Implementing Oracle Streams Replication: Tips and Techniques from Development

Patricia McElroy
Principal Product Manager
Oracle Streams
The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle’s products remains at the sole discretion of Oracle.
Agenda

- Streams Overview
- Best Practices for Implementing Streams
- Streams Active/Active Replication
- Custom Streams Processing
- Rolling Upgrade with Streams
- Summary
Oracle Streams

- Simple solution for information sharing
- Provides
  - Uniquely flexible information sharing
  - Active/Active replication
  - Real-time ETL for data warehouses
  - Availability during database migration
  - Availability during application upgrade
  - Message queuing
  - Event management and notification
Streams: Functional Components

Asynchronous Information Sharing Infrastructure
Streams Capture and Apply

Replication Configuration

Local (Upstream) Capture
If subset of data, network bandwidth requirements reduced significantly
Capture

- Low overhead, low latency change capture
  - Changes to the database are written to the online redo log
  - Oracle Streams extracts changes from the log as it is written
    - Log Buffer (11g), Redo, or Archived log files
    - Local or Downstream capture capability
  - Changes are formatted as a Logical Change Record (LCR), a representation of the change
- Background process
  - Auto restart (instance restart or RAC instance failover)
  - Automatic flow control
Logical Change Record (LCR)

- Database change = LCR
  - DML
    - Row change = LCR
    - Object name, owner, Type of DML, SCN,…
    - OLD, NEW values
    - Optional attributes: username, session, thread,…
  - DDL
    - Object name, owner, Type of DDL, SCN,…
    - DDL text
    - Optional attributes: username, session, thread,…
- LOB and LONG columns
  - Multiple LCRs per LOB or LONG column
  - Piecewise chunks
Staging

- Redo-based capture publishes messages into an in-memory staging area
  - Streams Pool of SGA
- Subscribers: other staging areas or Streams processes
- Messages remain in staging area until consumed by all subscribers
- Propagation between staging areas is scheduled through a propagation job
  - LCRs propagated from a queue in the local database to a queue in the target database
Apply

- The default apply engine will directly apply the DML or DDL represented in the LCR
  - Apply to local Oracle table or to non-Oracle table via dbLink

- Automatic conflict detection with optional resolution
  - Unresolved conflicts placed in error queue
  - Transactions can be re-applied or deleted from error queue

- Parallel apply maximizes concurrency

- Customizable apply processing
  - DML, DDL, Pre-commit, and Error handlers modify Apply processing
Rule-based Configuration

- Rule is expressed as SQL WHERE clause
  ```sql
  dbms_rule_adm.create_rule(
      rule_name=>'scott.rule1',
      condition=>' :dml.get_object_owner() = ''SCOTT'' AND :dml.get_object_name() = ''EMP''
  );
  ```

- Rule sets govern capture, staging, and apply
  - Inclusion
  - Negative

- Dynamic rule maintenance
  ```sql
  DBMS_STREAMS_ADM.ADD_TABLE_RULES
  DBMS_STREAMS_ADM.ADD_SCHEMA_RULES
  DBMS_STREAMS_ADM.ADD_GLOBAL_RULES
  ```
Rule-based Transformations

- Implicit datatype conversion during Apply (11g)
- Declarative Transformations
  - Rename schema, table, column
  - Add or delete column
- Custom Transformations
  - User-supplied PL/SQL function
- Specified on a rule for capture, propagation, or apply
- A single rule can have multiple transformations
Declarative Transformations

BEGIN
    DBMS_STREAMS_ADM.RENAME_SCHEMA(
        rule_name => 'STRMADMIN.HR51',
        from_schema_name => 'HR',
        to_schema_name => 'HR_REPL',
        step_number => 0,
        operation => 'ADD');
END;
/

SELECT rule_owner||'.'||rule_name rule,transform_type,from_schema_name,to_schema_name from
DBA_STREAMS_TRANSFORMATIONS;
RULE TYPE FROM TO
------------------------ ------------- ----------- ------------
STRMADMIN.HR51 DECLARATIVE TRANSFORMATION HR HR_REPL
Downstream Capture

Update EMP set job=‘coding’ where empid=510;

Redo Transport Method ASYNC or SYNC

Capture

LGWR

RFS

Apply

empid|job        |...
100 | sales      |...
510 | coding     |...

