RMAN によるバックアップ取得操作

- 単純なフルバックアップ
- 差分（増分）バックアップ
- 累積増分バックアップ
- 増分更新バックアップ

・RMAN の差分バックアップの概念

・単純なフルバックアップ（オンラインバックアップ）

RMAN の起動

```
$ rman target USER/PASSWORD
```

並行処理（チャンネル）数の設定（設定数２）

```
rman> CONFIGURE DEVICE TYPE DISK PARALLELISM 2;
```

バックアップファイルの格納場所とバックアップのファイル名指定

（チャンネル１に対して）

```
rman> CONFIGURE CHANNEL 1 DEVICE TYPE DISK
        FORMAT '/backup/FULL_%d_%T_%U_ch1.bak';
```

（チャンネル２に対して）

```
rman> CONFIGURE CHANNEL 2 DEVICE TYPE DISK
        FORMAT '/backup/FULL_%d_%T_%U_ch2.bak';
```

%d：データベース名　%T：タイムスタンプ　%U：シーケンス番号

並行処理（チャンネル）数１の場合の格納場所とバックアップのファイル名指定

rman> ~~CONFIGURE CHANNEL DEVICE TYPE DISK~~
~~FORMAT '/backup/FULL_%d_%T_%U.bak';~~

%d：データベース名　%T：タイムスタンプ　%U：シーケンス番号

バックアップの採取

［イメージコピー（OS 上に存在しているファイルと完全に同一形式で）］として

rman> BACKUP　AS　COPY　DATABASE；

※ OS の COPY コマンドをすれば、リストア・ファイルとして使用出来る
ただし、ファイル名はバックアップ時に指定されたファイル名となっているので、コピー時にデータベースでの使用時の名前に変更（元の名前）する必要がある

