



Ask Mr. Catalog

Answers to common ICF catalog questions

BY STEPHEN M. BRANCH

This article continues my recurring series from *z/OS Hot Topics Newsletter Issue 21*, August 2009, GA22-7501-17. I hope you find these common ICF questions and answers useful. If you have a question, please send it to me at the following e-mail address: branchs@us.ibm.com

1. What are the benefits of VSAM striped data sets?

VSAM striped data sets have consecutive control intervals spread across multiple volumes. This format allows a single application request for records on multiple control intervals to be satisfied by concurrent I/O requests to multiple volumes. The expected result is often improved performance for sequential data access by achieving data transfer into the application at a rate greater than any single I/O.

You must allocate VSAM striped data sets as extended format data sets. VSAM supports all organizations (KSDS, ESDS, RRDS, VRRDS, and LDS) and supports up to 16 stripes although increased performance results beyond five or six stripes are marginal because of programming overhead. Random access performance is unaffected. z/OS does not support VSAM data sets for striped VSAM record level sharing (RLS).

How big should an ICF catalog be?

Base the size of your catalogs on application and recovery needs. The larger a catalog, the more time it takes to recover. Consider the following about your applications.

- How many entries do they require?
- What is the length of time necessary to recover a catalog?
- How long can you tolerate an outage for the application?

That said, these days catalogs are very resilient and DASD hardware is much more reliable. You can currently have approximately 3500 aliases per catalog. This might also be a factor in your catalog's size.

The current catalog allocation maximum size is 4 GB. This might also pose a problem as the sizes of DASD volumes grow and the numbers of data sets

increase. IBM is aware that many customers would like to see an increase in this limit.

2. What is the difference between the in-storage cache (ISC) and the catalog data space cache (CDSC)?

By default, catalog management uses the in-storage cache method to cache catalog records in the catalog address space (CAS) private storage. Each catalog has its own cache. When systems share a catalog, every time a shared system updates the catalog the ISC is invalidated.

The catalog data space cache (CDSC) caches catalog records in a data space. For CDSC, you must provide a CLASS statement in the COFVLFxx member of SYS1.PARMLIB to identify eligible catalogs. When sharing a catalog between systems, the sharing information is passed using the catalog self-describing VSAM Volume Record (VVR). The information is used to invalidate only those records that change. If the system makes more updates to a catalog than can be passed in the shared information, the entire CDSC for that catalog is invalidated. For more information, see the topic on “Defining the catalog data space cache” in *z/OS DFSMS Managing Catalogs*, SC26-7409.

In short, ISC uses main memory to cache catalog records and invalidates the entire cache when a shared system updates any record. CDSC uses a data space to cache catalog records. Invalidation happens to only those records changed by another system when the shared system receives sufficient information. CDSC is a better caching choice in most instances because it has better granularity for invalidation.

3. How often should I reorganize my user catalogs?

You only need to reorganize a catalog when it's approaching the maximum extent limit of 123. Catalog management provides a warning when 80% of the extents are used. You can choose to adjust the warning limit for your installation

Many people think there's a performance gain by reorganizing based on the number of CI or CA splits. This is a myth---don't let this be criteria for catalog reorganization.

The best practice is to monitor the number of extents in your catalogs and take action before you near the extent limit. If your catalog volume is nearly full, extra extents might not always be available.

Stephen M. Branch is an IBM Senior Software Engineer at Silicon Valley Lab whose 25 year career has dealt with all aspects of DFSMS. He is the author of the Catalog Search Interface (CSI). Stephen holds a Master of Science Degree from Louisiana State University.