

A VSAM/ICF Catalog White Paper

From Mainstar Software Corporation



Catalogs Do Break! — Part 2 *A Real-Life Story*

By Ronald K. Ferguson

Preface: *"We haven't had a catalog failure in 12 years!". That's the response we recently heard from the sysprog at a very large data center (to remain unnamed, to protect the guilty). We were preparing to demonstrate some Catalog RecoveryPlus backup and recovery features at this installation, and our questions about how this installation had prepared for an ICF catalog failure was what elicited this seemingly rash statement.*

We were to soon learn that this sysprog, for the test system we were running on, was so complacent about ICF catalogs that he wasn't even backing them up, much less testing to determine if they had any catalog failures! The system we were running on was just a test system, and with this attitude, it wasn't considered important to worry about the integrity of ICF catalogs. The more we delved into the situation, the more we learned just how catastrophic an ICF catalog failure would be. How this scenario played out might be a good illustration of just how complacent a lot of people are these days about ICF catalog failures.

Trolling For ICF Catalog Failures

Following our meeting, we took the statement from the sysprog as sort of a challenge, to see for ourselves whether the ICF catalog environment was as clean as he thought it was. In addition to the demonstration jobs on sample ICF catalogs that we would create just for this purpose, we decided to check out the existing catalogs on the test system we were running on.

We first set up an IDCAMS EXAMINE and DIAGNOSE ICFCATALOG job for every ICF catalog that was accessible on this system. The job that caught our attention is shown in *Figure 1*.

```
//TEST1A JOB ...
//S1 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  VERIFY FILE(INCAT)
/*

IDCAMS SYSTEM SERVICES                                TIME: 20:37:02          09/07/01    PAGE 1
  VERIFY DATASET(UCAT.ICFCAT09)
IDC0001I FUNCTION COMPLETED, HIGHEST CONDITION CODE WAS 0
  EXAMINE NAME(UCAT.ICFCAT09) -
    INDEXTST -
    DATATEST

IDC01700I INDEXTST BEGINS
IDC11724I DATA COMPONENT CA NOT KNOWN TO SEQUENCE SET
IDC21700I MINOR ERRORS FOUND BY INDEXTST
IDC01701I DATATEST BEGINS
IDC11724I DATA COMPONENT CA NOT KNOWN TO SEQUENCE SET
IDC21703I MAJOR ERRORS FOUND BY DATATEST
IDC0001I FUNCTION COMPLETED, HIGHEST CONDITION CODE WAS 8
IDC0002I IDCAMS PROCESSING COMPLETE. MAXIMUM CONDITION CODE WAS 8
```

Figure 1: VERIFY and EXAMINE of broken BCS

As you can see from the listing in *Figure 1*, the EXAMINE report certainly indicates potential problems that should be looked at with this BCS. The implication of message IDC11724I, is that catalog record data within one or more CAs might not be pointed to by the index, and could lead to serious problems at some time. The fact, though, that the installation's sysprog apparently wasn't aware of any problems, would seem to indicate that this catalog wasn't experiencing catastrophic failures at this time. Our immediate suspicion was that this catalog might be 'a sleeper', just waiting to jump up at the worst possible time and create a real problem.

One significant, and interesting, aspect of the above EXAMINE report, is that the INDEXTEST and DATATEST *both* identified the same problem with the IDC11724I message, but INDEXTEST then concluded that it was a "minor error", while DATATEST (correctly) concluded it was a "major error".

All of our recommendations to date have suggested that you should run an EXAMINE INDEXTEST on your BCSs, prior to each and every backup. In this particular problem situation, though, the INDEXTEST would have given you the impression this was just a small error, and maybe even, one that you needn't worry about too much. Also, in both INDEXTEST and DATATEST, the message id ends in 'I', which means this is a warning, rather than an error situation. In truth, for this catalog, the structural error inside the catalog is very definitely a major error, and one that will lead to catastrophic failures at some point.

We next ran an IDCAMS DIAGNOSE ICFCATALOG command against this BCS, and while we were initially surprised that it showed no problems at all, if you consider that EXAMINE is looking for 'structural' problems, whereas DIAGNOSE is looking for 'data content' problems, the result is not surprising, because the problem with this particular BCS is more structural than content related.

```
// TEST1B JOB ...
//S1 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
DIAGNOSE ICFCATALOG INDDATASET(UCAT.ICFCAT09)
/*

IDCAMS SYSTEM SERVICES TIME: 20:34:48 09/07/01 PAGE 1
DIAGNOSE ICFCATALOG INDDATASET(UCAT.ICFCAT09)
IDC11374I THESE ADDITIONAL CATALOG REFERENCED VOLUMES WERE ENCOUNTERED:
VSER02
VSER03
VSER58
VSER59
VSER60

IDC0001I FUNCTION COMPLETED, HIGHEST CONDITION CODE WAS 4
IDC0002I IDCAMS PROCESSING COMPLETE. MAXIMUM CONDITION CODE WAS 4
```

