ICF Catalog Management Overview

Janet Sun
Rocket Mainstar
jsun@mainstar.com
Session 9965
Agenda

• What is an ICF Catalog?
• Catalog Management
• Catalog Problems
• Catalog Maintenance Procedures
What is an ICF Catalog?
Catalog Structures

An ICF Catalog Consists of:

- A BCS (Basic Catalog Structure)
- One or more VVDSs (VSAM Volume Data Sets)
Catalog Structures

The BCS

• Is physically a VSAM KSDS
• Records have keys
• Limited to 123 extents
• Contains entries for all data sets, non-VSAM and VSAM, on tape or DASD
• Contents consist of associative information, volser, and other information needed to locate a data set

➤ Data set attribute, extent, and statistic information is not in the BCS – it’s in the VVDS
Catalog Structures

The BCS – Record Types

- Non-VSAM record (type code ‘A’)
- GDG sphere record (type code ‘B’)
- Cluster sphere record (type code ‘C’)
- Data component (type code ‘D’)
- Cluster extension record (type code ‘E’)
- Alternate index (type code ‘G’)
- Index component (type code ‘I’)
- GDG extension record (type code ‘J’)
- Path record (type code ‘R’)
- Truename record (type code ‘T’)
- User catalog connector records (type code ‘U’)
- Alias records (type code ‘X’)

SHARE in Orlando 2011
Catalog Structures

The BCS can be:

- A user catalog
- A master catalog

Alias Names:

<table>
<thead>
<tr>
<th>UCAT AB1</th>
<th>UCAT AB2</th>
<th>UCAT AB3</th>
<th>UCAT AB4</th>
<th>UCAT AB5</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABC1</td>
<td>CDE1</td>
<td>FGH1</td>
<td>JKL1</td>
<td>MNO1</td>
</tr>
<tr>
<td>ABC2</td>
<td>CDE2</td>
<td>FGH2</td>
<td>JKL2</td>
<td>MNO2</td>
</tr>
</tbody>
</table>
Catalog Structures

The VVDS

- Is physically a VSAM ESDS
- Records have component names, not keys
- Access is direct by RBA to a control interval
- Contains the BCS names for the data sets on the volume
- Record types are:
  - VSAM Volume Record (VVR) for a VSAM component
    - Type ‘Z’ for a primary record (DSNs, attributes, extents)
    - Type ‘Q’ for a secondary record (DSNs, extents)
  - Non-VSAM Record (NVR) for non-VSAM data sets
    - Type ‘N’ record (DSNs)
  - Various control records (VVCR, VVCN, VVCM)
Catalog Structures

VVDS Rules

- The name of the VVDS is:
  \texttt{SYS1.VVDS.Vvolser}

- The size of the VVDS is your choice and should be at least 1 cylinder
  - Implicitly allocated if you don’t allocate – the default size will be TRK(10,10)

- Since z/OS 1.7 a system default can be set
  - Default is not preserved across an IPL
  - \texttt{F CATALOG,VVDSSPACE(prim,sec)}
    - \textit{Space values specified are in tracks}
Catalog Structures

Volume Table of Contents (VTOC)

- A system data set which maintains extent and allocation information for a volume
- Used to find empty space for new allocations and to locate non-VSAM data sets
- Used to obtain information not kept in the VVDS for all VSAM data sets and SMS managed non-VSAM data sets
- Data Set Control Blocks (DSCBs)
  - “F0” – Free DSCB
  - “F1” – DSN and 3 extent definitions
  - “F3” – 13 extent definitions
  - “F4” – VTOC “DSN” definition
Catalog Structures

BCS Relationships

• For a VSAM component – the BCS points to the VVR in the VVDS
• For non-VSAM data sets – the BCS points to the VTOC Format 1 (“F1”) DSCB
  • Even if they are SMS managed
Working with Catalogs
Catalog Management

Identifying the Master Catalog at IPL

- Specified in the SYSCAT statement of LOADxx
  - In SYSn.IPLPARM or SYS1.PARMLIB
  - The value of “xx” is specified on the Hardware Management Console (HMC)

- If LOADxx is missing, then the location of the master catalog is specified via the SYSCATyy member of SYS1.NUCLEUS
  - The operator is prompted for “yy” at IPL
  - The default member is SYSCATLG
Catalog Management

