A PRACTITIONER'S GUIDE TO SUCCESSFULLY MIGRATE FROM ORACLE TO SYBASE ASE – PART 1

PETER DOBLER
PRESIDENT DOBLER CONSULTING
SEPTEMBER 14TH 2011
ABOUT PETER DOBLER

- Peter Dobler is an accomplished IT database professional, with 25+ years of experience, who delivers results by building sophisticated data systems; by implementing advanced application software; and, by improving efficiencies and reducing costs. He founded Dobler Consulting, (www.doblerconsulting.com) a Tampa, Florida consulting firm that delivers implementation expertise for Sybase, Oracle and MS SQL Server in 2000. Dobler Consulting is a Sybase System Integration partner and a premier partner for Sybase Professional Services. Peter designed and deployed large scale, high performance, enterprise level DW systems exceeding 40 TB and managed large IT groups of DBAs, System Engineers, Storage Engineering, and Security experts. He is a nationally recognized premier Sybase expert who frequently gets invited to speak at Sybase events. He can be reached at: mailto:pdobler@doblerconsulting.com or 813-322-3240.

ABOUT THIS PRESENTATION

• This session is part 1 of a 2 part session
  – Overview of the migration process
  – 7 step migration process walkthrough
  – Migration path
• Session 2 will cover the deep dive into the technical aspects.
  – Portability Check
  – Technical Challenges
  – Sample Code and Workarounds
MIGRATION MYTHS

Typical database migration myths are:

• Boardroom decision based on personal preferences -> financials
• Too complex -> more work -> financials
• No good reason for migration -> resistance -> more work -> financials
• Vendor A is better than vendor B -> financials
• Bottom-Line: Migration decisions are made by executives, not technologists
MIGRATION SCOPE DEFINITION

It is necessary to understand the underlying differences between Oracle and Sybase ASE. This knowledge will guide you through the migration process and is the foundation for a successful migration.

One important fact about comparing Oracle with Sybase ASE is to understand which part of Oracle will be compared with which Sybase product. Unlike Oracle, which integrates all its technology into one system, Sybase opted to separate these tasks with unique and specialized products that communicate well with each other.
## ORACLE AND SYBASE COMPARED

<table>
<thead>
<tr>
<th>Oracle</th>
<th>Sybase</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Database Server</td>
<td>Sybase ASE</td>
</tr>
<tr>
<td>Oracle OLAP and DW</td>
<td>Sybase IQ</td>
</tr>
<tr>
<td>Oracle Analytics</td>
<td>Sybase IQ</td>
</tr>
<tr>
<td>Oracle RAC</td>
<td>Sybase ASE Cluster Edition</td>
</tr>
<tr>
<td>Oracle Times Ten</td>
<td>Sybase ASE 15.5 In-Memory Database</td>
</tr>
<tr>
<td>Oracle Data Guard</td>
<td>Sybase Mirror Activator / Sybase Replication Server</td>
</tr>
<tr>
<td>Oracle Streams / Golden Gate</td>
<td>Sybase Replication Server</td>
</tr>
</tbody>
</table>
MIGRATION CHECKLIST

An Oracle to Sybase migration project should be divided into individual sub-tasks.

1. Planning the migration
2. Designing architecture migration
3. Implementing schema migration
4. Implementing data migration
5. Implementing application migration
6. Implementing operation migration
7. Validating the migration
PLANNING THE MIGRATION
PLANNING THE MIGRATION

An Oracle to Sybase migration project will contain the same project management components as any other project.

- Milestones
- WBS
- Resource allocations

The key to a successful migration is identifying possible show-stoppers before the project starts.
PLANNING THE MIGRATION

Identify Functionalities and Performing a Portability Check

• Identify all the Oracle specific functionalities that are being used in the application(s).
• Have a migration solution or workaround for every Oracle component in place.
• Perform a portability check on the entire environment and all applications. (Session 2 will provide a deep dive)
PLANNING THE MIGRATION

Identifying Incompatibilities

- Identify Oracle to Sybase migration incompatibilities. For example:
  - Object tables (Sybase supports Java object columns)
  - Pseudo Columns (ROWNUM, Flashback columns)
  - SQL Syntax (SQL extensions for OLAP, ORACLE SQL Syntax)
  - SQL*Plus
PLANNING THE MIGRATION

Identifying Possible Show Stoppers

• If incompatibilities have been identified, plan for additional time to rewrite the application.

