A VSAM/ICF Catalog
Technical Paper
From Mainstar Software Corporation

MAINSTAR

Why Catalog RecoveryPlus?

Part 2: ICF Catalog Availability

By Ronald K. Ferguson

Preface: *This three-part series on Mainstar's* ***Catalog RecoveryPlus (CR+)*** *focuses on ways that you can achieve reliability, availability, and visibility – what we're calling the RAV factor – of your ICF catalogs.*

Your z/OS environment – indeed, your entire IT department – has three purposes in life: gathering, storing, and providing access to data that's important to your business. Because none of these objectives is possible unless you can access the data we're combining all three into one term: data access.

Depending on the nature of your business (or government agency, or whatever your IT department supports), if any of the mechanisms for gathering, storing, or accessing your data are not available, you could be in serious difficulty in a matter of hours (or days, depending on how critical the specific data is).

As we stressed in Part 1 of this series, your ICF catalogs are a critical metadata structure of your z/OS environment, providing you with access to your data – in other words, the availability of that data.

There are many ways you can improve the availability of your ICF catalog environment – and therefore, the availability of your data.

This article will help you identify several of the steps you can take, and hopefully, get you to thinking more about increased data availability as a priority goal you can work towards. Each aspect will be illustrated through use of ***CR+****, Mainstar's product for ICF catalog backup and recovery, diagnostic, and repair. IBM recognizes the importance of* ***CR+*** *for business resiliency and security, stating in their z/OS v1.7 Software Announcement of July 27, 2005, "IBM recommends Mainstar Catalog RecoveryPlus (5620-FGY)".*

Data Availability: How Do You Get It?

Many people refer to ICF catalogs and catalogs as if they were interchangeable terms. There's nothing wrong with that, so long as you don't lose sight that, technically, catalogs are much more than just a single structure to worry about.

In fact, you really need to keep in mind the entire range of metadata in your z/OS system that provides reliable and accurate access to all of your data. This metadata is pervasive throughout z/OS, including (but not limited to) the following:

- ICF catalogs (consisting of the BCSs and VVDSs): without them, you can't access any of your cataloged data
- VTOCs and VTOCIXs: without them, you can't access data on disk volumes
- DFSMShsm CDSs (or the equivalent structures if you use CA-DISK or FDR/ABR): without them, you can't access migrated or archived data
- Tape management catalogs: without them, you can't access tape data
- RACF (or other security system): without them working properly, you can't access data (period!)
- Job scheduling data bases: without them, you can't run jobs, and without that, you can't access data
- And on and on…

In this article, we're only concerned with the availability of ICF catalogs, and while it may not be semantically correct, we'll refer to this as "catalog availability".

Seven Ways to Increase Data Availability

1. Plan Your Recovery Well

Let's create an acronym to help you keep the essential steps in mind: PPP – Plan, Practice, and Perform.

It's surprising how many installations take backups of their catalogs, hoping they'll never have to use them, and therefore, never get around to really figuring out what they must do if a recovery from backup is ever necessary.

For example, many installations take regularly-scheduled backups with a utility program such as EXPORT, DFSMShsm, or DFSMSdss, and don't have *any* process for forward recovery. In other words, it never occurs to them that a 12-hour-old catalog backup, restored as a point-in-time copy of the failed catalog, will be out of sync in potentially devastating ways.

Catalog 'forward recovery' – a critical concept – can be defined as a process which brings a point-in-time catalog backup 'forward' from its backup time to current time, so that it matches the data that it catalogs.

A catalog forward recovery requires software that's designed to use SMF data to update the catalog point-in-time backup with all changes since the time of backup.

Interestingly, basic z/OS has never had such a basic facility imbedded with it, but since 1985 IBM has had a field developed program (FDP) called ICFRU (ICF Forward Recovery Utility) that is now incorporated into z/OS at V1.7. ICFRU isn't feature-rich, but it can get the job done, requiring a multi-step process to forward recover a catalog. As IBM states in their z/OS V1.7 Announcement Letter of July 27, 2005: "If you need a more complete solution than that provided by ICFRU, IBM recommends Mainstar **Catalog RecoveryPlus**". **CR+** can be licensed from IBM, through program number 5620-FGY, or directly from Mainstar or its distributors.