SYSCAT Statement in LOADxx

SYSCAT volserxycscatname

*-----1-------2-------3-------4-------5-------6-------7--
*        VVVVVV    |<------- CATALOG DATASET NAME ------->|VVVVVVVV
*        |    |                                                | HLQ  |
*        |    |                                                | OF   |
*        CATVSN                                                | TCDB |

- Columns 10-15 – The volume serial of the master catalog
- Column 16 – The character ‘1’ unless SYS% to SYS1 conversion is active, in which case this will be a ‘2’
- Column 17 – Alias name level of qualification (default: 1)
- Columns 18-19 – CAS service task lower limit (default: X’3C’)
- Columns 20-63 – The 44-byte data set name of the master catalog
- Columns 64-71 – The high level qualifier of the tape volume catalog (default: SYS1)
- Column 72 – Specify ‘Y’ to enable AUTOADD when CAS makes the first connection to the coupling facility
Catalog Management

SYSCATyy Member in SYS1.NUCLEUS

Bytes

<table>
<thead>
<tr>
<th>VVVVV</th>
<th>&lt;-------- CATALOG DATASET NAME --------&gt;</th>
<th>VVVVVVV</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>HLQ</td>
</tr>
<tr>
<td>VOL</td>
<td></td>
<td>OF</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TCDB</td>
</tr>
</tbody>
</table>

- Columns 1-6 – The volume serial of the master catalog
- Column 7 – The character ‘1’ unless SYS% to SYS1 conversion is active, in which case this will be a ‘2’
- Column 8 – Alias name level of qualification (default: 1)
- Columns 9-10 – CAS service task lower limit (default: X’3C’)
- Columns 11-55 – The 44-byte data set name of the master catalog
- Columns 55-62 – The high level qualifier of the tape volume catalog (blank means there is no tape volume catalog)
- Column 63 – Specify ‘Y’ to enable AUTOADD when CAS makes the first connection to the coupling facility
Catalog Management

Types of Entries in a Master Catalog

- System related data sets
  - SYS1 and other data sets needed at IPL time
  - Page data sets
  - IODF

- User catalog connector records
  - Created with IDCAMS IMPORT CONNECT

- Alias records
  - Created with IDCAMS DEFINE ALIAS
  - An alias name defined in the master catalog can be used to reference a user catalog
Catalog Management

Defining a Catalog

- Use IDCAMS DEFINE USERCATALOG
- Cannot span volumes
- Cannot be defined as a striped data set
- Can be an Extended Addressibility data set with z/OS 1.12 or higher
- Allocate in cylinders
  - Results in maximum possible CA size of 1 cyl
- Specify secondary space value > 1 cyl
  - To prevent each CA split from requiring another extent
Catalog Management

Defining a Catalog (continued)

- Recommend Data CISIZE of 4KB
  - Provides a compromise between minimizing data transfer time and reducing the occurrence of spanned records
- Use minimum Index CISIZE of 3584 if using a 4KB Data CISIZE
  - Don’t take the default
- Start with STRNO(3)
  - Default is 2
- Don’t code BUFFERSPACE
  - Use BUFND and BUFNI instead
- Take the defaults for SHAREOPTIONS and RECORDSIZE
Catalog Management

Example of Defining a Non-SMS Managed Catalog

```
//DEFCAT   EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*  
//SYSIN     DD *  
DEFINE USERCATALOG -  
   (NAME(CATALOG.USERCAT1) -  
   VOLUMES(VOL001) -  
   ICFCATALOG -  
   CISZ(4096) STRNO(3) -  
   CYLINDERS(20,5))
/*
```
Catalog Management