• The main goal of the planning stage is to identify possible major code rewrites that increase the cost of the migration itself.
  • Can lead to deferring the migration project until major application re-design is due.
  • Can stop the project all-together.
DESIGNING ARCHITECTURE
MIGRATION
DESIGNING ARCHITECTURE MIGRATION

The architecture migration basically includes:

- Database definition and creation
  - Sizing
  - Security
  - Options
- Mapping of Oracle database elements to the Sybase ASE counterparts.
DESIGNING ARCHITECTURE MIGRATION

Major Differences Between Oracle and Sybase

• Major architecture differences between Oracle and Sybase are:
  • Concurrency and Multi-Versioning
  • Multi-Process vs. Multi-Thread Architecture
  • Instance and Database vs. dataserver and databases
  • Transaction Processing
  • ANSI SQL vs. T-SQL
DESIGNING ARCHITECTURE MIGRATION

Be Aware

• Concurrency and Multi-Versioning
  • The default transaction isolation level for both Oracle and Sybase ASE is isolation level 1 / Read Committed.
  • To mimic ANSI transaction handling and Oracle read versioning with the least code re-write, you can use set chained on and isolation level 3.
  • This creates a bad situation for Sybase ASE. Isolation level 3 in Sybase ASE creates a holdlock for every query, and that creates a cascading blocking effect.
IMPLEMENTING SCHEMA MIGRATION
IMPLEMENTING SCHEMA MIGRATION

In Oracle, a schema is equivalent to a user login. A schema always has a login, whereas a user login may or may not have a schema. An Oracle schema typically translates directly into a Sybase ASE user database. Pay close attention to synonym definitions, as they can spawn additional requirements in the schema migration.

It is very important to completely understand the security and user permission requirements on every schema. The security dependencies will guide the Sybase ASE user database design.
IMPLEMENTING SCHEMA MIGRATION

Data Types Limits

• The main focus on a schema migration is on data type conversions. For most Oracle data types, there is a Sybase ASE equivalent, with some minor exceptions:
  • CLOB and BLOB exceed the Sybase ASE max. allocation for TEXT and IMAGE
  • CHAR, VARCHAR2 and RAW exceed Sybase ASE maximum in a 2k page configuration.
  • VARCHAR2 exceeds Sybase ASE maximum in a 4k page configuration.
IMPLEMENTING SCHEMA MIGRATION

High-level Tasks

• Reverse engineering of all schemas
  • Always use the data dictionary stored in the production database.
  • Use **PowerDesigner** to reverse engineer schemas and convert data types
• Plan the object placement
• Create the new index strategy
• Migrate schema security and rules
IMPLEMENTING DATA MIGRATION
IMPLEMENTING DATA MIGRATION

The data migration task can be broken up into the following three subtasks:

1. **Planning**: It is important to understand the various options available for migrating the data, particularly the advantages and limitations of each of the options. This subtask involves evaluating the characteristics of the source data, and evaluating environmental and business constraints.

2. **Execution**: This subtask involves transferring the data using the Sybase data migration tool, migrating the Oracle database objects to Sybase database objects.

3. **Validation**: This subtask accounts for all the data validation and verifies data integrity.
IMPLEMENTING DATA MIGRATION

Planning

• The method of data migration is largely dependent on the size of the database, and down time granted for the data transfer.
• Low volume and/or long down times allow for flat file exports and bcp imports.
• High volume systems and/or short downtime requirements should use Sybase Replication Server for the data movement.
IMPLEMENTING DATA MIGRATION

Execution

- Oracle does not provide a tool for flat file export. Custom stored procedures need to be created or a 3rd party tool like FACT has to be used.
- Enterprise Connect Data Access (ECDA) allows for data transfer via proxy tables and by-passes flat files.
- Sybase Replication Server allows for up to last minute transaction synchronization with the shortest down time.
IMPLEMENTING DATA MIGRATION

Execution

• Alternatively Oracle tools, Oracle Streams or Golden Gate, for example, can be used for the data transfer as well.

• For large datasets it is highly recommended to use some form of data replication. But only Sybase Replication Server provides the Data Assurance option that allows the validation of source and target data. This will eliminate the need to write custom code to compare source and target database.
IMPLEMENTING DATA MIGRATION

Validation

• Before allowing access to a production system after a migration, careful data validation steps have to be implemented.

• Create custom scripts to compare the source database with the target database.
  • Or, utilize the Sybase Replication Server Data Assurance feature.
IMPLEMENTING DATA MIGRATION

With Full Data Load, Check for Bottlenecks and Prepare System

• Check the tuning of the database and memory usage settings applied
• Check for sufficient disk space for the expected size of the database 12 months ahead.
• Check for sufficient size of the tempdb database for handling temporary working storage, sort activity and for temporary user tables.
• Check for sufficient size of transaction log of the user database created.
• Prepare database monitoring activities
• Document database and server configuration settings
IMPLEMENTING APPLICATION MIGRATION
IMPLEMENTING APPLICATION MIGRATION

The application migration task is by far the largest and time consuming part of the entire migration project. The task can be divided into several application parts:

- Embedded SQL application
  - C or COBOL pre-compilers are doing 99% of the migration. Only the SQL has to be checked for compatibility.
- ODBC client application
  - The connection string changes, and possible stored procedure calls are changing too.
IMPLEMENTING APPLICATION MIGRATION

Continued...