Mainstar's **CR+** includes RECOVER BCS with the FORWARD option, a one-step procedure that's easy to set up, easy to use, and provides easy to read output.

Any forward recovery facility, however, requires testing – on a regular basis – to ensure that the process you've set up still fits with your other installation methodologies, and with the experience and knowledge level of the staff that's available to do the recovery (i.e., the person who set it up might not be available when a recovery is necessary).

CR+ RECOVERY provides a SIMULATE keyword, allowing you to test this on a real catalog in your installation. SIMULATE is a perfectly safe method of testing, without any system or overhead impact to the real catalog, that helps you to feel comfortable that your recovery jobs are already set up and ready to go, with good processes and procedures already in place to identify and gather what you need for a quick and accurate recovery – including the right catalog backup, plus *all* of the necessary SMF data, from all systems in your shared catalog environment.

2. When a BCS Outage Occurs, Fast *Recovery Is Critical*

A friend who spent many years flying prop-driven de Havilland Beavers and Otters on charter to remote Canadian and Alaskan fishing lakes felt his life was too slow, and he decided it was time to plunk down $35,000 of his own money to get his biz-jet rating, so he could become a corporate pilot. He made this comment when he completed his training and got a much higher-paying job flying a Lear jet: *"there's just nothing that compares to speed".*

Well, the same is true of recovery from a catalog outage: when your business relies on critical data to prosper and survive, there's nothing that compares to a *speedy, accurate* recovery – and that means data availability.

The worst imaginable situation is for you (or anyone on your catalog management team) to find yourself *learning* how to do a catalog recovery when you already have an outage – or worse yet, setting up the recovery process at that time. This will almost certainly lengthen your recovery time by hours, not minutes, and will significantly increase the likelihood of a recovery with errors, or worse, an unsuccessful recovery.

In planning your ICF catalog recovery methodology, take the following into account:

- How will you identify – and find – the catalog backup that you need to restore? Typically, catalog backups are set up as GDGs, so pointing to the (0) generation and version is a safe bet.
- What if you want to recover to yesterday's 4:00 AM backup, and you're taking multiple

backups each day? The best plan may be setting up a pseudo, homegrown GDG facility, where the date-time of backup is right in the backup data set name. This can be done with products such as SAR, so you might want to consider ideas such as this.

- How do your SMF data sets correlate with your ICF catalog backups? SMF has been around for many years. By now, almost every installation has their homegrown processes – set up years ago – for how they handle their various SMF requirements. This includes automatic switching of the current MANx/y data sets, how they get dumped, what types of data sets they get dumped to, whether there are SMF record-type extractions for the various business processes requiring SMF data, whether there are consolidation processes in place, and so on.

It might seem like a no-brainer to understand how all of this is interrelated, but in fact, it isn't that easy or straightforward. How all of your SMF processing is handled should be an important part of your decision about catalog backup frequency, and it will be important in the process setup that you choose for your recovery methodology.

For example, let's assume you have 6 LPARs sharing access to a critical user catalog (defined as critical because it requires the fastest possible recovery if there's an unplanned outage). This catalog is backed up once a day at 4:00 AM. However, you are force-switching MANx/y data sets at midnight on all systems. Now, at 10:00 AM this catalog breaks and you need to forward recover it from the 4:00 AM point-in-time backup. It's certain that all dumped SMF data sets from midnight, from all LPARs, must be included in the forward recovery. In addition, when you decide a catalog recovery is necessary, a forced switch (I SMF) and dump of the SMF MANx/y data sets from all systems must also be performed, and then those newly dumped data sets must be included with the standard daily dumps in your forward recovery process. Also, any automatic SMF dumps that occurred between midnight and 10:00 AM from busy systems that are 'cutting' a lot of SMF records must be included.

Hopefully, this illustrates that if this process isn't carefully thought through, the risk of leaving out one or more critical SMF dump data sets can be high, with the result that the recovery might look good, but in fact, not actually be successful (and therefore, only discovered later).

Again, being able to test this in real-world situations, in real-world levels of system activity, is vital. Many people rely on tiny test catalogs that aren't shared across many LPARs, with just a few jimmied-up transactions between backup and recovery, and with SMF processes that aren't as complicated as the production environment. Worse, they are run by someone who knows exactly how they set it all up, and not with real Q&A methods for testing.