バックアップされたファイルの名前の例）

BACKUP_ORCL_20180628_DATA_D-ORCL_I-142908724_TS-SYSTEM_FMO-1_45T6M4RU.BAL

［バックアップセット（RMAN でしかオープン出来ない形式）］として

rman> BACKUP　AS　BACKUPSET　DATABASE
PLUS　ARCHIVELOG　　DELETE　ALL　INPUT；

↑ アーカイブ Redo ログ・ファイルもバックアップに含める

↑ バックアップ終了後、入力で使用したアーカイブ Redo ログ・ファイルは、削除する

※　全体のバックアップを取得した時点で、全てのアーカイブ Redo ログ・ファイルは不要な状態となっている

しかし、障害発生時にバックアップしたデータファイルをリストアすると、バックアップの採取タイミングの違いによりデータファイルの SCN 値が異なっている

この違いを一致させるために RECOVER 処理を行うが、この時にはバックアップの実行を開始した後の Redo ログ・ファイルが必要となる

これを取得するために、PLUS　ARCHIVELOG オプションを指定している

・差分（増分）バックアップ

直前のバックアップ（フル or 差分）からデータ変更があった部分をバックアップします

リストア時には、フルバックアップと、それ以降の全ての差分増分バックアップが必要になります

【週末のフルバックアップ】

rman> backup incremental level 0 database；

【通常日の差分増分バックアップ】

rman> backup incremental level 1 database；

・累積増分バックアップ

フルバックアップ以降からのデータ変更があった部分をバックアップします

（累積増分には、前日の差分増分も含まれてきます）

リストア時には、フルバックアップと、直前の累積増分バックアップだけで必要になります

【週末のフルバックアップ】

rman> backup incremental level 0 database；

【通常日の累積増分バックアップ】

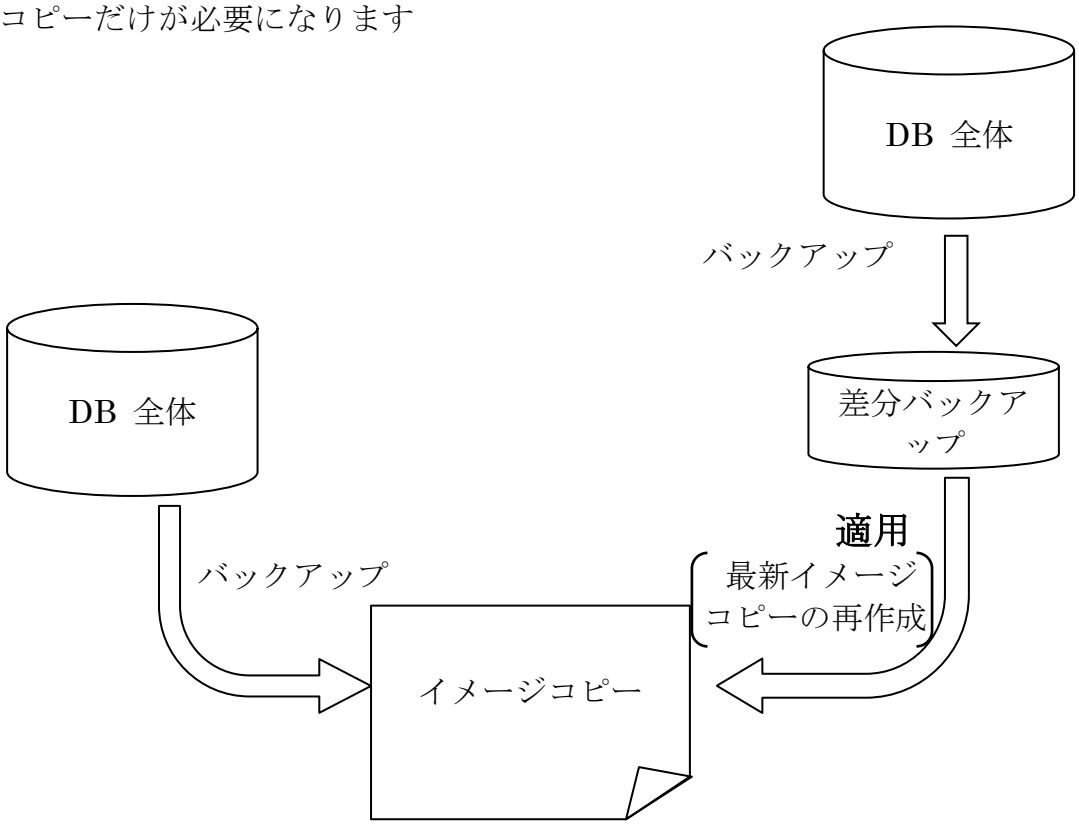
rman> backup incremental level 1 cumulative database；

・増分更新バックアップ

直前のバックアップ（フル or 差分）からデータ変更があった部分をバックアップします

最初に採取したフルバックアップに、今回採取した差分バックアップを適用して、フルバックアップのイメージコピーにこの差分データも反映した新たなイメージコピーを、再作成しておきます

リストア時に必要なバックアップは、前日の差分データを反映して再作成したイメージコピーだけが必要になります



※ バックアップ JOB には、週末のフルとか通常日の差分とかの分かれた JOB は、ない

イメージコピーのバックアップが無ければ、イメージコピー形式で作成して、あれば増分バックアップを取得する

その後で、最新のイメージコピーを再作成のために、差分バックアップを適用する

【フルバックアップの採取】操作

~~・バックアップの採取~~

~~データベース全体のバックアップのイメージコピーを採取~~

~~rman> backup incremental level 0 for recover of copy with tag 'incremental_update' database；~~

【差分・バックアップの採取】操作

※ ただし、全体イメージのコピー元バックアップが無ければ、全体のイメージコピー形式を作成する、有れば増分だけのバックアップ）

・バックアップの採取

rman> backup incremental level 1 for recover of copy with tag 'incremental_update' database；

・イメージコピーに差分バックアップを適用

（前回作成したイメージコピー形式のファイルに対して、採取バックアップのデータを反映させて、最新の状態のイメージコピー形式のファイルを再作成する）

rman> recover copy of database with tag 'incremental_update'；

【参考情報】

増分更新バックアップで作成されたデータベースを示すデータファイルを削除した場合、再度、新規で増分更新バックアップ処理を行うためには、以下の処理が必要となる

・crosscheck backupset；

・delete expired backupset；

・crosscheck copy；

・delete expired copy；

・ブロック変更追跡の有効化

バックアップ採取後に、次回バックアップのための**ブロック変更履歴用の物理ブロックインデックス**を作成して、この変更インデックスから変更部分の差分バックアップを作成する
通常では、物理ファイルの全ブロックを読込み、変更があったかの確認を行い、そこから差分バックアップを作成している

・ブロック変更追跡の有効化

sql> alter database enable block change tracking using file '変更履歴用の物理ブロックインデックスのファイルパス名';

・ブロック変更追跡の無効化

sql> alter database disable block change tracking;

・ブロック変更追跡の有効／無効状態の確認

sql> select status from v$block_change_tracking ;

ブロック変更追跡を利用した場合のバックアップ操作手順

フルバックアップ時

1）ブロック変更追跡の無効化
2）変更履歴用の物理ブロックインデックスのファイルの削除
3）ブロック変更追跡の有効化
4）Level 0 のフルバックアップの採取

通常時は、

5）Level 1 の差分バックアップの採取