Offload Capture Processing From Production Database
BEGIN dbms_streams_adm.MAINTAIN_SCHEMAS(

    SCHEMA_NAMES       => 'HR,SCOTT',
    SOURCE_DATABASE    => NULL,     -- local database
    DESTINATION_DATABASE => 'TARGET_global_name',
    PERFORM_ACTIONS    => TRUE,
    BI_DIRECTIONAL     => TRUE,
    INSTANTIATION      => DBMS_STREAMS_ADM.INSTANTIATION_SCHEMA_NETWORK,
    SOURCE_DIRECTORY_OBJECT => null,
    DESTINATION_DIRECTORY_OBJECT  => null,
    SCRIPT_DIRECTORY_OBJECT => 'SCRIPT_DIR',
    SCRIPT_NAME        => 'generated_schemas_script.sql'
);

END;
/

Example Configurations using MAINTAIN_SCHEMAS

- Reporting
- Updatable
- Offload Prod
- Cascade

- Disseminate
- Consolidate
- Bidirectional
Streams in Oracle Database 11g

Improve Performance, Manageability

- Streams 11g Performance paper available on OTN
  - Demonstrates high performance throughput even across the WAN
- Performance Advisor identifies topology and analyzes end-to-end throughput.
- Split/Merge of Streams for Hub & Spoke replication
  - Maintains high performance for all replicas
  - Automated, fast “catch-up” for unavailable replica
- Table data comparison package
  - Compare data between tables in different databases
  - Identify differences by rowid
  - Converge differing rows
- Documentation: 2Day+ Data Replication & Integration
Streams Best Practices


10.2 Software Updates

- Use latest patchset of Oracle Database 10g Release 2 (10.2.0.4)
- MAA paper on *Oracle Streams Configuration Best Practices*
- Metalink article 418755.1 lists recommended patches and parameter settings for Oracle Streams
- Streams management via Enterprise Manager
  - MAINTENANCE tab of Enterprise Manager (10.2)

Watch OTN Streams website for custom software
- [http://otn.oracle.com/products/dataint/content.html](http://otn.oracle.com/products/dataint/content.html)
11.1 Software Updates

- Use latest patchset of Oracle Database 11g Release 1 (11.1.0.7)
- Streams management via Enterprise Manager
  - DATA MOVEMENT tab of Enterprise Manager

Watch OTN Streams website for custom software
- http://otn.oracle.com/products/dataint/content.html
General Configuration Tips

• Keep LCRs from each source database separate
• Separate queue for each capture/apply stream

• init.ora:
  - \_job\_queue\_interval=1 (set at the source site, indicates scan rate interval (seconds) of job queue) (10g recommendation)
  - streams\_pool\_size=200M (set at source and destination, ref. to V\$STREAMS\_POOL\_ADVICE)
  - aq\_tm\_processes should not be set to 0 or 10 in init.ora, spfile or by ALTER SYSTEM commands.

• For WANs: (SQL\*NET parameters)
  - Increase SDU (sqlnet.ora, tnsnames.ora, listener.ora)
  - Increase send\_buf\_size, recv\_buf\_size
  - Refer to MAA best practice paper: Oracle Streams Configuration Best Practices: Oracle Database 10g Release 2 Best Practices
Rule Tips

- Spelling counts!
- `source_database_name = GLOBAL_NAME` of source
  - Must be correctly specified for each process
- Eliminate duplicate or overlapping rules in a rule set
- Ensure that capture rules do not include objects with unsupported datatypes
  - `MAINTAIN_SCHEMAS` (and `GLOBAL`) avoid unsupported datatypes automatically
  - Use the negative rule set to eliminate specific tables or schemas
  - Use the `and_condition` clause on `ADD_SCHEMA_RULES` (or `GLOBAL`) to avoid tables with unsupported datatypes in 10.2 (`inclusion_rules=FALSE`)
    
