Technique for Cloning a Small-to-Medium Data Base

Paul Kimpel

2019 UNITE Conference

Session MCP 4060

Wednesday, 2 October 2019, 4:00 p.m.

The Need…

- Create a test copy of a production data base
- Preserve existing TEST DMSII tailored software
- Preserve ability of previously-compiled programs to access the TEST data base
- Avoid "death by AX commands" to override data base timestamps
- Use standard DMSII tooling if at all possible

Considerations

- Both data bases:
  - Must exist, must have current audit trail files
  - Must have *completely identical schemas*
  - Must have *completely identical physical attributes*
  - Must be at the same DMSII release level
  - Must be *completely shut down* during the clone
- Some tables may need to be customized for the TEST environment – users, permissions, etc.
  - Copy/restore these tables across the clone
  - May need extract/restore programs to preserve more complex test environment data

Basic Technique, part 1

1. Note current TEST audit number
2. Completely shut down both data bases
3. Using Library/Maintenance
   - Back up TEST data base CONTROL file
   - Back up any TEST-specific tables or data
   - Copy all PROD data base files to TEST locations
4. On the new TEST copy
   - Remove all LOCKEDFILE attributes from the files
   - Fix up the GLOBAL data set name (if any)
5. Restore the TEST CONTROL file
6. Restore any saved TEST-specific tables/data

Basic Technique, part 2

7. *The Final Critical Step:*
   - Do RECOVER UPDATE on restored CONTROL file
   - Specify the current TEST audit number
   - **`RUN *SYSTEM/DMCONTROL("DB=MYDBDESC RECOVER UPDATE AUDITNUM=1234")`**
   - Will fix up the CONTROL file and correct all data base timestamps
8. If all of that worked okay, the TEST data base should now be usable

Paradigm 2019 MCP 4060 5

Notes

- Makes a complete copy of the PROD DB – not suitable for extracting a subset of test data
- Could also be used to restore a gold-standard copy of a test data base to a test environment
- Clone time is proportional to data base size
  - 2.5 GB data base, Libra 460, single disk family
  - Clone time = 20 minutes
- Rollback or recovery prior to point of clone not possible for TEST (will corrupt the data base)
- See sample WFL job:
  - http://www.digm.com/UNITE/2019/

Paradigm 2019 MCP 4060 6

CLONEMPDB.wfl

```
00100100  BEGIN JOB MPTEST/CLONEMPDB(STRING AUDITNBR);
00100200  NOJOBSUMMARYIO;
00100300  JOBSUMMARY = SUPPRESSED;
00110000
00115000  STRING
00115100    REPLY;
00118000  TASK
00118100    T;
00900000
00900100  DISPLAY "This job will clone (PROD)MPDB to (TEST)MPTEST";
00900200  DO BEGIN
00900300    DISPLAY "This job will destroy all existing data in (TEST)MPTEST";
00900400    REPLY:= ACCEPT("Make sure both DBs are inactive. Proceed? [Y/N]");
00900500    END
00900600  UNTIL REPLY="Y" OR REPLY="N";
00900700
00900800  IF REPLY NEQ "Y" THEN
00900900    ABORT "Clone of MPDB to MPTEST aborted";
00901000  ELSE
00901100    BEGIN
00901200    % FIRST, SAVE THE DATA THAT IS SPECIFIC TO MPTEST.
00901300    INITIALIZE(T);
00901400    COPY (TEST)MPTEST/CONTROL AS (TEST)SAVE/MPTEST/CONTROL,
00901500         (TEST)MPTEST/OLEDB/= AS (TEST)SAVE/MPTEST/OLEDB/=,
00901600         (TEST)MPTEST/CODESF/= AS (TEST)SAVE/MPTEST/CODESF/=,
00901700         (TEST)MPTEST/USERF/= AS (TEST)SAVE/MPTEST/USERF/=,
00901800         (TEST)MPTEST/USERGROUPF/= AS (TEST)SAVE/MPTEST/USERGROUPF/=
00901900    FROM DISK TO DISK [T];
00902000    IF T(VALUE) NEQ 0 THEN
00902100      ABORT "COPY from MPTEST to SAVE failed";
00902200
00902300    % NEXT, REMOVE THE EXISTING MPTEST FILES.
00902400    REMOVE (TEST)MPTEST/= FROM DISK;
00902500
00902600    % COPY THE (PROD)MPDB FILES TO (TEST). THIS MAY REQUIRE A PRIVILEGED USER.
00902700    INITIALIZE(T);
00902800    COPY (PROD)MPDB/= AS (TEST)MPTEST/= FROM DISK TO DISK [T];
00902900    IF T(VALUE) NEQ 0 THEN
00903000      ABORT "COPY MPDB to MPTEST failed";
00903100
00903200    % NOW REMOVE LOCKEDFILE FROM THE COPIED FILES.
00903300    ALTER (TEST)MPTEST/= (LOCKEDFILE=FALSE);
00903400
00903500    % FIX UP THE GLOBAL DATASET FILENAME
00903600    CHANGE (TEST)MPTEST/MPDB/DATA TO (TEST)MPTEST/MPTEST/DATA FROM DISK;
00903700
00903800    % NOW COPY THE SAVED MPTEST FILES BACK OVER THE NEW ONES FROM MPDB.
00903900    INITIALIZE(T);
00904000    COPY (TEST)SAVE/MPTEST/= AS (TEST)MPTEST/= FROM DISK TO DISK [T];
00904100    IF T(VALUE) NEQ 0 THEN
00904200      ABORT "COPY from SAVE to MPTEST failed";
00904300
00904400    % FINALLY, THE THING THAT MAKES THIS APPROACH WORK -- DO A DMCONTROL
00904500    % RECOVER UPDATE ON THE SAVED DATABASE CONTROL FILE WITH CURRENT AUDIT NR.
00904600    INITIALIZE(T);
00904700    RUN *SYSTEM/DMCONTROL(
00904800        "DB=*MPTEST ON DISK RECOVER UPDATE AUDITNUM=" & AUDITNBR) [T];
00904900    IF T ISNT COMPLETEDOK THEN
00905000      ABORT "DMCONTROL RECOVER UPDATE failed";
00905100
00905200    % IF ALL OF THAT WORKED, FIRE OFF THE HANGAROUND TO SEE IF IT CAN OPEN THE DB.
00905300    START (TEST)WFL/MPTEST/HANGAROUND;
00905400    END;
00905500
00905600  DISPLAY "MPDB to MPTEST database clone completed";
00999900  ? END JOB
```