



Ensuring Catalog and Control Data Set Integrity

Part I: Catalog Integrity

Steven Clar and Janet Sun

For reliable data access, your ICF catalogs and DFSMSHsm Control Data Sets must be accurate, consistent, and available. Unfortunately, effective management of these critical structures can be complicated and time-consuming. In Part I, discover how to keep your catalogs healthy – and still have time to go home at night.

Mainstar
Software
Corporation

P.O. Box 4132
Bellevue, WA 98009

www.mainstar.com

info@mainstar.com

Why do catalogs and Control Data Sets matter?

Because ICF catalogs and DFSMSHsm Control Data Sets (CDSs) manage and provide access to your data, their integrity is essential for your business to stay up and running. Both are part of the system infrastructure that supports your applications. By keeping these fundamental structures healthy, you can enjoy the benefits of a more resilient environment – and avoid costly outages to applications and data.

In the first article of this series, we'll discuss your ICF catalogs and how you can ensure their integrity.

ICF Catalogs

When a catalog is unavailable, all applications that access that catalog suffer an outage. Substantial amounts of business data will be unavailable until the catalog is recovered. In other words, ICF catalogs are a single point of failure.

Compare the loss of a catalog to the loss of your company phone and email list – how would you contact someone when you needed them? Similarly, with just one catalog out of commission, you can't access the data that it managed – even the data that's essential for your basic business operations.

Here are some typical catalog problems:

- Volumes crash 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 develop.

So, why do these problems occur?

Commonly, BCS, VVDS, and VTOC records get out of sync. As a result, many data centers have an unclean catalog environment – and don't even know it. This can cause unforeseen circumstances to occur, when you least expect it, and oftentimes when you're trying to manipulate something in your catalog.

For example, IBM documentation clearly states that you should DIAGNOSE and clean up any errors between a BCS and its VVDSs before you ever attempt a REPRO

MERGE CAT. The reason is, there are errors that creep into catalogs that don't cause processing problems on a day-to-day basis, but can make it impossible to perform a MERGE CAT operation, or in the worst case, can severely damage your BCSs involved in the MERGE CAT.

If you establish a regular and consistent schedule for running DIAGNOSE across your entire environment, and fixing the errors as they occur, your catalogs will be in considerably better shape, and when it comes time to do something like a MERGE CAT, the process will be less prone to error.

Achieving Catalog Integrity

By implementing an effective catalog management strategy, you can avoid the majority of catalog problems. Here are the key techniques for outage prevention:

- Examine and diagnose regularly, and resolve problems
- Back up frequently
- Reorganize when needed

In addition, your organization needs a forward recovery strategy, so that you can recover swiftly if an outage *does* occur.

Diagnose Regularly and Resolve Problems

A DIAGNOSE facility should be a day-to-day tool that you use to see 'disconnects' between the BCS and VVDS record structure.

If you're counting on IDCAMS DIAGNOSE, you have a huge job on your hands. In today's world, setting up an IDCAMS DIAGNOSE between a BCS and its related VVDSs is incredibly unwieldy, and even if you do, the output is typically quite large. You have to review it, you have to determine the problems, and you have to create the fixes yourself and run them. This simply isn't practical!

With Mainstar's ICF catalog management solution, Catalog RecoveryPlus (CR+), catalog diagnostics are much more straightforward.

CR+ DIAGNOSE provides an extensive multi-volume data set analysis. Two of the primary modes of operation for the CR+ DIAGNOSE command are:

- DIAGNOSE BCS-VVDS: compares BCSs towards VVDSs to find catalog entries for data sets which don't exist.

There are errors that creep into catalogs that don't cause processing problems on a day-to-day basis, but can make it impossible to perform a MERGE CAT operation.

- **DIAGNOSE VOLUME-BCS:** compares from the volume perspective (VVDS and VTOC) back towards BCSs to find uncataloged data sets and orphaned components.

The output from these CR+ DIAGNOSE commands includes extensive problem descriptions, so you can identify and understand each problem, easily and thoroughly.