    ```
    and_condition => ':lcr.get_compatible() > dbms_streams.compatible_10_2'
    ```
- Avoid complex rules
  - LIKE
  - Functions
  - NOT (9.2)
Recommended Streams Parameters

CAPTURE

- Reduce the value for `checkpoint_retention_time`
  - Minimize metadata stored within database
  - Automates move of FIRST_SCN of capture
  - Managed with `ALTER_CAPTURE` procedure

APPLY

- Modify parameters with `SET_PARAMETER` procedure
  - Enable parallelism to process multiple transactions at a time (based on transaction dependencies)
  - After initial setup and validation testing complete, allow apply to continue processing when user transaction fails with `DISABLE_ON_ERROR = N`
Managing Apply Errors

Enterprise Manager

Database instance: sample-view > Streams > Apply Errors: APPLY$_STRM_3

Select Local Transaction ID | Message Number | Message Count | Commit SCN | Source Database | Error Number | Error Message
--- | --- | --- | --- | --- | --- | ---
 | 5.17.1327 | 1 | 1 | 2244639 STRM_WORLD | 26796 | ORA-26796: A row with key "ORDER_NUMBER" (1028) exists but has conflicting columns "STATUS" in table DBO.ORDERS ORA-01403
as data source ORA-96512 at 'SYS.LCRS_ROW_RECORD', line 446 ORA-96512 at "DEMO.ORDERS_DML_HANDLER", line 1 ORA-96512 at line 1

Compare Values: APPLY$_STRM_3

- **Old Value**: shows the old column value in the row LCR. This is the column value for the row before the DML change.
- **New Value**: shows the new column value in the row LCR. This is the column value for the row after the DML change.
- **Current Value**: shows the column value in the row at the destination database. This is the row that will be modified by the row LCR when the apply process applies the change.

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Column Type</th>
<th>Old Value</th>
<th>New Value</th>
<th>Current Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>STATUS</td>
<td>SYS VARCHAR2</td>
<td>PENDING</td>
<td>Accepted</td>
<td>MAKERFAIL</td>
</tr>
<tr>
<td>ORDER_NUMBER</td>
<td>SYS VARCHAR2</td>
<td>1028</td>
<td>1029</td>
<td></td>
</tr>
<tr>
<td>DELIVERY_DATE</td>
<td>SYS DATE</td>
<td>20-SEP-08</td>
<td>1028</td>
<td></td>
</tr>
<tr>
<td>PART_NUMBER</td>
<td>SYS VARCHAR2</td>
<td>4455</td>
<td>4455</td>
<td>4455</td>
</tr>
</tbody>
</table>

Home | Targets | Deployments | Alerts | Compliance | Jobs | Reports | Setup | Preferences | Help | Logout
Streams Monitoring*

- Runtime views for statistics and process state
- STRMMON – MetaLink Note:290605.1 – Oracle Streams STRMMON Monitoring Utility
- UTLSPADV – 11g STRMMON replacement
  - Package in $ORACLE_HOME/rdbms/admin directory
- Healthcheck – MetaLink Note:273674.1 – Configuration Report and Health Check Script
- Alert Log –
  - Be aware of messages from the Capture process, logged when Capture sees at least 1 large or long-running transaction in its active transaction workload

*S298778 - Performance Analysis of Oracle Streams Configurations: Maximum Availability Architecture Best Practices, 1:30 - 2:30 pm, Moscone South Room 102
Message Tracking: Source Database

- Enable Message Tracking in Session
- Execute DML to monitor
- Monitor View from any session
  - Action identifies processing within component
  - Action_Details provides further information about that processing
Create A Heartbeat Table

- Quick status for DBA
- Generates activity in Database
- Implement periodic Job to update “heartbeat” table
  - 2 columns: Dbname, timestamp
- Configure Streams capture/apply for “heartbeat” table
- Configure Conflict Resolution method
  MAXIMUM or OVERWRITE
Periodic Maintenance

