mertechdata

Technical FAQ: Btrieve – Oracle

Migration and Deployment

Contents

Introduction

Mertech's Btr2SQL database migration tool smoothly migrates a Btrieve database to an Oracle back-end. The migration process creates the required tables and indexes and copies data to the Oracle server. This white paper answers frequently asked questions about the migration and deployment process.

What objects are created on Oracle?

Indexes

If a table has at least one index that is not unique and the table does not have a primary key defined in the DDFs:

- The *MDS_RECNUM* column is a primary key. *MDS_RECNUM* is added as the last segment to any index that is not already unique. Thus, **all** indexes are created as unique. This guarantees proper record traversal forward and backwards through otherwise duplicate values.
- A sequence (*<tablename>_S0*) is used to increment the *MDS_RECNUM* value. The INSERT command uses *sequence.NEXTVAL* to set the field value.

If a table has a primary key and has indexes that are not unique, the segments of the primary key are appended to the non-unique keys to make them unique.

Auto-Increment fields

The Mertech driver supports auto-increment fields and Btrieve auto-increment fields are created on the server. A sequence (*<tablename_S<fieldnum>*) is used to increment the unique value. The INSERT command uses *sequence.NEXTVAL* to set the field value.

Triggers

- If Btr2SQL creates a table with an auto-increment column and you use Mertech's driver, the Mertech driver manages the auto-increment values. However, if there are inserts into the table outside of the Mertech driver you need to enable a trigger that performs the necessary operations for auto-increment handling.
- To support descending index segments and case sensitive indexes, the Mertech driver makes use of a special key called an inverse key. Btr2SQL creates inverse keys during data migration. Any updates to index fields in a migrated table require that you also update the inverse key correctly, much the same as in auto-increment handling.
- You can use the Btr2SQL *Enable Trigger to Handle Auto-Increment and Inverse Key* option to enable the trigger to handle both auto-increment and inverse key values. The trigger is named <tablename>_T.

Application Locks

Btrieve allows an application to lock records outside of a transaction. This is counter-intuitive to an SQL database and conflicts with normal transaction processing. Therefore, the standard server record locks cannot be used. Instead, the Mertech driver utilizes the *DBMS_LOCKS* package and manages its own record locks.

Locks are acquired through *dbms_lock.request* and individual locks are released with *dbms_lock.release*. When a table is closed or a transaction is ended, a large number of locks may be needed to be released. This falls to the *mds_release_locks_v1* stored procedure , which is automatically created by the migration utility. If you create the database manually, you need to create this stored procedure:

```
CREATE OR REPLACE PROCEDURE mds_release_locks_v1
(p_lockList IN varchar2, p_numReleased OUT PLS_INTEGER)
AUTHID DEFINER AS
    -- p_lockList is a comma separated list of lock IDs
    -- p_numReleased returns the number of locks that released successfully

    lockId varchar2(50);
    startPos int := 1;
    endPos int;
    lockStatus int;
    BEGIN
        p_numReleased := 0;
        while startPos <= length(p_lockList) loop
            -- parse lock list
            endPos := instr(p_lockList, ',', startPos);
            if endPos = 0 then endPos := length(p_lockList)+1; end if;
            lockId := substr(p_lockList, startPos, endPos - startPos);
            startPos := endPos + 1;

            -- Do the release
            lockStatus := dbms_lock.release(to_number(lockId));

            ---- Give some feedback
            --SELECT DECODE(lockStatus,
            --  0,'Released',
            --  3, 'Parameter Error',
            --  4, 'Not owned',
            --  5, 'Illegal Lock Handle')
            --  INTO lockStatus_s FROM dual;
            --dbms_output.put_line(lockId || ': ' || lockStatus_s);

            if (lockStatus = 0) then
                p_numReleased := p_numReleased + 1;
            end if;
        end loop;

    END;  -- mds_release_locks_v1()

grant execute on mds_release_locks_v1 to public

drop public synonym mds_release_locks_v1

create public synonym mds_release_locks_v1 for sys.mds_release_locks_v1
```

License counting view

mds_session_info is created as a view against V$SESSION. Select rights on *mds_session_info* must be given to the user running the application but not to the underlying system table.

```
create or replace view mds_session_info as
select username, machine, terminal, module from v$session

grant select on mds_session_info to public

drop public synonym mds_session_info

create public synonym mds_session_info for sys.mds_session_info
```

Can I use different table spaces?

The Btr2SQL migration tool allows you to choose different table spaces for index and data tables. You can also choose to migrate some tables to one tablespace and other tables to another tablespace. If more control (such as a separate tablespace for BLOBs) is needed, we suggest that you:

1. Use the Btr2SQL *Generate* | *SQL Script for Creating Tables* option
2. Modify the generated script as needed
3. Run the modified script in SQL*Plus or another tool
4. Use Btr2SQL to copy the data

Which files are used?

