Adager's Repacking of IMAGE datasets

Ken Paul
Adager Corporation
Sun Valley, Idaho 83353-3000 • *USA*
http://www.adager.com

Adager's *Repack* function can improve your database performance very dramatically and it deserves special attention. In this article I'll discuss the reasons and options for repacking, and the crucial repack elements: how to choose a path to repack, how often to do it, and how long it takes.

Repacking optimizes the chained access of a detail dataset with at least one path. It also may be used to optimize the serial access of any detail dataset. Adager's Repack Dataset command, also known as DetPack when applied to detail datasets, is available to all Adager users in both compatibility and native mode.

Why would I want to repack a detail dataset?

As entries are added and deleted to detail datasets, the time it takes to retrieve entries with a given key value can take longer and longer. By placing the members of each chain in contiguous locations (so that their physical order coincides with their logical order on the chain), Adager's DetPack minimizes the number of blocks occupied by each chain of the repacking path, thereby reducing disc I/O during chained access along this path. In other words, since the system brings into memory more than one record at a time, the odds of a subsequent requested record already being in memory can be greatly increased through repacking. Serial scans of the dataset will also benefit. For example, if prior to repacking, only 50% of the entries below the HighWater mark are occupied, after repacking a serial read of the dataset with a large number of entries will be about twice as fast.

Which repacking option should I use?

Adager provides you with FOUR repacking options: SERIAL+, SORTED+, CHAINED+ and SuperCHAINED+. All of them squeeze out empty (deleted) entries and lower the HighWater mark to equal the number of entries in the dataset. More importantly, all of the options preserve the chronological order of the entries on ALL chains of ALL paths. Additionally, Adager lets you specify a new capacity for the detail dataset.

SERIAL+

The SERIAL+ option squeezes out empty entries and lowers the HighWater mark without further reorganization. A serial read of the dataset before and after the repack will encounter the records in the same order. This option is useful if you want to lower the capacity below the current HighWater mark and do not want to reorganize the chains. It is also useful when you need to maintain the serial order of the entries within the dataset. If the dataset is a stand-alone detail your application will benefit from the improved locality of the data.

SORTED+

The SORTED+ option sorts the entire dataset according to the values of a virtual sort key which you specify using up to four fields of the dataset. When this option is complete, a serial pass of the dataset will find the records in ascending order based on the fields which you provided as the sort key. The order of records within a given chain is not changed because Adager "PRESERVES the chronological order of the entries on ALL the chains of ALL the paths." This option benefits applications which do serial scans with "control breaks" when the value of the virtual sort key changes. You might also use this option if you do many sorts with SuperTool using the same keys. This option may not benefit chained access to the dataset along any path, unless you have specified some path's search field as the primary sort key in your virtual sort key.

CHAINED+

The CHAINED+ option reorganizes the detail's chains to coincide with the serial order of the master ChainHeads. This is the default option for detail datasets with at least one path (i.e. not stand-alone details). This option resembles the results of a CHAINED DBUNLOAD followed by a DBLOAD. However, there are two major differences. First, Adager preserves chain chronology on all paths while DBUNLOAD/DBLOAD only preserves chain chronology on the primary path. Second, Adager allows you to select which path to repack while DBUNLOAD/DBLOAD can only use the primary path.

SuperCHAINED+

The SuperCHAINED+ option guarantees that each detail chain, except possibly the last one, spans the minimum number of IMAGE blocks. This option mostly helps on-line performance on Classic HP 3000s, where IMAGE's I/O is done at the block level. On the MPE/iX HP 3000s, where IMAGE's I/O is done at the page level, this option is less likely to have a positive impact compared with the extra time it may take to place the records in the optimum order.

Difference between CHAINED+ and SuperCHAINED+

To see the difference between a CHAINED+ and a SuperCHAINED+ repack let's look at the following example.

A path whose master dataset's entries are organized sequentially in order "ABCD." The "messy" detail set (in desperate need of repacking) has 21 entries. The path that we have selected for repacking has 4 chains (6 entries have search field value "A," 8 entries have search field value "B," 4 entries have search field value "C" and 3 entries have search field value "D"). The detail's blocking factor is 10.