Example of Defining an SMS Managed Catalog

```plaintext
//DEFCAT EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=* //SYSIN DD *

DEFINE USERCATALOG -
   (NAME(CATALOG.USERCAT1) -
      MGMTCLAS(MCVSAM) -
      STORCLAS(SCSMS) -
      DATACLAS(DCVSAM) -
      ICFCATALOG -
      CISZ(4096) STRNO(3) -
      CYLINDERS(20,5))
/*
Catalog Management

BCSs Can Be Shared

- **If sharable:** SHR(3 4), the default, tells catalog management the BCS is expected to be shared, so all necessary shared access integrity code is executed.
- **If not:** SHR(3 3) tells catalog management the BCS is not going to be shared, so no shared access integrity code is executed.
- The BCS must be on a shared DASD device.
Catalog Problems
Catalog Problems

Things go bump in the night …

- Volumes fail and have to be restored
- Catalogs (BCSs and VVDSs) become corrupted and have to be restored
- Data sets become uncataloged, don’t ever get cataloged, or are cataloged to the wrong catalog
- Data sets are cataloged, but don’t exist
- CAS or caching problems
Problem Indicators

- Inconsistencies between a BCS and its own records, or between a BCS and its related VVDSs
  - Cluster records with missing truename records
  - Orphaned truename records (truename without a matching cluster sphere record)
  - Truename loop failure (cluster sphere record that points back to a different cluster sphere record)
  - Invalid data content inside a BCS or VVDS
  - BCS cluster sphere record that is missing its corresponding VVDS record (or vice versa)
Problem Indicators

• Missing BCS self-describing cluster sphere record
  ✓ *Is always the first record in the BCS*

• Broken sequence set (SSI) forward address chain pointer problem
  ✓ *Can be caused by using incorrect share options (SHR) when defining the BCS*

• Duplicate or out of sequence records in BCS
  ✓ *Can be caused by a system or CAS crash in the middle of a CI or CA split*
Finding Catalog Problems

Using Access Method Services (IDCAMS)

- **EXAMINE INDEXTEST** – ensures that sequential and key direct access is accurate
- **EXAMINE DATATEST** – reads all data CIs to ensure structural integrity
- **DIAGNOSE ICFCATALOG** (without compare) – checks information integrity within each BCS record (inside-the-BCS only)
- **DIAGNOSE VVDS** (without compare) – checks information integrity within each VVDS record (inside-the-VVDS only)
- **DIAGNOSE** is a tool that you use to see synchronization problems between the BCS and VVDS record structure
IDCAMS EXAMINE

Using IDCAMS EXAMINE to Find Problems

//STEP1 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*  
//SYSIN DD *
   EXAMINE NAME(CATALOG.UCATTEST) INDEXTEST DATATEST
/*
Messages from IDCAMS

EXAMINE

- You want to see:
  - IDC01724I INDEXTEST COMPLETE – NO ERRORS DETECTED
  - IDC01709I DATATEST COMPLETE – NO ERRORS DETECTED

- You may get other messages…
  - IDC0xxxx – Supportive informational messages
  - IDC1xxxx – Individual data set structural error messages
  - IDC2xxxx – Summary error messages
  - IDC3xxxx – Function not performed error messages
Messages from IDCAMS

BCS EXAMINE With Errors

EXAMINE NAME(CATALOG.ICF.VTSO002) INDEXTEST DATATEST

IDC01700I INDEXTEST BEGINS
IDC01724I INDEXTEST COMPLETE - NO ERRORS DETECTED
IDC01701I DATATEST BEGINS

IDC11733I DATA COMPONENT KEY SEQUENCE ERROR
