>
<td>More than an hour</td>
<td>Less than 5 minutes</td>
</tr>
</tbody>
</table>
1. **Oracle WITH Statement Problem:**
   - Most of reports and application logic was implemented using Oracles WITH alias feature
   - We observed if WITH alias referred more than twice in SQL statement, it creates REDO as oracle materializes it in temporary table

   Ex : WITH DataA as ( SELECT ...)
   SELECT ... FROM DataA
   UNION ALL
   SELECT ... FROM SomeTable WHERE ( sub query using DataA)

   • All such queries reviewed and re-written with simplified logic

2. **Minimize use of Oracle Global Temporary Tables**

3. **Avoid multiple updates on same table in given transaction, if possible merge update statements in one.**

   **Reduction in Oracle Archive log creation by 30%**
### Achievements

<table>
<thead>
<tr>
<th>Problems</th>
<th>Before Optimization</th>
<th>After Optimization</th>
</tr>
</thead>
<tbody>
<tr>
<td>System hang</td>
<td>Unpredictably for 5-10 min</td>
<td>No hang issues</td>
</tr>
<tr>
<td>Data Extraction process</td>
<td>6-15 min</td>
<td>3-6 sec</td>
</tr>
<tr>
<td>Shipment tracking reports</td>
<td>10 – 15 min</td>
<td>Less than 2 sec</td>
</tr>
<tr>
<td>Monthly and Weekly analysis</td>
<td>More than an Hour</td>
<td>Less than 5 min</td>
</tr>
<tr>
<td>Oracle Archive Log</td>
<td>200-250 GB/Day</td>
<td>50-80GB/Day</td>
</tr>
<tr>
<td>Data Archival</td>
<td>---</td>
<td>6 Hours / Day</td>
</tr>
</tbody>
</table>

- Data volume is kept in control without investing substantially for Oracle costly features, Table Partitioning
- All office across the country are saving 3-4 Hour man-hours per day
- System has scaled from existing 1800 user base to 4000+ users
- System rolled out at all service points, Hubs and Warehouses successfully.