- DBMS_CAPTURE_ADM.BUILD()

- DBMS_CAPTURE_ADM.PREPARE_GLOBAL_INSTANTIATION()
  - Default prepare enables supplemental logging
  - To avoid supplemental logging configuration, use ‘NONE’ instead of default ‘KEYS’
Streams Active/Active Replication
Active/Active Replication

Multiple databases replicating data from shared tables and all replicas updatable

- Active/Active enables
  - Full Utilization of Hardware Investment
  - Site Autonomy & Distribution of Load
  - Geographically diverse locations

- Active/Active Considerations
  - Potential for Data Conflicts due to multiple updates to the same row at the same time at different databases
  - Appearance of Data loss (latency)
  - Application dependencies
Active/Active Replication = Streams

- Automatic Conflict Detection
- Optional Automatic Conflict Resolution (Oracle or user-supplied)
- Optimized for near real-time replication across WAN

• S300460 - Deploying Active-Active Data Centers Using Oracle Database Solutions, 9:00 - 10:00 am, Moscone South Room 103
Conflict Detection and Resolution

Deployment Tips

• Potential for updates to same row at same time in multiple locations
  • Compare “old” values in LCR to existing value at target
  • Supplemental logging for “old” values
• Avoid conflicts, as much as possible
  • Locally partition workload across geographic regions (ex: Follow the sun workload)
  • Partition applications across databases (ex: DB1 updates schema ABC, DB2 updates schema XYZ)
  • Ensure that user sessions are “sticky” (user affinity)
• Implement conflict resolution methods and handlers to automate error handling
  • For ACTIVE/ACTIVE use MAXIMUM with a TIMESTAMP column
  • MINIMUM, OVERWRITE, DISCARD possible for other scenarios
  • Error Handlers for more complex business requirements

Best Practices: Streams documentation and future white papers in MAA OTN site
Streams Active/Active

Deployment Tips

• If replicating DDL, perform DDL from single database
  • Consider impact of specific DDL across multiple databases
    • Example: Compiling procedures, functions
• For key columns generated from sequences:
  • Manage separate sequence ranges across databases
    • Change start and increment by settings
      • Odd/even
      • Last digit indicates database (modulo)
  • Use unique string in combination with sequence as key
• Avoid replication cycles by setting Streams Tag
  • Tag is null by default
  • Changes made by Streams Apply process have non-NULL tags
  • Rules indicate if tagged LCRs are replicated
  • Tag can be set in local user session to avoid replication of specific session changes.

Best Practices: Streams documentation and future white papers in MAA OTN site
Financial Services Company Using Streams, RAC, and Data Guard

- Streams
  - 2 node RAC
  - 3-way multimaster

- Data Guard for DR

Follow the Sun Workload

- Physical Standby Database
- Physical Standby Database
- Physical Standby Database

- AMER Database
- APAC Database
- EMEA Database

- Database A
- Database B
- Database C
Streams and Real Application Clusters

Deployment Considerations

• Streams Processes (Capture/Apply)
  - Processes run on “owning instance” of associated queue
  - If owning instance unavailable, queue ownership automatically moves to another instance and processes restart
  - Ownership can be assigned to specific instances (primary and secondary)
  - If multiple queue tables exist within database, each can run on separate instance

• Propagation between databases
  - Queue_to_queue (set to TRUE) propagation automates delivery to specific “owning instance” of target queue, using registered database service
  - Database service name based on queue name and global name of database
  - Oracle Net configuration expects global name of database in CONNECT_DATA service_name clause

Best Practices: Streams documentation and future white papers in MAA OTN site
Automotive Manufacturer
Oracle Streams Hub and Spoke Configuration

- **$100,000 - $200,000/hr per site savings in downtime costs**
- Fully bi-directional, automatic conflict detection and resolution
- Minimum subset of data replicated across WAN (about 1/3 out of 200 tables)