IDC01717I DATA KEY -- TMVS328.V.D.PMTP0725.ACCINFO.IND
IDC01714I ERROR LOCATED AT OFFSET 00000009

IDC11734I SEQUENCE SET AND DATA CI KEY SEQUENCE MISMATCH
IDC01716I INDEX KEY -- TMVS328.V.D4.PMCOPY1.PDBAUEPC.D
IDC01717I DATA KEY -- TMVSF5D.QAREPRO.DB2.EXTRACT
IDC01714I ERROR LOCATED AT OFFSET 00000FF6

IDC21703I MAJOR ERRORS FOUND BY DATATEST
IDC0001I FUNCTION COMPLETED, HIGHEST CONDITION CODE WAS 8
IDCAMS DIAGNOSE

Using IDCAMS DIAGNOSE

//STEP1    EXEC PGM=IDCAMS
//INCAT    DD DISP=SHR, DSN=CATALOG.UCATTEST
//SYSPRINT DD SYSOUT=*  
//SYSIN     DD *
   DIAGNOSE ICFCAT INFILE(INCAT)
/*
Messages from IDCAMS

DIAGNOSE

• IDC21364I – ERROR DETECTED BY DIAGNOSE:
  {VVDS|ICFCAT} ENTRY: entry RECORD: rec OFFSET: offset
  REASON: reason-code

• This message produces a return code of 8
• Some common reason-codes…
  • 11 – Incomplete delete detected
  • 20 – Association not found
  • 23 – Truename loop failure
  • 33 – Incomplete update detected
  • 45 – Volume cell not found
Fixing Catalog Problems

If EXAMINE Identified the Problem
- Can delete data set and recover from recent backup
- Can unload a non-BCS KSDS data set by reading data component directly as an ESDS, sorting data, and then loading into new KSDS
- If a BCS index component is severely damaged, the BCS must be recovered from backup
  - Repair can be accomplished with IBM Tivoli Advanced Catalog Management for z/OS
Fixing Catalog Problems

If DIAGNOSE Identified the Problem

- May indicate an incomplete catalog entry
- If it is an entry in a BCS, delete the catalog record and attempt to recatalog

  DELETE xxx NOSCRATCH
  DEFINE xxx … RECATALOG

- If the truename exists without the associated cluster records:
  DELETE xxx TRUENAME
Fixing Catalog Problems

If DIAGNOSE Identified the Problem (continued)

• If it is an entry in a VVDS:
  
  DELETE xxx VVR
  
  DELETE xxx NVR

• It may be possible to recatalog the data set
  
  DEFINE CLUSTER(NAME(xxx) … RECATALOG)

• If missing some portions from the volume, then it must be deleted
  
  ➢ IDCAMS cannot recreate the data
Catalog Backup
Catalog Backup

IDCAMS EXPORT

- Unloads the catalog records and alias names
- EXPORT may not be able to back up the BCS if it is damaged
- If there is a broken sequence set chain pointer, EXPORT will back up only what can be accessed by the index
  - Many records may not be backed up, but a return code of 0 will be received
Catalog Backup

Using IDCAMS EXPORT to Back Up Catalogs

//STEP1    EXEC PGM=IDCAMS
//OUTCAT   DD DSN=BACKUP.CATALOG.TEST,DISP=(NEW,CATLG),
//          SPACE=(CYL,(20,5)),UNIT=SYSDA
//SYSPRINT DD SYSOUT=*  
//SYSSIN   DD *  
    EXPORT CATALOG.UCATTEST OUTFILE(OUTCAT) TEMPORARY
/*
Back up as often as you can

- How often is enough? It depends…
  - At least once a day for all catalogs
  - More often for volatile catalogs – where you are creating lots of SMF records – indicating heavy data set DEFINEs, DELETEs, and allocation extensions
  - More often for critical catalogs – ones that would present a major problem if recovery isn’t fast
  - Less often for non-volatile catalogs
Backing Up the BCS

Rule #2: Verify all BCSs are included in the backup

- When was the last time you audited your backup job to see the list of catalogs backed up?
- Obtain a list of connected catalogs in all master catalogs