At your request, a fix file is generated, containing all the IDCAMS or CR+ commands needed to correct the problems that were found. In some cases, the fix is so extensive that it might require additional research on your part.

The CR+ DIAGNOSE VVDS-VTOC command differs somewhat from the previous two commands. It performs an analysis of the VVDS and VTOC only, and you will want to run it in SIMULATE mode first. It is not possible to create IDCAMS fixes for the types of errors identified here, so the AUTOFIX mode is used to make the corrections directly after you have reviewed the results from the simulation.

The CR+ MAP command simplifies the management and tuning of your VSAM KSDS data sets including BCSs. MAP helps you identify data insertion patterns, splits, and dead CIs and CAs. Generally, you would want to run MAP against a data set repeatedly over a period of time to determine the amount and location of activity within the cluster. CR+ MAP provides a recommendation report that gives options for CFSIZE changes and suggested buffering values, as well as identifying incorrect file allocation parameters.

Back up your catalogs as frequently as you can – at least once a day, but more often if possible.

Back Up Frequently

Frequent backups lead to successful and speedy recoveries, because they ensure you have the most up-to-date data possible. Back up your catalogs as frequently as you can – at least once a day, but more often if possible. Following these best practices will help you ensure that these backups are effective:

- Make at least two copies of the backup file.
- Establish a procedure to verify backup return codes.
- Ensure that you know where your SMF data is located.
- Execute IDCAMS diagnostics:
 - EXAMINE INDEXTEST with every backup.
 - EXAMINE DATATEST and DIAGNOSE BCS on a regular basis: daily, weekly, or monthly as resources permit.

With the BACKUP BCS command in CR+, you can use

powerful mask filtering operands to specify which BCSs to back up. Even better, simply specify ** to pick up every catalog that's connected to the master, including the master itself.

All BCSs that are selected in one BACKUP command are written to a single output file, from which any BCS can subsequently be selected for recovery.

Along with the catalog's records, all associated aliases for the BCS are backed up from the current system's master catalog, and these aliases will be restored by default with the RECOVER command. In addition, the BCS's definition is backed up, enabling the CR+ RECOVER command to automatically perform the DELETE and DEFINE USERCATALOG command for you. You can modify many of the allocation attributes right on the RECOVER command, and they will be applied to this definition.

The CR+ BACKUP command calls IDCAMS and issues an EXAMINE INDEXTEST under the covers, by default, just prior to starting the backup for each catalog. At your discretion, this can be upgraded to a slightly more expensive EXAMINE DATATEST, or even a DIAGNOSE BCS command, so that you can determine at backup time whether the catalog has structural errors in its index or data component. Performing these tests alerts you to problems at the earliest possible time, and you can fix them before they become more serious.

Many of the CR+ commands include the option to simulate the execution of the command. For BACKUP, this lets you determine if your BCS name selection masks are set up correctly before you put the command into production. Even better, you can use SIMULATE in a regularly-scheduled production job stream to run just the periodic DIAGNOSE and EXAMINE commands in a separate step from your actual catalog backups.

Reorganize when Needed

Most installations have requirements for 24x7 data availability, so it can be difficult to schedule an outage to perform catalog maintenance. However, there are a number of common issues that normally require an outage.

Catalogs that have structural problems normally need to be taken out of service for repair. Some catalogs have what is called a creeping key problem where the key, or data set name, is always increasing. We see this most often with certain report management systems which have a date and

time stamp as a part of the data set name, and it is always increasing. If you have a lot of activity in your catalog caused by a lot of data set deletes or defines, your catalog may be a good candidate for reorganization. In addition, it is often difficult to schedule outage time for removing the obsolete IMBED or REPLICATE attributes from catalogs.

Installations can reorganize as needed without affecting business-critical applications by taking advantage of the CR+ REORG command. With the command, you can reorganize a BCS while it is open and active, even if it's being shared in VVDS-mode or ECS-mode. If you have a BCS that is running out of extents and it is difficult to schedule outage time to do a standard reorg, CR+ REORG is just what you need to keep that BCS running.