Figure 2: DIAGNOSE ICFCATALOG of broken BCS

Catalog Backups

We naturally assumed that this installation would have regularly-scheduled backups of all ICF catalogs, not just those on their production system, but also on the test and development system (which is where we were running our tests). To say we were incredulous that *no catalog backups* were being taken at all on the test system is a bit of an understatement! The only thing we could conclude is that the sysprog, figuring that a test system doesn't warrant this level of integrity support, and if the unthinkable should ever happen, well, tough luck! In checking around, we found that if the unthinkable happened, months and months of work by a dozen people would be wiped out, which would have been very time consuming and difficult to recover from.

Our immediate task at this point was to set up an IDCAMS EXPORT backup for the catalog in error, since this installation doesn't currently have the Catalog RecoveryPlus product. Our goal was

to make certain that everything possible was done to protect the applications using this catalog. The EXPORT job and its output is in *Figure 3*. Note that EXPORT gave a RC of 0, but more importantly, the output shows the total of catalog records dumped was 4,622 (remember this number).

```
//TEST1C JOB ...
//S1 EXEC PGM=IDCAMS
//OUT1 DD DSN=TEST1.EXPORT.BACKUP,DISP=(,CATLG),
//          UNIT=SYSALLDA,SPACE=(CYL,(1,1),RLSE)
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
EXPORT UCAT.ICFCAT09 -
  OUTFILE(OUT1) -
  TEMPORARY
/*

IDCAMS SYSTEM SERVICES                TIME: 20:25:02          09/07/01    PAGE 1

EXPORT UCAT.ICFCAT09 -
  OUTFILE(OUT1) -
  TEMPORARY
IDC0005I NUMBER OF RECORDS PROCESSED WAS 4722
IDC0594I PORTABLE DATA SET CREATED SUCCESSFULLY ON 09/07/01 AT 20:25:02
IDC1147I IT IS RECOMMENDED THAT DIAGNOSE AND EXAMINE BE RUN BEFORE
IDC1147I IMPORT OF CATALOG
IDC0001I FUNCTION COMPLETED, HIGHEST CONDITION CODE WAS 0
IDC0002I IDCAMS PROCESSING COMPLETE. MAXIMUM CONDITION CODE WAS 0
```

Figure 3: EXPORT BCS Backup

As part of our Catalog RecoveryPlus testing, we decided to take a second backup of this BCS, using the CR+ BACKUP command. *Figure 4* shows the output, and several details should be noted.

```
//TEST1D JOB ...
//S1 EXEC PGM=CAT00010
//OUT1 DD DSN=TEST1.CRPLUS.BACKUP,DISP=(,CATLG),
//          UNIT=SYSALLDA,SPACE=(CYL,(1,1),RLSE)
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  BACKUP BCS(UCAT.ICFCAT09) -
  OUTFILE(OUT1) -
  EXAMINE(INDEXTST)
/*

CAT01001I Catalog RecoveryPlus (c) MAINSTAR SOFTWARE CORPORATION 07 SEP 2001 20.27.15 PAGE: 1
  BACKUP BCS(UCAT.ICFCAT09) -
  OUTFILE(OUT1) -
  EXAMINE(INDEXTST)
CAT02016I THE FOLLOWING DSNs/MASKS ARE TO BE PROCESSED DURING BCS PROCESSING.
UCAT.ICFCAT09

CAT02031I BASED ON THE MASKS/DSNS ENTERED, THE FOLLOWING CATALOGS HAVE BEEN SELECTED.
UCAT.ICFCAT09
CAT02045I START OF DATA UNLOAD FOR UCAT.ICFCAT09 ON 07 SEP 2001 (2001250) AT 20.27.15
CAT02092I CALLING IDCAMS TO PERFORM EXAMINE
IDC01700I INDEXTST BEGINS
IDC11724I DATA COMPONENT CA NOT KNOWN TO SEQUENCE SET
IDC21700I MINOR ERRORS FOUND BY INDEXTST
CAT02050W IDCAMS (4) ISSUED RETURN CODE 4
CAT02046I RECORD SUMMARY FOR UCAT.ICFCAT09
  CLUSTER  GDG  E  J  NONVSAM  TRUENAME  PATH  UCAT  ALIAS  OTHER  TOTAL  TOTAL BYTES
    6,587    1  0  0    6,530    13,089    0    0    0    0    26,207    3,123,309
CAT02051I NUMBER OF SPANNED RECORDS ENCOUNTERED: 0
CAT02047I END OF DATA UNLOAD FOR UCAT.ICFCAT09 ON 07 SEP 2001 AT 20.27.18
CAT02009I BACKUP FUNCTION COMPLETE. RETURN CODE 4
CAT01009I Catalog RecoveryPlus EXECUTION COMPLETE. HIGHEST RETURN CODE WAS 4
```

