



如何在ASE中重組(Defragment)分散的資料

作者：倍力資訊第一事業部 系統顧問
張安翔 先生

資料表上進行頻繁的資料刪除、插入等操作，久而久之會造成資料庫碎片的大量生成，資料庫性能會下降，此時就需要定時進行reorg操作。何時該執行reorg命令，以下情況發生時，執行reorg命令非常有用：

- 大量轉移的行導致讀取操作過程中有額外的I/O。
- 遇到的頁有非連續的可用空間需要回收，而造成插入和讀取較慢。
- 大量I/O操作較慢。

reorg

利用此一功能來重寫表格中所有的列到新的分頁上，以達到分散資料及未使用空間的重組。有了reorg和reorg reclaim_space指令現在可以配合經常性的新增 和刪除動作，以有效控制這些表格的成長。

語法：

(1) **reorg reclaim_space**，回收因刪除操作和行（row）縮短更新操作，所產生分頁上剩餘的未用空間。此命令將會連續重寫剩餘的行，而將所有未使用的空間留在頁尾，如果所有行都已刪除且沒有剩餘的行，則它將釋放該頁，語法如下：

```
reorg reclaim_space tablename [indexname]  
[with {resume, time = no_of_minutes}]
```

(2) **reorg forwarded_rows**，撤消行的轉移操作，將轉移行移回到它自己的主頁（如果有足夠的空間），或者刪除行並將它重新插入一個新的主頁，語法如下：

```
reorg forwarded_rows tablename  
[with {resume,time = no_of_minutes}]
```

(3) **reorg compact**，回收空間並撤消行的轉移，此命令將逐頁地掃過表格並撤消所有行的轉移，語法如下：

```
reorg compact tablename
```

[with {resume, time = *no_of_minutes*}]

(4) `reorg rebuild`，除了和`reorg compact`一樣執行空間回收並撤消行的轉移外，`reorg build`另外還會執行：

1. 重寫所有的行以便與表格的叢集索引（Cluster Index）一致(如果有)。
2. 對資料頁寫入行，以便與使用`sp_chgattribute`對空間管理設定所作的改變保持一致。
3. 刪除並重新建立表格的所有索引。

語法如下：

```
reorg rebuild tablename [indexname]
```

範例 1 `reclaim_space`：

```
reorg reclaim_space titles
```

範例 2 `reclaim_space`（含Index `titleind`）：

```
reorg reclaim_space titles titleind
```

範例 3 `reorg compact`：

對表格`titles`執行`reorg compact`為時120分鐘，但如果120分鐘內沒完成則視同執行失敗，下回須重新執行。

```
reorg compact titles with time = 120
```

範例 4 `reorg compact`（含`resume`）：

對表格`titles`執行`reorg compact`為時30分鐘，如果30分鐘內未完成，則下回可繼續處理未完成的作業。

```
reorg compact titles with resume, time = 30
```

範例 5 `reorg rebuild`：

執行`reorg rebuild`的前提條件：

- 將資料庫選項`select into/bulkcopy/pllsort`設定為`true`，並在資料庫中執行`checkpoing`。
- 確認具有與表格及其索引大小相等的額外磁碟空間。

要將`select into/bulkcopy/pllsort`設定為`true`，並對資料庫進行檢查點操作，可使用以下`isql`命令：

```
1> use master
2> go
1> sp_dboption pubs2, "select into/bulkcopy/pllsort", true
2> go
1> use pubs2
2> go
1> checkpoint
2> go
```

reorg rebuild titles