Because all of the above is commonplace for many of the environments that we see, it's also common that recovery from a catalog failure typically requires hours, not minutes.

Your installation almost certainly has a Service Level Agreement (SLA) for your disaster recovery (DR) time – i.e., the time that you're expected to be up and running with critical applications after a disaster is declared – but do you have an SLA for recovery from failure with ICF catalogs? If you have less than 100 ICF catalogs, through which you access millions of data sets, simple math should tell you that this is a single point of failure – and you need to be prepared for recovery, and importantly, prepared for a *fast* recovery!

When we ask the people who manage ICF catalogs when they had their last catalog outage, they will typically say, "Several years ago". When we ask the disaster recovery (DR) people when they had their last real disaster (for which they actually declared a disaster), they typically say, "Never". Why is it, then, that the DR plan is tested at least once a year, with people held accountable for success or failure, yet this level of planning and testing is rarely true for ICF catalogs? The level of consciousness about this needs to be raised – particularly at the management level – so that ICF catalog 'disaster planning' is at least as high in priority as overall data center DR planning.

3. VVDS Failures Are Possible – Recovering a VVDS Is Faster than Full Volume Recovery

As mentioned many times in Mainstar papers and presentations, the VVDS is 'the other half' of the ICF catalog environment. It's at least as important as the BCS, which is the half that we commonly refer to as 'the catalog', as it contains the information really necessary to open a data set.

Yet, the VVDS is given very little thought or priority in recovery thinking and planning – and that thinking should change in today's world of

very large, multi-volume data sets and huge data base subsystems.

The VVDS is physically a VSAM data set, and it can 'break' due to errors in VSAM software code just as any VSAM application data set can break from this same software. Many of the CATBREAKER-coded APARs in IBMLINK are indeed errors related to VVDS processing, but are grouped under the category of a 'catalog error'. Sometimes the errors within a VVDS are so serious and severe that SMS sets an indicator that no further allocations will occur on this volume until the error is corrected. To know about this error situation, though, requires proper diagnostics run on the VVDS – and even that's not often done.

The IBM *Managing Catalogs* reference manual (SC26-4914) states (italics added for emphasis):

> The VVDS and VTOC should not be backed up as data sets, but are *backed up as part of a full volume dump* using DFSMSdss or DFSMShsm. The entries in the VVDS and the VTOC are backed up with the data sets they describe when the data sets are backed up with the IDCAMS EXPORT command, DFSMShsm, or DFSMSdss logical dump.
>
> There are two ways that a VVDS or VTOC can be recovered:
>
> **1.** *Restore the volume* containing the VVDS or VTOC, or
>
> **2.** Rebuild the VVDS and VTOC by *recovering the data sets on the volume.*
>
> Restoring the volume is the easiest way to recover a VVDS or VTOC. However, this is seldom practical because the data sets restored will not be current. *To rebuild the VVDS, you must delete it and then recover all VSAM and SMS-managed data sets which were on the volume.*

These statements are true, but only in the context that there isn't any facility within z/OS to recover a VVDS *and synchronize it with the data on the volume*. Nevertheless, it describes a recovery methodology that is laborious – and worse – very slow. If the volume suffering a VVDS failure is part of a multi-volume data set, you have an extensive volume-level recovery on your hands that might require hours to recover.

CR+ has a facility for forward recovery of a VVDS (but not a VTOC), giving you the capability to incorporate VVDS backup and recovery in your overall ICF catalog rapid recovery plan. This facility can help you recover from a VVDS failure in minutes, rather than hours – that's data availability.

The **CR+** BACKUP VVDS command can backup VVDSs based on the following selection:

- On a single volume
- On all volumes that match a volser mask
- On all volumes in a one or more storage groups
- On all volumes on the system

The command is extremely fast to execute: approximately 30 ms per VVDS. As an example, you can back up standard VVDSs on 1,000 volumes in about 30 elapsed seconds; extrapolating further, you could back up VVDSs on 10,000 volumes in about 5 elapsed minutes. These timings are in line with what those required for backing up your catalogs, making this a viable feature.