1 TB central engineering repository (hub) maintained at HQ

Replicas (spokes) at factories worldwide for fast, reliable, local access

Replicas synchronized with Streams

Physical standby for protecting central repository

*Case study available on OTN*
Streams Rolling Upgrade
Database Maintenance with Streams

- Migrating between Platforms or Character Set
- Interoperate between Database Releases
- Application Upgrades
- Database Customization

- Streams Configuration procedures
  - PRE_INSTANTIATION
  - POST_INSTANTIATION
  - CLEANUP
- Initial target database instantiation via RMAN, Physical Standby, or full exp/imp or Datapump
Extended Datatype Support (EDS)

- EDS package generates Streams configuration, table triggers and Dml Handlers to synchronize tables having these types
- For tables containing the following unsupported types:
  - Object columns with simple object types
  - Object tables
  - XMLtype
  - Varrays
  - Partial Spatial types (SDO_GEOMETRY)
- Metalink Article 556742.1
Streams Rolling Upgrade
Extended Datatype Support

insert into EMP values (1001, ‘Smith’, ‘Sales’, 42, sysdate, 30000, 10, 19);

Using EDS - Streams Rolling Upgrade

- Prior to Rolling Upgrade:
  - Load EDS package into Streams Administrator schema
  - Use the appropriate source setup procedure of EDS package
    - Generates scripts to create the logging table, base table trigger, and logging table trigger, and Streams directives on the source database
- During Rolling Upgrade:
  - Run generated scripts in documented order
  - Activate the physical standby as independent database (target)
    - Disable jobs at target and create database link to source database
  - Run the EDS destination setup procedure on source database
  - Perform generated scripts in documented order
  - Start Streams processes after upgrade/migration tasks complete
  - Switch users to new database when “caught up”
Managing Application Connections

- Create a service that is not managed via service_names init.ora parameter
- Configure application to use the service
- Manage service so that it is only started on the source database

- For reference: Client Failover in Data Guard Configurations for Highly Available Oracle Databases
Streams Custom Processing
Customizing the Apply Process

- Configure Streams for EMP table.
- Create custom PL/SQL procedure:
  - Can be used to execute SQL, perform lookups, modify data
  - LCR.EXECUTE performs the change in the LCR
  - Do not COMMIT within procedure (APPLY will commit at end of transaction)
  - Procedure should handle errors
- Register customized procedure with APPLY as DML Handler:
  - Register for EMP table
  - Error_handler=>FALSE
  - Table and operation specific

**Example:** Journal Table

```sql
Insert into EMP
values(1,'Josh Grob',...);
```

![Diagram](https://via.placeholder.com/150)

```sql
Redo Log
```

```sql
Capture
```

```sql
DML Handler
```

Store Primary Key, DML Type, Flag, Commit SCN in journal table

```sql
INSERT INTO scott.emp_journal
(empno, dml_type, flag, commit_scn)
VALUES (1, 'I', 'N', 43657832);
```
DML Handler Procedure

Logical Change Record Methods
DML Handler Procedure (cont)

---

ELIF cmntyp = 'UPDATE' THEN
   -- get old primary key. Old primary key can not be NULL
   ad := lcr.get_value('OLD', 'empno');
   IF (ad IS NOT NULL) THEN
      ret := ad.getnumber(o$empno);
   ELSE
      raise_application_error(-20000, 'Old primary key value can
       not be NULL');
      END IF;
   -- get new primary key values. For updates to non-primary key
   -- new primary key is null
   ad := lcr.get_value('NEW', 'empno', 'N');
   IF (ad IS NOT NULL) THEN
      ret := ad.getnumber(n$empno);
      key_chgd := TRUE;
   ELSE
      key_chgd := FALSE;
   END IF;

   IF key_chgd THEN
      IF key_changed, treat as delete/insert pairs
         o$cmntyp := 'D';
      ELSE
         o$cmntyp := 'U';
      END IF;
   END IF;

   INSERT INTO scott.emp_journal (empno, cmntyp, flag, commit
   VALUES (o$empno, o$cmntyp, '0', n$cscn);

