

An ICF Catalog White Paper

From Mainstar Software Corporation



Shared Master Catalogs - Are They Worth Doing?

By Marty Hasegawa

Preface: This White Paper is a general discussion regarding the implementation of ICF catalogs that serve as the System Master Catalog to multiple MVS system images. Most large data centers have already implemented Shared Master Catalogs for quite some time. This discussion is directed more toward those who might consider implementing, and to those who are unfamiliar and curious about Shared Master Catalogs. The Extended Catalog Facility is not included as part of this discussion.

Background

The ability to run multiple MVS images off a single master catalog has been available prior to OS/390™, but subsequent improvements to the operating system have made it much more feasible. Any MVS image can share a master catalog with another, as long as basic cross-system ENQ/RESERVE integrity is preserved. Sharing master catalogs is a suggested Parallel Sysplex™ implementation option. However, sharing master catalogs is neither a prerequisite to parallel sysplex, nor is parallel sysplex a requirement for sharing master catalogs. A master catalog can also service some, but not all, of the MVS images that share DASD. For the remaining discussion here, we'll use the generic term "MVS" to represent the base operating system component for any of IBM's MVS/ESA™, OS/390™, and z/OS™ products.

Who Benefits?

There are basically four distinct areas of responsibility that might benefit from a Shared Master Catalog implementation.

The most obvious benefits might be to the staff who are accountable for the hardware budgets. It

would seem reasonable to free up (N-1) DASD volumes for other usage, where "N" is the total number of MVS images. As we continue our discussion, we'll find this is not necessarily true.

Another area that might benefit is the staff who maintain the high-level-qualifier dataset name aliases, that provide ICF catalog search capability on dataset locate. These people are usually Storage Managers, but might also be Security Administrators, Systems Programmers, or even "all of the above". Having a single maintenance target for adding and deleting dataset high-level-qualifier aliases would theoretically eliminate the possibility of "out-of-synch" user catalog aliases. This, however, assumes that there will only be one shared master catalog. If more than one shared master catalog exists, then we are not necessarily better off than before. When shared master catalogs change in name and number from time to time, the single-point-of-maintenance theory falls apart completely, and the potential of out-of-synch user catalog aliases dramatically increases.

A third area of benefit's go to the Systems Programmers, who truly reap the benefits of sharing master catalogs. Releases of the operating system are typically migrated from the sysprog-testing image, to test images (quasi-production images), and finally to the actual production images. The shared master catalog can often be considered a part of the IPL volume set. Since datasets residing on the sysres set are typically cataloged in the master catalog, the master catalog must accurately reflect the dataset placements across the sysres volumes. This scenario simplifies the install and backout procedures for Operations, particularly when the new release of the operating system has redistributed the placement of the individual sysres datasets. It also simplifies the process of cloning new system images as the environment laterally expands.

The fourth area that can benefit from sharing master catalogs is Disaster Recovery, which typically includes Systems Programmers. Only one master catalog is needed to recover a multiple image production environment at the contingency site.

Setting up the Shared Master Catalog Environment

There are four key mechanisms available to support a Shared Master Catalog implementation. They are: System Symbolics, Symbolic volumes in nonVSAM catalog entries, Symbolicrelate Alias entries, and effective use of symbolics within SYS1.PARMLIB.

System symbolics are defined at IPL time, primarily through the IEASYMxx members of SYS1.PARMLIB. Symbolics that are inherently provided by the operating system include &SYSPLEX &SYSNAME and &YSR1. Through the use of the IEASYMxx members, additional customized symbolics can be created using CLIST-like assignment statements. The "Systems Programming Reference" manual provides further detail on this subject.

NonVSAM datasets cataloged on the sysres volumes can specify symbolic volume serial numbers using symbolic names such as &YSR1 or any other name generated via the IEASYMxx members. To allow devicetype independence, DEVICETYPES(X'00000000') can be specified to imply an esoteric unit name of SYSALLDA. Prior to the availability of system symbolics, a volume serial specification of 6 asterisks ('*****') implied the volume that the current system was IPL'ed from. This older technique is still supported, and is equivalent to &YSR1.

Symbolicrelate Aliases, introduced with DFSMS™ 1.5, are catalog alias entries that have pseudo-truenames containing an ampersand (&) somewhere within the related dataset name. Unlike other types of alias entries, the "true" nonVSAM catalog entry has no back-reference to the Symbolicrelate Alias entry. Symbolicrelate Aliases resolve to the truenam (or perhaps the "untrue" name) at the time the alias name is referenced. Symbolic substitution is dependent upon the set of system symbolics that are active on the current MVS image.