CR+ REORG automatically removes the IMBED and REPLICATE attributes. You can change other attributes during REORG processing, such as the space allocation values, free space, CI size, or number of buffers. There is also an option to release unused extents after the reorganization process is complete.

Forward Recovery

Forward recovery for an ICF catalog involves updating the BCS or VVDS with all the changes that have occurred since the time of the backup and restore. SMF records are the only practical way to forward recover a BCS or VVDS. Since there is an SMF record written for every data set define, delete, and extension, a forward recovery can be performed by merging these SMF records, collected from all systems that are sharing the BCS or VVDS, into the backup records as the RECOVER is performed. These records, fed into a BCS forward recovery facility, are your lifeline.

These SMF record types are required for BCS forward recovery:

- Type 61 – Data set define
- Type 65 – Data set delete
- Type 66 – Data set alter

This SMF record type is required for VVDS forward recovery:

- Type 60 – VVDS update

SMF type 61, 65, and 66 records track each DEFINE, DELETE, and extent alteration that you execute for a BCS. Type 60 records are created for every VVDS record update.

To help ensure these SMF records are available when you need them, the CR+ BACKUP and RECOVER commands check your system to see if SMF record collection is in effect for these record types. If it's not, CR+ delivers informational and warning messages to alert you.

CR+ RECOVER includes a SIMULATE feature, with several options. This is an extremely valuable tool, and one that we believe you should use as often as you can. With SIMULATE, you can query definition information about BCSs on the backup file. More significantly, it provides a pre-execution test to determine if a real-life command is going to work. You get to see error messages as if you actually ran the command – without deleting anything in the production environment if you've specified anything wrong.

Here is an example of how to SIMULATE a forward recover of a BCS:

```
RECOVER BCS(CATALOG.UCATA) -  
        INFILE(BACKUPDD) -  
        FORWARD(SMFFILE(SMFDD))  
        SIMULATE(FORWARD)
```

In this example, a full simulation of a forward recovery of a BCS is performed with complete safety. This is a great way to test forward recovery against your production catalogs, without ever touching them!

Conclusion

So, is the integrity of your ICF catalogs really important?

You better believe it is! A single catalog failure can have disastrous results. Fortunately, with the following tried-and-true catalog management techniques, you can avoid outages and be prepared for anything:

- Execute regular diagnostics and review results
- Perform regular and frequent backups
- Reorganize when needed.
- Prepare and practice forward recovery using simulation

Performing preventive maintenance can be quick and effective with the powerful day-to-day ICF catalog management tools in CR+, even in environments requiring 24x7 data availability. As a result, you can sleep at night, knowing that you have 100% catalog structural integrity.

Stay tuned for the second article in this series to find out how

You get to see error messages as if you actually ran the command – without deleting anything in the production environment if you've specified anything wrong.

you can ensure DFSMSHsm CDS integrity!

For more information visit our web site, www.mainstar.com, or contact us at experts@mainstar.com to arrange a personal briefing.

Steven Clar, Product Manager

During his 25 years in mainframe storage management, disaster recovery, and MVS systems, Steven Clar has supported and managed shops ranging from one terabyte of storage to over one hundred terabytes of storage, in industries ranging from banking and insurance to pharmaceutical and outsourcing companies. As a Mainstar Product Manager, he travels extensively, sharing his knowledge and discussing emerging challenges in storage management.

Janet Sun, Product Manager

Janet Sun has over 20 years of mainframe systems programming experience in the areas of operating system management, performance management, storage management, data security, web server management, and disaster recovery planning. She frequently presents at user conferences such as SHARE and the IBM Storage Symposium.

Mainstar is a registered trademark of Mainstar Software Corporation. Catalog RecoveryPlus and *FastAudit/390* are trademarks of Mainstar Software Corporation. Mainstar Software Corporation is a wholly owned subsidiary of Rocket Software, Inc. All other products or company names are used for identification purposes only and may be trademarks of their respective owners.

010-0113-01 (11/13/06)