Figure 4: Catalog RecoveryPlus BACKUP BCS command

First, an IDCAMS EXAMINE INDEXTEST was internally issued, and it too came up with the likely structural error that we already suspected. Since we didn't ask BACKUP to run the DATATEST version of EXAMINE, the "minor error" condition was given.

Very significantly, though, notice that the total number of records backed up is 26,207 - compared to 4,722 backed up with EXPORT. Our structural error in the BCS now starts to show itself, giving us a glimpse of just how serious this error really is, if left uncorrected.

Even if this installation is using EXPORT (which they aren't running at all right now), if the structural error grew into a catastrophic error, forcing a forward recovery from a backup, there's simply no way that you'd ever be able to apply SMF data from far enough back in time to fully recover this catalog. The only hope would be a painstaking, dataset-by-dataset recovery - and that could only be done for the disk datasets - tape datasets that have catalog problems would probably never be found and fixed.

BCS Forward Recovery

At this point, it was obvious that this BCS needed a forward recovery, and the only way to accomplish this is with the Catalog RecoveryPlus RECOVER command. Approval was obtained to use an evaluation copy of the software, to perform a real repair on a production catalog (test system ... but this was a production catalog for that system).

Figure 5, on the next page, shows the successful recovery. Because the BACKUP was just taken a few minutes earlier, and because we knew there hadn't been any processing through this catalog since we first identified the error situation, we elected not to do a forward recovery, but rather, just a simple restore of the BCS and its aliases. In a real-life situation such as this, it's your choice whether to do a forward recovery or the restore, as we elected.

First, the recovery was successful - but not without extenuating circumstances. The output from the job produced over 190,000 lines, much of it removed to make the report in this White Paper more readable. Nevertheless, it contained page after page of "DUPLICATE KEYS IN BACKUP", with thousands of catalog records shown as being duplicate. There's no way to know what caused this, and with no way to be certain which is the good record, and which are bad, the standard decision is to keep the latest creation date record, and produce a print dump of the record.

