



CCAPRINT

A Newsletter Excerpt for Model 204 Users

February 2008

USE OF AND ACCESS TO PRODUCTS AND FEATURES ARE IN ACCORDANCE WITH THE TERMS AND CONDITIONS OF THE USER'S SOFTWARE LICENSE. THE PRESENTATION OF MATERIAL HEREIN DOES NOT, IN ANY MANNER, MODIFY SUCH TERMS AND CONDITIONS.

The Multiple Uses of the RENAME FILE Command

By James Damon

Sometimes you need to simply rename a file. Sometimes it is useful to make a complete copy of a production Model 204 file for use in a different environment, such as testing, QA, or development. Sometimes you may also want to use the copied file in the same Online--under a different file name, of course. The V6R1.0 RENAME FILE command has this covered. This article uses code examples to illustrate how to create duplicates and/or rename files most efficiently for each situation.

Renaming Using DUMP and RESTORE Processing

In simple cases you can dump the source file to a sequential dataset, then restore it to a different ddname and dataset--the target file--using the RESTORE 192 command. The resulting target file is an exact copy of the source file under a different file name.

In the Figure 1 example the source file, MUSIC, may be a single or multi-dataset file. The target file, SONGS, may also be a single or multi-dataset file. This example assumes a single-dataset file for SONGS.

Figure 1. Rename files using DUMP and RESTORE commands

```
ALLOCATE DMPMUSIC WITH OLD DSN=CCA.MUSIC.DUMP.M204
ALLOCATE MUSIC CCA.MUSIC.M204
ALLOCATE SONGS CCA.SONGS.M204
OPEN MUSIC
DUMP FILE MUSIC TO DMPMUSIC
CREATE SONGS
END CREATE
OPEN SONGS
RESTORE 192 FROM DMPMUSIC
```

The sequence of commands in Figure 1 makes a copy, via DUMP and RESTORE, of the file MUSIC. As a result of the RESTORE 192 command that new copy is named SONGS. There is one ddname=SONGS and one dataset associated with that ddname.

Making Files Copies with Sequential Copy Utilities

You can also copy the original dataset(s) to new datasets using IEBGENER, DFDSS, IDCAMS, or similar utility. The new dataset(s) can then be allocated to a different Online where the file would have the same name, the same DDNAMEs, FILEDEFs or DLBLs but different dataset(s). However, the file name or ddnames cannot be changed using this method.

Note that these utilities cannot be used if you have the obsolete file skewing option enabled and are changing device types--moving from a 3380 to a 3390, for instance. These utilities may also not work when copying DSORG=DA datasets from 3380s to 3390s. Model 204 files are no longer set to DSORG=DA, unless you use the DIRECT keyword (which is unnecessary) in the DEFINE or ALLOCATE command, but there may still be some DA files in existence

Renaming files consisting of multiple datasets and ddnames

Assume now that MUSIC is a multi-dataset file consisting of the ddnames in Figure 2. Dataset names can be anything you require:

Figure 2. DDNAMEs and dataset names associated with MUSIC

```
//MUSIC DD DSN=CCA.MUSIC.M204  
//MUSIC1 DD DSN=CCA.MUSIC1.M204  
//MUSIC2 DD DSN=CCA.MUSIC2.M204  
//MUSIC3 DD DSN=CCA.MUSIC3.M204
```

Also assume that, instead of using the DUMP and RESTORE commands, you have copied the four MUSIC datasets to the four SONGS datasets as shown in Figure 3.

Figure 3. Dataset names associated with SONGS

```
DSN=CCA.SONGS.M204  
DSN=CCA.SONGS1.M204  
DSN=CCA.SONGS2.M204  
DSN=CCA.SONGS3.M204
```

To use these datasets in a file named SONGS requires that this collection of datasets first be processed by the RENAME FILE command.

Using the RENAME FILE Command

In V6R1.0 single or multi-dataset files can be renamed in place. For these files you can give the target file a different name and different DDNAMEs, FILEDEFs or DLBLs via the RENAME FILE command.

All of the datasets associated with the MUSIC file have been copied to the corresponding SONGS datasets shown in Figure 4 and allocated to an Online with the MUSIC ddnames. The MUSIC ddnames are required until the file is renamed because, internally, this collection of datasets is still the MUSIC file.

Figure 4. SONGS ddnames and datasets

```
//MUSIC DD DSN=CCA.SONGS.M204
//MUSIC1 DD DSN=CCA.SONGS1.M204
//MUSIC2 DD DSN=CCA.SONGS2.M204
//MUSIC3 DD DSN=CCA.SONGS3.M204
```

The following series of commands will rename this copy of the MUSIC file to the SONGS file. The File Parameter List (FPL) in SONGS will also be updated to reflect the new ddnames.

Figure 5. Renaming the MUSIC file

```
OPEN FILE MUSIC
STOP FILE MUSIC
STOP FILE SONGS
RENAME FILE MUSIC SONGS (MUSIC1 SONGS1) (MUSIC2 SONGS2) (MUSIC3 SONGS3)
*** M204.0608: FILE CLOSED: SONGS
*** M204.2576: DDNAME RENAMED FROM MUSIC1 TO SONGS1
*** M204.2576: DDNAME RENAMED FROM MUSIC2 TO SONGS2
*** M204.2576: DDNAME RENAMED FROM MUSIC3 TO SONGS3
*** M204.2576: FILE RENAMED FROM MUSIC TO SONGS
```

At this point, before you can open the SONGS file, you must issue the ALLOCATE commands shown in Figure 6.

Figure 6. Allocating ddnames and datasets for the SONGS file

```
ALLOCATE SONGS CCA.SONGS.M204
ALLOCATE SONGS1 CCA.SONGS1.M204
ALLOCATE SONGS2 CCA.SONGS2.M204
ALLOCATE SONGS3 CCA.SONGS3.M204
START FILE SONGS
OPEN FILE SONGS
```

You could also make the MUSIC file available in the same Online with the commands in Figure 7. This requires first disassociating the MUSIC ddnames from the SONGS datasets using FREE commands, then associating them with the MUSIC datasets.

Figure 7. Allocating ddnames and datasets for the MUSIC file

```
FREE MUSIC
FREE MUSIC1
FREE MUSIC2
FREE MUSIC3
ALLOCATE MUSIC CCA.MUSIC.M204
ALLOCATE MUSIC1 CCA.MUSIC1.M204
ALLOCATE MUSIC2 CCA.MUSIC2.M204
ALLOCATE MUSIC3 CCA.MUSIC3.M204
START FILE MUSIC
OPEN FILE MUSIC
```

In Summary

If there is a requirement to simply rename a file in place, the RENAME FILE command is really the only option. Also, if you need to rename the file name and ddnames of a multi-dataset file, the RENAME FILE command is the most straightforward method.

There is a performance advantage in using RENAME FILE compared to DUMP and RESTORE processing. RENAME FILE processing requires one half the number of I/Os DUMP and RESTORE processing requires. It also reads and writes a track at a time without using the disk buffer pool. This avoids adding an additional load to the buffer pool, which happens with DUMP and RESTORE processing.

© 2008 Computer Corporation of America
200 West Street, 3rd Floor West, Waltham, MA 02451