The SYS1.PARMLIB dataset is subject to system symbolic translation for nearly all

members. For instance, the IEASYSxx members can select other members based upon symbolics, and various members can specify particular dataset names that are translatable via symbolics. In this way, each system image can use the same SYS1.PARMLIB to access its unique set of non-shareable datasets such as LOGREC and DAE.

Given the availability of these support features, the final and most important ingredient is having a logical and consistent set of naming conventions for the system datasets and the sysres volume serial numbers. The IEASYMxx member should be able to easily derive the remaining volume serial numbers for all system IPL volume set based upon knowing the value of &YSR1. Non-sysres system datasets (such as the PAGE and SMF datasets) should allow for &SYSNAME substitution within the dataset name. PAGE datasets must be cataloged in the master catalog, whereas the SMF datasets (traditionally known as MANX and MANY) can be cataloged in a user catalog, provided that 'SYS1' is not used for a high-level qualifier.

System PAGE and SWAP datasets present interesting scenarios. Being cataloged to the master catalog, they must have unique names that associate their identity to a specific MVS image. Depending upon prior naming conventions, it might have been possible to crash another running system by IPL'ing an image with the incorrect master catalog. With shared master catalogs, it is now possible to crash another running system by IPL'ing an image using a duplicate SYSNAME.

As future releases and occasional maintenance levels of the operating system are built and migrated into the production environments, master catalogs can follow the sysres versions they support. The process of cloning new system images for additional images can become almost automatic through the use of shared master catalogs and shared sysres volume sets.

Disadvantages of Shared Master Catalogs

As is so often the case, changes to the environment can resolve one set of issues, only to replace them with an entirely new set of issues. We've already discussed the potential negative impacts made upon user catalog alias maintenance procedures. An additional set of issues arise due to the creation of a new "single-point-of-failure" exposure. Disaster Recovery is

simplified, but *local* recovery can become a bit more complex in cases where a failure causes the unavailability of the shared master catalog. Contingency strategies for such an event might include the use of more than one shared master catalog to service a split number of system images, or, providing a “spinning” standby alternate master catalog volume, or even both. We have now reclaimed at least a portion of the DASD that we had originally promised to free up.

Maintaining a “spinning” alternate master catalog (aka, a “hot” backup) introduces yet a couple more issues. The first issue is relatively minor. Complications can arise when the VVDS VVR records for the PAGE and SWAP datasets refer back to the original master catalog name instead of the alternate. Fortunately, the operating system doesn’t seem to care about this discrepancy for these specific datasets. However, IDCAMS DELETE will vehemently complain. Mainstar’s **Catalog RecoveryPlus** can handle this problem, with its ALTER BCS-BACK-POINTERS command, allowing you to change the VVDS pointers so they specify the alternate master catalog.

The second and more critical issue regarding the alternate master catalog is keeping the alias maintenance current. An out-of-date backup could lead to serious consequences, even in contingency recovery situations. If strict procedural discipline is not a viable option, some form of automated synchronization will be necessary. Again, **Catalog RecoveryPlus** can provide much-needed support in this area, with its DIAGNOSE ALIAS command that provides the capability to compare the alias environments on two master catalogs, creating fixes to synchronize one to the other.

Conclusion - Are Shared Master Catalogs Worth Doing?

Generally, sharing master catalogs does not make much sense if the data center only runs two or three MVS images that share DASD. Sharing master catalogs makes a lot of sense to Systems Programmers who tend data centers that run large numbers of system images. It will not reclaim a significant amount of DASD unless the number of system images is fairly high. (This is becoming a moot issue due to the plummeting prices of DASD). There has also been no reports of performance gain or loss. Storage Administrators exchange an old set of issues for a new set.

Sharing master catalogs can definitely benefit Systems Programmers and Disaster Recovery, positively affecting the efficiency of those related processes. In summary, shared master catalogs can benefit installations that run a large and variable number of MVS images. For all other situations, shared master catalogs are merely another way of doing things.

Marty Hasegawa - is a Software Developer for Mainstar Software Corporation.

Marty Hasegawa had been a Systems Programmer for large-scale MVS systems for 23 years prior to joining Mainstar in 2000. His highly regarded expertise includes a wide range and depth of technical knowledge in operating system components, applications platforms, information security, and the constant dilemma of making technology work for people.

The following terms are trademarks of the IBM Corporation in the United States and other countries:

DFSMS™
MVS™
MVS/ESA™
OS/390™
Parallel Sysplex™
z/OS™