Rapid Unload / Load Oracle data using WisdomForce FastReader

WisdomForce Technologies, Inc.
http://www.wisdomforce.com

Note*: The latest copy of this document is available at
http://www.wisdomforce.com/dweb/resources/docs/UnloadLoadUsingFastReader.pdf

Steps for rapid Oracle database migration

1) The schema(s) should be exported using the exp utility using the ROWS=N parameter. During this stage all the objects which are **non data**, such as PL/SQL, Sequences, Views, Table definitions, Object privileges etc. will be exported. There will be one such exp file per each schema exported. FastReader allows exporting of the schema definitions such as tables, constraints, sequences, triggers and generating scripts.

2) The data from the large tables (fact tables and the large dimensions) should be exported using FastReader.

3) The data from the small code tables and dimensions can be exported either using FastReader or native exp utility. Native exp should be used with the TABLES parameter specifying only the small tables to be extracted. There will be one exp file per each schema exported.

4) The destination database should be setup with all the same global roles, users, user privileges, system triggers (on_logon) and tablespaces (i.e. all the settings which are not exported using exp). If there is a need for a different tablespace/storage configuration then this should be taken into account in the latter stages.

5) The schema(s) should be imported using imp (schemas and small tables). *FastReader allows exporting of the schema definitions such as tables, constraints, sequences, triggers, etc. to be created on destination database*

6) Global object privileges and object synonyms should be created as they were on the original instance. These definitions are not exported during the schema export by exp.

7) The data from tables which were extracted using FastReader can be loaded into the target (destination) database. FastReader is utilizing the existing high-speed loaders for each supported database vendor (*for instance, when data to be loaded into Oracle, then sqlldr will be utilized*). FastReader creates automatically the control files for every supported target database vendor that reflect what tables/columns/data to be loaded and the scripts for actually executing the load.
Before the load all the constraints and triggers on the large tables which can prevent “direct load” mode or hurt load performance should be disabled.

Constraints on target database can be disabled / enabled also from FastReader GUI (Configurator).
If materialized views are used with "refresh on commit" option, these should be disabled and refreshed completely at a later phase.
For better control of the process the indexes maybe be dropped and recreated after the load. If the indexes remain then during the direct load, Oracle will build them using an efficient method similar in performance to creating an index on a full table.
It is possible to load several tables and/or partitions simultaneously, by running several SQL loaders at once. This can improve load performance significantly on a multi-processor machine with good I/O.

8) Indexes, if removed, should be recreated on the large loaded tables, this is a major, computationally intensive step, comparable to the unload/load phase, Care should be invested in improving the performance of this step by increasing the sort area or PGA memory sizes, performing several builds simultaneously under the available constraints (for instance you cannot effectively build two indexes on the same table at the same time), possible creating partitioned indexes in an unusable state and the rebuilding several partitions simultaneously, etc.

9) Constraints and triggers on the large tables should be enabled, constraints should be enabled with the INVALIDATE clause, for a performance improvement.

10) Materialized views should be recreated and/or rebuilt using "Complete Refresh". It is also possible that materialized views can be moved with their data intact and then their status changed to TRUSTED or created as if from pre-built tables but this step is more complex in terms of correct execution and setup

11) Indexes on the materialized views should be recreated (if not already present)

Advanced

FastReader's compress-on-the-fly / QZip feature can be utilized to avoid expensive unloads to disk and then the zip operations. Built-in FastReader's compress-on-the-fly / QZip combination is significantly faster then zip/unzip.

WisdomForce provides customers with unload/transfer/load cascade template-scripts, which can be used for automatic data transfer. By time the data unload is completed, the load is finished as well.

For more information about FastReader capabilities please use FastReader's User Guide.