Before repacking, the first 5 blocks of the detail dataset look like this:

Block 1: CAD.ABBC.A
Block 2: .BCBAB.D.B
Block 3: A . . . . DB . . .
Block 4: . . . C . A . B . .
Block 5: . . . + ...

("." means a free entry; "+" marks the first entry above the HighWater mark)

The CHAINED+ option produces this result:

Block 1: AAAAAABBBB
Block 2: BBBBCCCCDD
Block 3: D+ . . . . . .
Block 4: . . . . . .
Block 5: . . . . . .

The SuperCHAINED+ option produces this result:

Block 1: AAAAAACCCC
Block 2: BBBBBBBBDD
Block 3: D+ . . . . . .
Block 4: . . . . . .
Block 5: . . . . . .

The SuperCHAINED+ option minimizes the number of times that each chain (along the repacking path) crosses a block boundary. To accomplish this goal, this option does not necessarily keep the chains in the same order as the master entries (for instance, the detail entries for chain "C" come before the detail entries for chain "B"), although entries within a chain are kept in chronological order.

Which detail path do I want to repack?

Choosing the correct detail path to repack is THE most important factor, from a performance viewpoint. You do NOT want to repack a detail path where there is only one detail entry for every master entry, even if this is the primary path. Repacking along this type of path will have minimal (if any) impact on database performance: these paths can be thought of as already packed so a different path should be chosen. Choosing the primary path may not be a good idea if the primary path just "happened," as a result of the database designer not marking explicitly a primary path in the original schema, and IMAGE, by default, selecting the first unsorted path as the primary path. Since you can only repack along one detail path, repacking an undeserving primary path would not benefit anybody.

In general, you want to select a detail path which is frequently accessed and which has chain lengths whose optimization will result in a decrease of the overall I/O activity for the dataset. In cases where there are several candidates, the selection process may involve several attempts by actually repacking the detail dataset along different paths and measuring the performance improvements for each one. Since repacking a detail dataset is usually performed periodically, this is a worthwhile investment.

A good working knowledge of your application software can help you decide which detail path to repack. If you are not familiar with the software (for instance, if you purchased it from a third party supplier), check with your online users to find out which inquiries they are performing most often and relate their observations to a path in the given detail dataset. You may also want to contact your software supplier to obtain their suggestions.

Why can't I repack each path of my detail dataset, one after the other?

It doesn't make sense to repack *all* paths in a detail dataset. It only makes sense to repack *one* path per detail dataset because, as you repack the records of a detail dataset along one path, you are—sadly—"unpacking" the records along the other path(s).

How often should I repack a given detail dataset?

The frequency with which you should repack a dataset depends on the turnover rate of the entries in the dataset. As entries are added and/or deleted from a detail dataset, the benefits of the last repack are diminished. The first-ever repack for a detail dataset always takes the longest time because the entries are in a very "messy" order. If subsequent repacks are done on a consistent basis the time to do these repacks is greatly reduced. Users have reported a five-fold time reduction after the first repack

even with an increase in the number of entries processed. Some users even repack all of their detail datasets every night because this way the daily overhead is minimal.

One of the best times to repack a detail dataset is after an archival process which has removed many entries from the dataset. Once these entries have been deleted it is a good idea to take the remaining entries and pack them before adding more entries. You may also want to look at your month-end or year-end processing to see if you should add repacking of your detail datasets to this process.

What determines how long it takes to repack a detail dataset?

There are many factors which affect the time it takes to repack a detail dataset. The main ones are: the original "messiness" of the dataset, the Adager repack option chosen, the number of entries in the datasets involved, number of paths, the dataset's entry and block lengths, the type of HP 3000 system, and other processes concurrently running in the system.

Can I repack and resize my detail datasets at the same time with Adager?

Of course. Adager allows you to change the capacity of a detail dataset while reorganizing it.This feature is very useful when you wish to reduce the capacity of a detail dataset to a minimum even when the current value of the HighWater Mark is greater than the number of entries in the dataset. Adager also allows you to specify the device number, device class or volume class where you want the resulting dataset file to reside.

Do I need to do anything special to invoke Adager's Native Mode DetPack function on my MPE/iX machine?

No. Adager can tell if it is running on a Spectrum machine and it will invoke its Native Mode Detpack automatically. All of the features of the Classic DetPack are contained in the Native Mode version including changing the capacity of the detail dataset as your repack it.