As part of the standard processing, RECOVER deletes and redefines the user catalog, which removes all of the alias records from the master catalog, necessitating the re-insertion of all aliases. In this instance, the user catalog had 41 total aliases, and these were all successfully applied to the master catalog.

```

//YCJRES1Z JOB ...
//SI EXEC PGM=CAT00010
//INFILE1 DD DSN=YCJRES1.CRPLUS.VSBOX09.BKUP3,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
    RECOVER BCS(UCAT.ICFCAT09) -
        INFILE(INFILE1)
/*
CAT01001I Catalog RecoveryPlus (c) MAINSTAR SOFTWARE CORPORATION 07 SEP 2001 21.35.43 PAGE:1
    RECOVER BCS(UCAT.ICFCAT09) -
        INFILE(INFILE1)
CAT04079I DEFAULT IN EFFECT: ALIAS
CAT04080I PRINT OPTION IN EFFECT: NONE
CAT04029I 21.35.43 RECOVERY STEP: CHECK BCS STATUS - STARTED
CAT04029I 21.35.43 RECOVERY STEP: CHECK BCS STATUS - COMPLETED
CAT04029I 21.35.43 RECOVERY STEP: LOCK BCS - STARTED
CAT04029I 21.35.43 RECOVERY STEP: LOCK BCS - COMPLETED
CAT04029I 21.35.43 RECOVERY STEP: PROCESSING BACKUP - STARTED
CAT04035I RECOVERY STARTED FOR BCS: UCAT.ICFCAT09          BACKUP DATE/TIME: 2001250/20:27:15
CAT04056W DUPLICATE KEYS IN BACKUP; LATEST CREATION DATE KEPT

RECORD SOURCE UNDETERMINED
00000000 00AD0034 C3004800 2DC4C2F2 E5F7F1F0 *...C...DB2V710*
00000010 C74BC4E2 D5C4C2C3 4BC4E2D5 F8C4F7F1 *G.DSNDBC.DSN8D71*
00000020 C14BE7C5 D4D7F14B C9F0F0F0 F14BC1F0 *A.XEMPL1.I0001.A0*
00000030 F0F44040 40000014 01E2E8E2 F1404040 *04 ....SYS1 *
00000040 40280125 9F00000F 01000031 C4006300 * .....D...
00000050 2AC4C2F2 E5F7F1F0 C74BC4E2 D5C4C2C4 *.DB2V710G.DSNDBD*
00000060 4BC4E2D5 F8C4F7F1 C14BE7C5 D4D7F14B *.DSN8D71A.XEMPL.*
00000070 C9F0F0F0 F14BC1F0 F0F40000 1401E2E8 *I0001.A004....SY*
00000080 E2F14040 40402801 259F0000 0F010000 *S1 .....
00000090 1E0400E2 C2D6E7F2 F4301020 0F820000 *...SBOX24...b..*
000000A0 01600000 00000000 00000000 00 *-.....

RECORD SOURCE UNDETERMINED
00000000 00AD0034 C3004800 2DC4C2F2 E5F7F1F0 *...C...DB2V710*
00000010 C74BC4E2 D5C4C2C3 4BC4E2D5 F8C4F7F1 *G.DSNDBC.DSN8D71*
00000020 C14BE7C5 D4D7F14B C9F0F0F0 F14BC1F0 *A.XEMPL1.I0001.A0*
00000030 F0F44040 40000014 01E2E8E2 F1404040 *04 ....SYS1 *
00000040 40280114 2F00000F 01000031 C4006300 * .....D...
00000050 2AC4C2F2 E5F7F1F0 C74BC4E2 D5C4C2C4 *.DB2V710G.DSNDBD*
00000060 4BC4E2D5 F8C4F7F1 C14BE7C5 D4D7F14B *.DSN8D71A.XEMPL.*
00000070 C9F0F0F0 F14BC1F0 F0F40000 1401E2E8 *I0001.A004....SY*
00000080 E2F14040 40402801 142F0000 0F010000 *S1 .....
00000090 1E0400E2 C2D6E7F2 F4301020 0F820000 *...SBOX24...b..*
000000A0 01600000 00000000 00000000 00 *-.....

CAT04029I 21.35.54 RECOVERY STEP: PROCESSING BACKUP - COMPLETED
CAT04029I 21.35.54 RECOVERY STEP: REWRITE TO BCS - STARTED
CAT04029I 21.35.57 RECOVERY STEP: REWRITE TO BCS - COMPLETED
CAT04029I 21.35.57 RECOVERY STEP: UNLOCK OF BCS - STARTED
CAT04029I 21.35.57 RECOVERY STEP: UNLOCK OF BCS - COMPLETED
CAT04029I 21.35.57 RECOVERY STEP: APPLYING ALIASES - STARTED
CAT04038I ALIAS NAMES FROM BACKUP; TOTAL: 41
CAT04038I ABC110
CAT04038I AMN110
CAT04038I BBC110
CAT04038I DB2D
CAT04038I DB2DIC
CAT04038I DB2T
CAT04038I DB2LOGA
CAT04038I DB2P
... and on and on for a total of 41 aliases
CAT04039I ALIASES APPLIED TO CATALOG: MCAT.SYSA.SYSCTA
CAT04029I 19.36.00 RECOVERY STEP: APPLYING ALIASES - COMPLETED
CAT04036I RECOVERY COMPLETED FOR BCS: UCAT.ICFCAT09
CAT04009I RECOVER PROCESS COMPLETED; RETURN CODE: 4
CAT01009I Catalog RecoveryPlus EXECUTION COMPLETE. HIGHEST RETURN CODE WAS 4

```

Figure 5: Catalog RecoveryPlus RECOVER Command

Conclusion

We can't stress just how true this is: **Catalogs do break!** Too many people still think otherwise, but hardly a week goes by that we don't find more and more original ways that catalogs are still breaking - and in more serious ways than we ever thought possible. Maybe the problem is that Catalog Management is so tolerant and forgiving that you're lulled into a sense of false security. Chances are, when the inevitable catalog failure occurs, recovering from it is going to be worse than ever if you aren't ready for it. As this White Paper illustrates, if you don't have Catalog RecoveryPlus for that recovery, your chances for succeeding with that recovery aren't as high as they should be.

Ronald K. Ferguson - is Founder, President & CEO of Mainstar Software Corporation

Ron Ferguson has a technical background in large-scale OS/390 systems. As a software instructor for 20+ years, he has presented over 600 courses on VSAM and ICF catalogs, and is recognized worldwide as an expert in these areas. Ferguson travels widely, meeting with customers and presenting at national and international conferences.