The **Full Transportable Export/Import (FTEX)** method is an efficient way to migrate an entire database from one platform to another, especially between platforms with different operating systems (e.g., Windows to Linux). It combines the benefits of transportable tablespaces with Data Pump for non-transportable data. Here's a detailed guide:

**Prerequisites**

1. **Database Version**: Both the source (Windows) and target (Linux) databases must be Oracle 11.2.0.3 or higher (Oracle 19c is supported).
2. **Compatible Character Sets**: Ensure the source and target databases have compatible character sets.
3. **Endianness**:
   * Check the endian format of the source and target platforms:
   * SELECT \* FROM v$transportable\_platform;
   * If the endian formats differ, you need to convert the tablespace files using RMAN.
4. **Target Database**: Ensure the target database is already created and operational.

**Step-by-Step Process**

**1. Prepare the Source Database**

**A. List the Tablespaces**

Identify all user tablespaces to be transported:

SELECT tablespace\_name FROM dba\_tablespaces WHERE tablespace\_name NOT IN ('SYSTEM', 'SYSAUX', 'TEMP', 'UNDO');

**B. Put Tablespaces in Read-Only Mode**

Transportable tablespaces must be in read-only mode:

ALTER TABLESPACE USERS READ ONLY;

ALTER TABLESPACE DATA\_TBS READ ONLY;

**2. Perform the Full Transportable Export**

Use **Data Pump Export (expdp)** to export metadata and data for the entire database.

**Command**:

expdp "sys/oracle as sysdba" directory=DATAPUMP\_DIR dumpfile=full\_transportable\_%U.dmp logfile=full\_transportable\_export.log full=y transportable=always

**Options**:

* full=y: Indicates a full database export.
* transportable=always: Specifies that transportable tablespaces are used.
* dumpfile: Output dump file(s) for metadata and non-transportable data.

**Expected Output (Log File)**:

Starting "SYS"."SYS\_EXPORT\_TRANSPORTABLE\_01": ...

Processing object type TRANSPORTABLE\_EXPORT/PLUGTS\_BLK

Processing object type TRANSPORTABLE\_EXPORT/TABLE

Processing object type TRANSPORTABLE\_EXPORT/INDEX

. . exported "HR"."EMPLOYEES" ...

Exported tablespaces: USERS, DATA\_TBS

**3. Copy Datafiles and Dump Files to the Target**

**A. Identify Datafiles**

Find the datafiles of the tablespaces:

SELECT file\_name FROM dba\_data\_files WHERE tablespace\_name IN ('USERS', 'DATA\_TBS');

**B. Transfer Files**

Copy the datafiles and dump files from the source (Windows) to the target (Linux). Use scp, rsync, or any file transfer tool.

**Example Commands**:

scp C:\ORACLE\ORADATA\PRODDB\USERS01.DBF oracle@linux-server:/u01/oradata/PRODDB/

scp C:\ORACLE\DATAPUMP\full\_transportable\_%U.dmp oracle@linux-server:/u01/datapump/

**4. Prepare the Target Database**

**A. Create Directory Object**

Create a directory object in the target database for the dump files:

CREATE OR REPLACE DIRECTORY DATAPUMP\_DIR AS '/u01/datapump';

GRANT READ, WRITE ON DIRECTORY DATAPUMP\_DIR TO PUBLIC;

**B. Precreate Tablespaces (Optional)**

If the datafiles are not in the default location, precreate tablespaces on the target:

CREATE TABLESPACE USERS DATAFILE '/u01/oradata/PRODDB/USERS01.DBF' SIZE 500M REUSE;

CREATE TABLESPACE DATA\_TBS DATAFILE '/u01/oradata/PRODDB/DATA\_TBS01.DBF' SIZE 500M REUSE;

**5. Perform the Full Transportable Import**

Use **Data Pump Import (impdp)** to import the metadata and integrate the transported tablespaces.

**Command**:

impdp "sys/oracle as sysdba" directory=DATAPUMP\_DIR dumpfile=full\_transportable\_%U.dmp logfile=full\_transportable\_import.log full=y transport\_datafiles='/u01/oradata/PRODDB/USERS01.DBF','/u01/oradata/PRODDB/DATA\_TBS01.DBF'

**Expected Output (Log File)**:

Import: Release 19.0.0.0.0 - Production

Processing object type TRANSPORTABLE\_EXPORT/PLUGTS\_BLK

Processing object type TRANSPORTABLE\_EXPORT/TABLE

Processing object type TRANSPORTABLE\_EXPORT/INDEX

Successfully imported tablespaces: USERS, DATA\_TBS

**6. Post-Migration Steps**

**A. Verify Tablespaces**

Ensure the tablespaces are online and accessible:

SELECT tablespace\_name, status FROM dba\_tablespaces;

**B. Change Tablespaces to Read/Write Mode**

Set the transported tablespaces to read/write mode:

ALTER TABLESPACE USERS READ WRITE;

ALTER TABLESPACE DATA\_TBS READ WRITE;

**C. Verify Data**

Check if the data was imported correctly:

SELECT COUNT(\*) FROM hr.employees;

**D. Compile Invalid Objects**

Compile any invalid objects:

EXEC DBMS\_UTILITY.compile\_schema(schema => 'HR');

**7. Cleanup**

* On the source database, you can revert the tablespaces to read/write mode if needed:
* ALTER TABLESPACE USERS READ WRITE;
* ALTER TABLESPACE DATA\_TBS READ WRITE;
* Remove temporary dump files and logs if no longer required.

**Rollback Plan**

If the migration fails:

1. Investigate errors in the export/import logs.
2. Correct the issue (e.g., file paths, permissions).
3. Repeat the export/import process.

**Advantages of Full Transportable Export/Import**

* Faster than traditional export/import for large databases.
* Combines transportable tablespaces and Data Pump to handle both transportable and non-transportable data.

Let me know if you encounter any specific issues or need further clarification! 😊