```
LISTCAT UCAT
  LISTING FROM CATALOG -- CATALOG.MASTER.CAT
USERCATALOG --- CAT.ICF.USER1
USERCATALOG --- CAT.ICF.USER2
...
```

- Compare the list to your catalog backup job, and ensure that all are backed up
Backing Up the BCS

Rule #3:  Double check that the backups are working

- Establish a regular method to check catalog backup return codes
- Ensure you run EXAMINE INDEXTTEST on each catalog – and then check the output!
- Consider duplexing your backups – and create a third copy for your disaster recovery (DR) site
Back up the BCS

Rule #4: Verify that you can recover

- **Can you locate your backups?** If duplexed, catalog each in a different user catalog
- **Can you locate your SMF data?** How many systems are sharing the catalog? What catalog is it cataloged in?
- **Test, test, test** – if a problem occurs and you can't recover, your goose is ...
Catalog Recovery
Catalog Recovery

- Catalogs can be recovered/restored with IDCAMS IMPORT (after having been EXPORTed)
- Because of the dynamic nature of catalogs, this process is useful only when a backup has been taken immediately before the recovery
Recovery Using IDCAMS

Using IDCAMS IMPORT to Recover a Catalog

//STEP1 EXEC PGM=IDCAMS
//INCAT DD DSN=BACKUP.CATALOG.TEST,DISP=SHR
//SYSPRINT DD SYSOUT=* 
//SYSIN DD *
    IMPORT INFILE(INCAT) OUTDATASET(CATALOG.UCATTEST) ALIAS
/*
Catalog Forward Recovery
BCS Forward Recovery

SMF Records Are the Only Way

- For BCS forward recovery, SMF record types required:
  - Type 61 – Data set define
  - Type 65 – Data set delete
  - Type 66 – Data set alter
- These records, written between the time of backup and restore, identify all new data sets created, deleted, and extended
- You must have something, and you must know how to use it!
- You should practice BCS forward recovery
ICFRU

Used for BCS Forward Recovery

- IBM field developed product
- Incorporated into DFSMS 1.7 and higher
- Takes IDCAMS EXPORT copy of the BCS and SMF records from all sharing systems as input
- Creates a new EXPORT format backup which is used as input to IDCAMS IMPORT to rebuild the catalog
ICFRU

Components of ICFRU

- **CRURRSV** – Record Selection and Validation
  - Processes dumped SMF data sets
  - Extracts appropriate records
- **CRURRAP** – Record Analysis and Processing
  - Processes the extracted and sorted SMF records, together with an EXPORT copy of the catalog
  - Produces a new EXPORT format data set to be imported to build a new catalog
Other Catalog Maintenance Activities
Reorganizing Catalogs

What Is Reorganization and When Do I Do It?

- The process by which a catalog is backed up and then immediately recovered to rebuild the index and data components and is commonly used to:
  - Remove CA and CI splits
  - Reduce extents
  - Change an attribute
- Recommended only when the catalog is approaching maximum extents, or there has been a large amount of data set deletion activity against the catalog
- IDCAMS EXPORT followed by IMPORT can be used to reorganize a catalog
Splitting and Merging Catalogs

What Is Splitting and Merging?

- The process by which a group of data set entries are moved or copied from one catalog to another for space, performance, or other reasons
- Catalogs may be combined (merged) or divided (split) to achieve the best balance
Splitting and Merging Catalogs

Using IDCAMS

- IDCAMS REPRO MERGECAT is used to move data set entries either individually, by a high level qualifier group, or all entries from one catalog to another
  - The output catalog may already have other data set entries in it
  - Entries are deleted from the input catalog after they are successfully added to the output catalog
  - The VVRs for the data set entries moved are updated to point to the new output catalog
Using IDCAMS (continued)

- IDCAMS REPRO NOMERGECAT is used to **copy** the entire input catalog into a new, empty output catalog
  - The output catalog must be empty
  - The VVRs for the data set entries copied are updated to point to the new output catalog following REPRO NOMERGECAT processing
    - **All subsequent processing must be done to the new output catalog**
Disaster Recovery Planning

What About that Disaster Recovery (DR) Test?
• Disaster recovery is not your everyday task to perform, but it needs to be prepared for every day
• Plan the timing of your DR catalog backups so that you can know the state of the corresponding data to be recovered
• You want to have the catalog entries synchronized with the data recovered as closely as possible
Summary of Activities

Daily Activities
- Backup
- Diagnostics
- Disaster recovery backups

Periodic Activities
- Catalog reorganization
- Splitting or merging
Final Thoughts

- Daily activities ensure catalog integrity and prevent loss of data
- Minimizes risk of catastrophic catalog failure
- Ensures readiness in case of disaster
- Improves catalog recoverability in case of catalog failure
For More Information

- *z/OS DFSMS Access Method Services for Catalogs* – SC26-7394
- *z/OS DFSMS: Managing Catalogs* – SC26-7409
- *ICF Catalog Backup and Recovery: A Practical Guide* – IBM Redbook SG24-5644
Trademarks

The following terms are trademarks of Rocket Software, Incorporated in the United States and/or other countries:

   Rocket®, Mainstar®

The following terms are trademarks of the International Business Machines Corporation in the United States and/or other countries:

   IBM®, Tivoli®, z/OS®, DFSMSdfp™

Other company, product, and service names may be trademarks or service marks of others.

Copyright ©2011 Rocket Software, Inc. All Rights Reserved.