

- ・結合させる表間の列にインデックスを作成して、連携効率を上げるのは、選択率が低い状況に限る
(列の値の偏りが、分散している)

検索列（キー）が同一な値のレコードが多数存在する場合には、インデックスでの読込は、効率が落ちる場合がある。

ユーザー・テーブル

名 前	部署コード	情 報
愛川こずえ	1	
いとくとら	1	
	・ ・ ・ 100件	
ミンカ・リー	1	
足太ペンタ	2	

この状態で、

```
select 名前, 情報 from ユーザー・テーブル WHERE 部署コード=1
```

を行って、実行計画が `index unique scan` を行った場合には、`table access full` より効率が落ちる

< 理 由 >

インデックス読込では、`index unique scan` の後に `table access by index rowid` が行われる。

`table access full` では、ブロックをまとめてシーケンシャルに読込むようにし効率を高めている（まとめるブロック数は、`db_multiblock_read_count` の値 デフォルト：8）

`table access by index rowid` では、1ブロックごとのランダム（検索）読込みとなる

```
-----
| OPERATION                | PHV/Object Name | Rows | Bytes | COST |
-----
| SELECT STATEMENT         | ---- 1004337041 ---- |      |      | 87 |
| TABLE ACCESS BY        | DEPARTMENT      | 44 | 22 | 1 |
| INDEX ROWID             |                  |      |      |   |
| INDEX UNIQUE SCAN       | DEPT_ID_PK      | 1 |      | 0 |
-----
```

索引を使った検索の有効性

検索が、全体の30%以内であれば、索引を作成し索引検索が有効となる

列値がどの値の時に、インデックスが有効かを判断するためには、インデックスにヒストグラム・オブジェクト統計情報を作成する

インデックス使用が有効な事例)

トランザクション・テーブル

売り上げ No	売り上げ情報	処理済フラグ
0001		Yes
0002		Yes
		Yes . . . 多数
9981		No
9982		No

毎日のトランザクション・データに対し、業務において締め処理をバッチで行う教務形態を想定する

この場合、締め処理は、前日の処理終了後から増えたレコードを対象にする

トランザクション・レコードに処理済フラグを作成し、Yes or No を管理設定する

このようにすれば、日々の業務でレコードが増えていった場合でも、締め処理対象とするレコードの件数は、全体の一部となるだけで処理は軽い

行いたいことは、この締め処理において、対象データの抜き取り (No データのみの処理) を行うときにトランザクション・テーブル全件の**検索を行わない**ようにすることである

(ただし、Yes データを対象にする場合には、テーブルの全件検索を行わせる)

有効な方法)

トランザクション・テーブルの処理済フラグに、別途インデックスを作成する

トランザクション・テーブルの処理済フラグ列に対して、値分布を示すヒストグラムを作成する

こうすることにより、処理済フラグが No のデータ処理の時は、インデックスを使い

Yes のデータ処理の時は、テーブル全件検索を使う

処理側のソース・プログラムの注意点として、プログラムではバインド変数を使わないこと
バインド変数を使った場合は、SQL 文の実行計画作成のパス処理時点では、フラグ値が確定していないので、パス処理においてヒストグラムを活用した実行計画作成ができない

参考情報)

処理済フラグが No のときの実行計画

OPERATION	PHV/Object Name	Rows	Bytes	COST
SELECT STATEMENT	---- 2678922101 ----			2
TABLE ACCESS BY INDEX ROWID	TRANSACTION-T	2	22	1
INDEX RANGE SCAN	TRAN_FLG_IDX	1		1

処理済フラグが Yes のとき、およびバインド変数を使った場合の実行計画

OPERATION	PHV/Object Name	Rows	Bytes	COST
SELECT STATEMENT	---- 2678929999 ----			28002
TABLE ACCESS FULL	TRANSACTION-T	34587	169879	28002