• JDBC client application
  • Connection string changes
• Database-specific library application
  • Most difficult to migrate. Changing SQL*NET based API calls to CT-lib based API calls.
• C Applications
  • Migrate OCI calls to CT-lib calls.
• Oracle forms
  • Migrate to PowerBuilder Apps
IMPLEMENTING APPLICATION MIGRATION

Stored Procedures, Functions and Triggers

In addition to the application layer migration, stored procedures, functions and triggers have to be migrated as well. There are major differences between Oracle’s PL/SQL and Sybase T-SQL implementation that have to be accounted for. The most overlooked difference between PL/SQL and T-SQL is that PL/SQL checks each SQL statement for errors before proceeding to the next statement. T-SQL proceeds regardless of errors. This creates a lot of additional coding, which will be explained in Session 2.
IMPLEMENTING APPLICATION MIGRATION

Stored Procedures

PL/SQL vs. T-SQL

- Stored Procedures
  - Sybase T-SQL stored procedure can return an integer value; PL/SQL cannot.
  - PL/SQL contains pseudo data types like SYS_REFCURSOR, RECORD or TABLE. Views and temp tables have to be used in T-SQL to migrate these data types.
IMPLEMENTING APPLICATION MIGRATION

Functions

PL/SQL vs. T-SQL

• Functions
  • Most system PL/SQL functions exist in similar fashion in Sybase, but many have a different syntax and require rewriting of the SQL statements.
  • Many user defined PL/SQL functions can be migrated to T-SQL functions. PL/SQL functions that return an integer value can be migrated to a T-SQL stored procedure.
IMPLEMENTING APPLICATION MIGRATION

Triggers

PL/SQL vs. T-SQL

• Triggers

  • Oracle maintains BEFORE and AFTER STATEMENT triggers, as well as, BEFORE and AFTER ROW triggers.

  • Sybase ASE has only one after trigger, but all the Oracle trigger functionalities can be handled with the pseudo tables deleted and inserted. Oracle BEFORE STATEMENT triggers can be converted into rules or folded into the trigger code itself.
IMPLEMENTING APPLICATION MIGRATION

High-level Summary

• Address the transaction processing method change from ANSI to T-SQL
  • ANSI-SQL transactions start with either a insert, update or delete statement. T-SQL uses a begin transaction command to mark a transaction start.
  • Convert savepoints into nested transactions.
• Recreate stored procedures, user defined functions and triggers
  • Use of a 3rd party tool like SwisSQL is highly recommended
• Check all the SQL used in the application code for SQL language differences
• Check all the Java bases scripts for specific SQL
IMPLEMENTING APPLICATION MIGRATION

High-level Summary

• Check for business objectives and requirements met
• Verify the feed from other systems under load to ensure that system availability is unaffected at the time they run
• Check the security settings for the application set
• Check the required environment setting for the application
• Perform integration testing of the Application with third party system
• Determine the acceptable processing speed
IMPLEMENTING OPERATION MIGRATION
IMPLEMENTING OPERATION MIGRATION

Every production application contains a series of operational procedures and tasks that keep the application data secure and performing at its best.

Examples of operational tasks are:
- Backup and restore procedures
- Transaction log management
- Index maintenance tasks
- Data replication
VALIDATING THE MIGRATION
VALIDATING THE MIGRATION

Before opening the newly migrated application(s) to production use, a thorough migration validation has to be performed.

- Check for any system bottlenecks
- Count the number of objects in both the databases – Oracle and Sybase ASE
- Count the number of rows of each table in both databases
- Run the application under real world user load to simulate potential blocking contentions that could create a performance issue with transactions.
CONCLUSION
## MIGRATION PATH

<table>
<thead>
<tr>
<th>Full Migration</th>
<th>3rd Party Software / ERP Migration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Planning the migration</td>
<td>Planning the migration</td>
</tr>
<tr>
<td>Designing architecture migration</td>
<td></td>
</tr>
<tr>
<td>Implementing schema migration</td>
<td></td>
</tr>
<tr>
<td>Implementing data migration</td>
<td>Implementing data migration</td>
</tr>
<tr>
<td>Implementing application migration</td>
<td></td>
</tr>
<tr>
<td>Implementing operation migration</td>
<td>Implementing operation migration</td>
</tr>
<tr>
<td>Validating the migration</td>
<td>Validating the migration</td>
</tr>
</tbody>
</table>
SESSION SUMMARY

• You cannot control the why, but you have full control over the how.
• Find issues and show stoppers before you begin the migration project. This saves time, money and frustration.
• Create a matrix with all the matching functionalities that can be migrated via a search and replace method.
• Detect performance and blocking issues before going live.
• Utilize all the tool help you can get.
• Test, test and test some more.
SESSION 2 HIGHLIGHTS

The next session addresses:

• Oracle And Sybase ASE Compared
• Portability Check With Workarounds
• SQL Language Differences With Sample Code
• Application Migration Examples
• Data Migration Example
• Special Considerations
Q&A

Thank You.

Peter Dobler, Dobler Consulting
pdobler@doblerconsulting.com