When a VVDS failure occurs, the **CR+** RECOVER VVDS command has a forward recovery technique similar to that provided for RECOVER BCS, where SMF type 60 records are applied to the backup copy of the VVDS, bringing it forward from its backup point-in-time status to current time. Having the process set up ahead of time, tested, and ready to go when it's needed can cut recovery time to minutes compared to the volume-level recovery described in *Managing Catalogs*.

4. When 'Flaky' Data Access Situations Occur, Know How to Correctly – and Quickly – Analyze the Situation

One of the most time-consuming aspects of the typical BCS (catalog) or volume (VVDS) is identifying the problem and making the decision about how to move forward. Virtually every ICF catalog problem situation will be different from any you've ever experienced before (it's just the nature of it). Your best plan is to know ahead of time what diagnostics are available to you, how to run them, and how to analyze the output. From there, the task will be to convert that knowledge into a plan of action that correct the problem, quickly, and with the least amount of disruption to your business environment.

The diagnostic process is extensive, and

thoroughly covered in Chapter 3, Analyzing Catalog Integrity, in the IBM Redbook, *ICF Catalog Backup and Recovery*, SG24-5644. This Redbook is downloadable at no charge as a PDF file from the IBM web site (www.ibm.com).

Essentially, this chapter describes how to use system messages to help determine the error, then identify whether the error is logical within the catalog address space (CAS) control blocks, or if there is structural damage to an actual BCS or VVDS. It describes the diagnostic tools that are available – IDCAMS, ISPF, DFSMSdss, ICKDSF, and Mainstar's **CR+** – and it also explains how to use the diagnostic information from these tools, and how to set up the best repair scenario. Since catalog sharing is the source of many of today's ICF catalog problems, it explains how to analyze your catalog shared environment, to ensure it is set up correctly.

5. Catalog Reorganization

Many catalogs haven't been 'touched' in years, sometimes for fear of breaking something that isn't broken, but oftentimes because the applications using the catalog to access data are 24x7 and don't present opportunities for quiescing the catalog environment.

As a result, many catalogs have errors (some serious) within them that never get fixed, and have definition attributes that can never be cleaned up. For highly volatile catalogs that grow rapidly, critical business requirements oftentimes make it very difficult to schedule outages on applications to perform catalog maintenance, but the outage becomes inevitable at some point.

When a catalog has to be reorganized by 'normal' means, the process consists of the following steps:

- Quiescing all applications (batch and online) that have data sets cataloged in the catalog
- Backing up the catalog with IDCAMS EXPORT (or **CR+** BACKUP BCS)
- If attributes of the catalog are to be changed, issuing a user-created IDCAMS DELETE command to remove the existing catalog, then an IDCAMS DEFINE to allocate the new catalog and establish its attributes
- If attributes are not to be changed, the DELETE and DEFINE functions are automatically performed within the IDCAMS IMPORT or **CR+** RECOVER BCS
- Reloading the catalog backup using IDCAMS IMPORT or **CR+** RECOVER BCS
- Restarting the application processes that use the catalog

The **CR+** REORG While Open command is very easy to set up and run, and does not require applications to be quiesced, hence the keyword WHILE-OPEN. REORG While Open performs a complete reorganization of the catalog, including repair of any structural inconsistencies encountered during execution. Any definition attribute of the catalog can be changed, including allocation sizes.

REORG While Open significantly enhances data availability for you, as it allows you to perform maintenance and repair of your catalogs without an outage.

6. Disk Volser Renaming Projects

For various reasons, many installations have requirements to rename the volser on disk volumes.

If the volume doesn't currently have data on it, it's simply a matter of varying the volume offline, running ICKDSF and submitting a RELABEL command, then varying it back online.

If the volume does have data on it, the task is immediately more complicated, more error-prone, and more time consuming. Hence, this is a potential data availability conundrum. You have to first off-load all data on the volume to backup data sets, then perform the steps described in the above paragraph, then reallocate and reload the backup data sets to the volume. That doesn't sound like very much more processing, but if the volume has dozens or hundreds of data sets residing on it, this can be such a large task that it becomes impossible to do. The task is much worse if a catalog happens to reside on the volume to be renamed.