For each Btrieve data file, a corresponding interface file (.INT) is created. For example, 'filename_ext.INT' is generated for 'filename.ext'. The INT file contains metadata used at runtime and resides in the location where the original data file was located. The original Btrieve data file is no longer needed.

The Pervasive.SQL/Btrieve engine is only used by the migration tool. The Pervasive.SQL engine can be shut down after the data migration is performed.

What roles and privileges are required?

Runtime

A user requires the following roles and privileges to run deployed applications and access the Oracle database at runtime:

- Roles: CONNECT, RESOURCE
- System privileges: "SELECT ANY DICTIONARY", "SELECT ANY TABLE"
- A user also needs "UNLIMITED TABLESPACE" or a proper quota set.
- *GRANT EXECUTE ON "SYS"."DBMS_LOCK" TO "<user>"* (The DBMS_LOCK package must be installed, and it is by default)
- If the application will be creating tables on the fly (temp tables for instance), the user needs object creation rights. Application logic can be altered to login as a different user for these temp tables.

Optional Runtime

The driver makes queries against V$PARAMETER to establish some internal parameters. If access to the system view is not available, default values are used instead.

To retrieve the maximum number of cursors that can be open:

```
select value from v$parameter where name = 'open_cursors'
```

To retrieve the db_block_size value used for the storage calculation:

```
select value from v$parameter where name = 'db_block_size'
```

To retrieve the numeric characters defined at the server to be used as a separator:

```
select value from v$parameter where name = 'nls_numeric_characters'
```

Migration

The user doing a migration needs more rights to the database than the user running the application at deployment.

BTR2SQL needs access to V$PARAMETER view to retrieve information for storage estimation. This is the query:

```
select value from v$parameter where name = 'db_block_size'
```

Can I determine the number of licenses used?

You can use the following query to see how many licenses are in use by the Mertech driver. If this number exceeds the count provided by the license file, the driver will fail to run:

```
select count(distinct(machine)) from mds_session_info where module = 'ORA_BTR - SN:<your
serial number>'
```

How do I install the client?

Each workstation accessing the SQL database must have an Oracle client installed. The Instant Client is the smallest and easiest to install. A good starting point to find the right client is here: http://www.oracle.com/technology/software/index.html.

What things should I to look for in my code?

Look for calls to B_CREATE and / or file deletion through OS calls.

B_CREATE is supported; however the table will not be fully defined in SQL. This can be overcome with application modification to utilize either the *MdsAddTable* function or by using an INT file Template (for additional information see the Btr2SQL blog titled *Using BTR2SQL's PERMANENT_INT option with B_CREATE*).

Many apps create "temp" files for sorting and other short-lived purposes or create new files each quarter/year (for additional information see the Btr2SQL blog titled *Support for Temporary Data Files*). Instead of removing the temp file using an OS call, utilize *MdsDropTable* (or *B_DROP_FILE*).

There is also a *B_TRUNCATE_FILE* API if the application simply wants to delete all the records. This is much faster than the usual Btrieve method of read/delete each record and avoids dropping the table which may have adverse effects on the defined tablespaces.

How do I know if I am targeting Btrieve or SQL?

The normal approach to access the Btrieve DLL is either through the import library provided by Pervasive or a LoadLibrary("w3btrv7.dll"). Windows finds the dll on the PATH and loads it. How can you tell if the Pervasive or the Mertech dll was loaded?

Use 'B_EXTENDED_VERSION' (op 5026) to retrieve information about the Mertech driver or attached SQL back-end. The Extended Version function can do two things: retrieve information about the Mertech driver (version, dll name, etc.) or retrieve a version string from the active SQL server. Making this call against the Pervasive access dll causes a return code of 1, which is a useful way to know if your application is running against a Btrieve or a SQL back-end.

Can the application control login?

B_SQL_LOGIN and *B_SQL_LOGOUT* allow the application to avoid the login dialog and fully hide the user credentials used for accessing the database. This is documented in the SDK portion of the User Guide.

Can the application execute queries?

B_SQL_* functions allow execution of SQL queries and results retrieval on the same connection as the driver. This is documented in the SDK portion of the User Guide.

How do I look at the data on the server?

Now that the Pervasive Control Center is gone, how do I look at the data on the server? There are many tools which enable the user to execute queries, edit data, modify table structures, import/export, etc.

Oracle makes a tool called SQL Developer - http://www.oracle.com/technology/software/products/sql/index.html and there are also many tools available for purchase online. One such tool is Aqua Data Studio - http://www.aquafold.com/

Contact Information

If you would like to know more about Mertech's products, please visit our website www.mertechdata.com or contact us at:

Corporate Head Office

Mertech Data Systems, Inc.
18503 Pines Boulevard, Suite 312
Pembroke Pines, FL 33029
USA
Tel: +1 (954) 585 9016
Fax: +1 (866) 228 1213

California Office

Mertech Data Systems, Inc.
114 East Shaw Avenue, Suite 100
Fresno, CA 93710
USA