



dmunput

The under-appreciated DMF repair tool

Peter Edwards | Systems Support Manager

4 December 2012

CSIRO IM&T ADVANCED SCIENTIFIC COMPUTING
www.csiro.au



Oh, just forget it!

`dmunput` is intended to reverse the effect of a `dmpu`.

It removes files from DMF management by first recalling them to dual-state if necessary, and then converting them to regular (non-migrated) state, thereby soft-deleting the copies already maintained by DMF.

`dmunput` is available only to administrators.

Unfortunately, there is not yet any guidance on how it can be used to simplify the repair of several different types of problems.

Scenarios discussed

- Inodes with duplicate BFIDs
- Duplicate chunks in CAT DB
- Bad DCM attributes
- Bad tape copy
- Unexpected UNM state

Inodes with duplicate BFIDs

The problem

Diagnosed by dmaudit and (in theory) fixable interactively with dmaudit

Suppose you reload a directory `/fs/userdir/dir` from backups, and you put it somewhere temporary like `/fs/tmp/dir`.

If the originals are still on disk, two copies now exist both referring to the same DMF database records. The assumed 1:1 relationship is now broken without DMF knowing about it.

dmaudit will find and report it, and provides a mechanism to fix it interactively. This works well, but is confusing and slow. For each such file, you have to navigate up and down inside dmaudit's menu hierarchy making semi-comprehensible choices, and it's all manual. One file is fine, but 100 is unworkable.

Inodes with duplicate BFIDs

An alternative

1. reload the directory to `/fs/tmp/dir`
2. remove all DMF knowledge of these files with
`dmunput /fs/tmp/dir/*`
3. hand this directory over to the user by mv'ing it to `/fs/userdir/dir-reloaded` or by fiddling with permissions, or similar

DMF now believes (incorrectly) that the files in `/fs/userdir/dir` have been deleted, because `dmunput` has set the soft-delete flags in the shared database entries.

This will be detected by the next `dmaudit` run and is easily fixed interactively with `dmaudit`.

Duplicate chunks in CAT DB

The problem

Diagnosed by `dmaudit/dmatvfy` but fixed interactively with `dmcatadm`

This is usually the result of a tape merge which aborted somehow.

There is no easy tool to help you resolve this - you have to use `dmcatadm` to delete one of the duplicate chunks.

If the chunks cover the same byte range, which is the most common case, then it doesn't matter which you delete. If one is completely covered by the byte range of the other, you delete the smaller one.

This can be tricky, as a small mistake could result in both of the duplicates being deleted, which could result in loss of data.

Duplicate chunks in CAT DB

An alternative

An alternative is to determine the pathnames from the BFIDs with the help of `dmlookup` and then use `dmunput` to recall the files to disk and turn them into REG files, which will be re-migrated some time later on.

```
cherax# echo 4fcef48900000000081cf19 | dmlookup | dmunput
```

All the chunks previously containing the file's data will be soft-deleted, resolving the conflict.

`dmlookup` is a local script to convert BFIDs and fhandles to pathnames; see http://hpsc.csiro.au/users/dmfug/Meeting_Dec2012/Presentations/CSIRO_ASC_DMF_tools.pdf

Bad DCM attributes

The problem

Diagnosed by `dmdskvfy` and fixable interactively with `dmdskvfy`.

Sample output:

The following 2 files in MSP cache have DCM attributes which need correcting:

```
/dmf/cache/rus150/17/4fcef489000000000036db3a  
/dmf/cache/rus150/17/4fcef4890000000000373ff4
```

To fix the above incorrect attributes, stop DMF and rerun this command specifying the `'-a'` option.

Note the requirement for DMF to be down for a while, up to 45 minutes for CSIRO ASC.

Bad DCM attributes

An alternative

Instead, we use `dmlookup` to get the path of the user's file, and then run `dmunput` to recall the file and change it to state REG. The file will be re-migrated some time later.

```
cherax# echo 4fcef489000000000036db3a | dmlookup | dmunput
```

(If `dmlookup` can't find the file's path, it's probably been soft-deleted, and the warning will go away when the file is hard-deleted.)

Bad tape copy

The problem

Errors seen in logs, and emails concerning locked tapes received

If due to a media fault (rare in our experience) or minor corruption, this can be fixed with `dmmove` or `dmvoladm`.

The tape's HSPARSE flag can be set with `dmvoladm`, which will attempt to move all chunks on that tape to another. If any remain, use `dmemptytape` to finish the job.

(If your intent is remove the tape from the system, this is your only realistic option using standard tools.)

Or the files can be moved out of the VG to a different one with `dmmove`, and then moved back again with another `dmmove`.

Bad tape copy

An alternative

Given the pathnames or fhandles, you can convert the files back to regular state with `dmunput`, leaving them available for migration to different tapes later on.

```
cherax# dmunput -h \  
01000000000018857751791553287200000001003ec1267c1e6400000000
```

If you only have the BFIDs, `dmlookup` can convert them, as shown previously.

Unexpected UNM state

The problem

Diagnosed by `dmaudit` and fixable with `dmput`.

This is typically due to files which were being recalled when for some reason DMF went down. When it comes back again, `dmaudit` can see that the files are in UNM state, but the new instance of the `dmfdaemon` has no knowledge of it.

Another possibility is when a recall fails due to the user reaching their hard space quota.

Unexpected UNM state

No proper alternative

Best fixed with `dmput`, with or without `"-r"`, which forces it to a known state (OFL).

`dmunput` will also work, but is no easier and the file will be unnecessarily re-migrated later on, which could be an issue with very large files.

But occasionally `dmput` makes no difference, in which case `dmunput` should be used, even on large files.

Conclusion

Using dmunput in these ways may be a crude solution, but it can save a lot of administrator time at the cost of some extra file migrations.

Thank you

Peter Edwards
Systems Support Manager
Advanced Scientific Computing
e peter.edwards@csiro.au
w <http://www.hpsc.csiro.au/>

CSIRO IM&T ADVANCED SCIENTIFIC COMPUTING
www.csiro.au