CR+ provides the SUPERCLIP command, whose express purpose is to rename (CLIP) a volume's volser – with the necessity of off-loading data from the volume – and then update all of the volser pointers in catalog records that point to every data set on the volume. To accomplish this, SUPERCLIP varies the volume offline, issues an ICKDSF RELABEL command 'under the covers', varies the volume back online. Then, tracing backwards to the appropriate catalog for each data set on the volume, it updates the volser pointer to where the data set resides.

SUPERCLIP is used in data center consolidations and merges, where volsers need to match SMS constructs. It is also used heavily in testing situations, where clones of volumes are created, and it's necessary to use different volsers than the original.

7. At DR, Fast Catalog Conditioning Is Crucial

Every installation has a DR plan tailored to that specific environment, including hardware and software configuration (mirrored DASD, or not, for example), recovery Service Level Agreement (SLA) for how soon you have to be back on the air with critical applications.

How your data, including your ICF catalogs, gets to the DR site is another variable. It could be mirrored, by full volume dumps, by application backups using an aggregate system such as ABARS – and most likely, a combination of these.

Once the operating system and this data is 'laid down', the next critical task is conditioning the ICF catalogs to ensure they match the actual data. If they don't match, huge problems will develop as soon as you begin running jobs. Processes for performing catalog conditioning have been around years, including convoluted homegrown REXX procedures that parse LISTCAT output, or processes using kludged-together functions within early vendor products. Incredibly, these procedures typically required from 5 to 15 hours of processing time – and this must be completed before any real work can begin at the DR site. Talk about lack of data availability!

CR+ provides two methodologies for catalog conditioning, an extremely powerful and fast command called CATSCRUB, and a 'selective recovery' facility within the RECOVER BCS command. Oftentimes, both methods are used by an installation, for different situations that make up the overall DR plan. Mainstar believes that you need a catalog conditioning facility to fit your current plan: you shouldn't have to modify your plan to fit the vendor product.

CATSCRUB is designed for the mirrored or full volume recovery scenario (although it can be used in other ways too), where the catalogs are already on the volumes with the data they catalog at the time catalog conditioning begins. With CATSCRUB, you 'point it' at one or more catalogs and the disk storage volumes you want the catalog to match, and in the blink of an eye it scrubs the catalog(s) to exactly match the volumes. CATSCRUB provides over a dozen global filtering values, to identify categories of catalog records that you wish the command to act upon that might be different from others. For example, you might want all data sets with MIGRAT in the volser to be deleted or kept, depending on how you're handling DFSMShsm data sets at the DR site.

With selective recovery, the assumptions are that you have catalogs on backup volumes upon arrival at the DR process, and that during restoration of the catalogs, you tell it specifically which catalog backup records you wish recovered and, therefore, which ones you don't want restored. Additionally, selective recovery can be used to define new, empty catalogs at the DR site (for use with ABARS, for example), and to establish empty GDG bases that will be populated during individual data set restores.

Herein lies the fundamental rub – selective recovery for restoration of catalogs requires you to specifically identify what's to be recovered, either by data set name, volume, type, etc., (or masking values on any of these). With CATSCRUB, you don't have to know – you just want the command to force-match the catalog to existing volumes.

Typically, CATSCRUB cuts catalog conditioning time from many hours down to a handful of minutes. From many benchmarks, elapsed time for this is commonly cut from 5 hours to 15 minutes, or from 15 hours to 30 minutes. This is truly a benefit to data availability.

Conclusion

In summary, the gathering, storing, and processing of your data are the only reasons you have an IT organization – and that means availability to the files containing that data. Without availability, applications cannot run and your business is at risk.

Just as the success of setting up and testing your DR plan is critical in the event of an installation-wide failure, day-to-day maintenance and support of your ICF catalog environment enhances local data availability.

Reliability, availability, and visibility (RAV) of your ICF catalog is therefore of paramount importance. If you're willing to spend a bit of time and resources to maintain a clean and workable ICF catalog environment, you can significantly

improve the RAV factor for your systems.

For information on improving ICF catalog reliability, read the previous article in this series, "Why Catalog RecoveryPlus? Part 1: ICF Catalog Reliability". In the next article in this series, we'll focus on ICF catalog visibility.

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

Ron Ferguson has a technical background in large-scale z/OS 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. Ron travels widely, meeting with customers and presenting at national and international conferences.