   INSERT INTO scott.emp_journal (empno, cmntyp, flag, commit
   VALUES (n$empno, 'I', 'N', n$cscn);
---

-- Register Handler with Apply

-- set apply dml handler for scott.emp
BEGIN
   -- set apply dml handler
   dbms_apply_ads.set_dml_handler(
      object_name => 'scott.emp',
      object_type => 'TABLE',
      operation_name => 'DEFAULT',
      error_handler => FALSE,
      user_procedure => 'emp_handler',
      apply_database_link => NULL);
   /
   select * from dba_apply_dml_handlers;

   set instantiation SCN for scott.emp
   DECLARE
      iscn NUMBER;
   BEGIN
      iscn := dbms_flashback.get_system_change_number();
      dbms_apply_ads.set_table_instantiation SCN(
         source_object_name => 'scott.emp',
         source_database_name => 'scott',
         instantiation SCN => iscn);
   END;

   select * from dba_apply_instantiated_objects;
---
Think Outside the Box

Combine multiple features with Streams

- Triggers
- Dynamic SQL
- Flashback
- LCR modification
- DML & DDL handlers, Pre-commit handlers

Sample code available

- Procedural Replication
- Asynchronous Triggers
- Streams driven Materialized View Refresh
- Streams driven Continuous Query
- Streams Capture and Streams AQ
Summary
Oracle Streams

- Integrated Feature of the database
- Active/Active capabilities
  - Full Utilization of Hardware Investment
  - Site Autonomy & Distribution of Load
  - Geographically diverse locations
- Information Sharing between applications and databases
  - N-way multimaster (bi-directional replication), hub & spoke, many to one replication, custom capture/apply
  - Support data transformations, subsetting, custom apply functions
  - Support heterogeneous platforms and different character sets
  - Interoperate between database releases
- Reduce planned downtime
  - During database platform migrations
  - During application upgrades, when the logical structure of the database is changed
Resources

- Collateral, best practices, sample code for both Streams replication and Advanced Queuing: http://otn.oracle.com/products/dataint/

- MetaLink Note 418755.1 – 10.2 Streams Recommendations
### HA Sessions, Labs, Demos From Oracle Development

**Mon, Sep 22**
- 2:30 pm - Database 11g: Next-Gen HA, Moscone South 103

**Tue, Sep 23**
- 9:00 am - Active-Active Data Centers, Moscone South 103
- 11:30 am - Sharding with Oracle, Moscone South 302
- 11:30 am - HA with Oracle VM, Moscone West 3024
- 1:00 pm - Active Data Guard, Moscone South 104

**Wed, Sep 24**
- 9:00 am - Fusion Middleware Grid HA, Marriott Nob Hill AB
- 11:30 am - RMAN Best Practices, Moscone South 103
- 5:00 pm - Data Guard & Real Application Testing, Moscone 102
- 5:00 pm - EM in Secure MAA, Moscone West 2001

**Wed, Sep 24 (contd.)**
- 5:00 pm - E-Business Suite HA, Moscone West 2002/04

**Thu, Sep 25**
- 9:00 am - Oracle Secure Backup, Moscone South 102
- 10:30 am - Streams Replication, Moscone South 102
- 12:00 pm - Rolling Database Upgrades, Moscone South 103
- 1:30 pm - Streams Performance, Moscone South 102
- 3:00 pm - Oracle Grid Computing, Moscone South 303
- 3:00 pm - E-Business Suite R12 MAA, Moscone West 2007
- 3:00 pm - Siebel MAA, Moscone South 308
- 3:00 pm - Fusion SOA HA & Scalability, Marriott Salon 14/15

### Hands On Labs - Thu, Sep 25
- 10:30 - 11:30 am, 12:00 - 1:00 pm - Active Data Guard, Marriott Golden Gate A3

### DEMOgrunds, Mon-Thu
- Active Data Guard, Streams, Oracle Secure Backup, RMAN/Flashback, MAA
For More Information

search.oracle.com

or

oracle.com