Waiting to Expire: Dataset Aging

BY STEVE PRYOR

Like the milk carton in your refrigerator, data has a limited shelf life. And just as with the contents of the refrigerator, it's important to know when the data was last used and when it should be discarded. Storage administrators use data management programs such as DFSMShsm, FDRABR, and others to examine the dates associated with a dataset. These tools enable them to determine if the dataset should be moved to less expensive media if it has not been accessed recently, or deleted if it has expired. Aggressively managing data in this way ensures that expensive DASD space is used by the active production datasets that need it most.

In MVS systems, date information is maintained in several places. To ensure that data management programs properly migrate and delete files based on these dates, the storage administrator must understand the formats in which dates are kept and which locations take precedence. In this month's column, I will examine expiration dates and last-reference dates to see how they are stored and how they are used to keep data in its correct place in the storage hierarchy.

DETERMINING THE EXPIRATION DATE

The format-1 DSCB describes the characteristics and location of a dataset on a volume. While most DASD datasets do not have an explicit expiration date, it is also not uncommon for jobs to specify EXPDT or RETPD in the JCL that allocates a dataset. In this case, the specified expiration date is placed in the DS1EXPDT field of the format-1 DSCB. For SMS-managed datasets, the specified expiration date is also placed in the dataset's catalog entry. Non-SMS managed datasets do not normally have an expiration date in the catalog

entry, although one can be placed there by the IDCAMS "ALTER" command.

For VSAM datasets, the expiration dates placed in the format-1 DSCB and catalog are slightly different. Non SMS-managed VSAM datasets without an explicit expiration date receive "99365" instead of zeroes in the data component and index component format-1 DSCBs. If an explicit expiration date is specified via the "FOR" or "TO" parameter of the IDCAMS DEFINE statement, this date is placed in the catalog entries for the cluster and the data and index components, but the expiration date in the format-1 DSCB remains "99365." For SMS-managed VSAM datasets, the expiration date is also placed in the catalog, but the DS1EXPDT field in the format-1 DSCBs is always zero rather than "99365."

Datasets with an explicit expiration date cannot be deleted unless the "PURGE" parameter is specified during deletion, and thus are protected against inadvertent scratching. Normally, this is the intent of a user who specifies an expiration date. Occasionally, however, a dataset may wind up with an unwanted expiration date. This might occur, for example, when the ACS routines are used in Tape Mount Management implementations to redirect small tape datasets, typically GDGs, to DASD. JCL for these datasets often includes an expiration date of "99000," which means "retain until uncataloged" to the tape management system, but that is treated as a real expiration date (expire first day of 1999) for a DASD dataset.

SETTING BOUNDARIES FOR EXPIRATION DATE VALUES

To prevent this sort of error, the SMS management class construct includes a field called RETENTION LIMIT. This field is used to set a boundary for the expiration

date values that can be specified for a dataset. For example, if the JCL that allocates a dataset specifies LABEL=RETPD=60 and the dataset is assigned to a management class for which the RETENTION LIMIT is 30, then the dataset will expire 30 days from the current date, since the retention limit value overrides the explicitly specified value. Setting the retention limit to zero indicates that all expiration dates and retention periods coded in the JCL, IDCAMS control statements, or data class are to be ignored. The default is RETENTION LIMIT=NOLIMIT, meaning that all expiration dates are honored.

DFSMShsm daily space management processing and FDRABR superscratch processing provide the storage administrator with a means to delete old datasets whether or not they have an explicit expiration date. For SMS-managed datasets, management class fields provide criteria for choosing datasets that can be scratched based upon the number of days since they were last referenced (EXPIRE AFTER DAYS NON-USAGE) or upon a particular date or retention period (EXPIRE AFTER DATE/DAYS). Regardless of the value of these fields, however, a dataset with an explicit expiration date in its format-1 DSCB or catalog entry, SMS-managed or not, will not be scratched unless the specified date has passed. Furthermore, for DFSMShsm, even datasets that meet all the criteria for deletion will not be removed unless they also meet an "integrity age" requirement, which may be zero, one, or two days, depending upon the number of DFSMShsm hosts and the setting of serialization parameters.

Occasionally, an invalid date may creep into a format-1 DSCB. The date may then be propagated the dataset's MCC (back-up) record, MCD (migration) record, or

elsewhere, causing errors when attempts are made to restore the dataset. To correct an invalid date, the storage administrator needs to know how dates are stored in the creation date, expiration date, and last-referenced date fields in the DSCB. Each of these three fields (DS1CREDT, DS1EXPDT, and DS1REFD) occupies three bytes and is held in "yyddd" format. To accommodate four-digit years that may extend beyond the year 2000, the "yy" is a hexadecimal representation of three decimal digits that designate the century and year. A century number of "0"indicates the 20th century, while a century number of "1" indicates the 21st century, and "2," the 22nd century. Thus if the expiration date is set to X'630003', the dataset will expire on January 3, 1999 (X'63'= decimal 99). It follows then that if the expiration date is set to X'C60003', the dataset will expire on January 3,2099 (X'C6'= decimal 199). The largest year that can be specified for any type of date is 2155, since X'FF' is the largest value that can be held in the "yy" byte, and is equal to decimal 255, or the 55th year of the 22nd century. Fortunately, it will be some time before most storage administrators will need a larger date value than this.

Unlike the expiration date, the last-referenced date for a dataset is maintained only in the format-1 DSCB and not the catalog. It is set whenever a dataset is opened, and is used by the data management utilities to determine whether the dataset has remained unused for a long enough period to warrant migration to cheaper media or deletion. In the past, problems have arisen for datasets such as online databases that are opened once and then may remain open for days or weeks before being closed and reopened again. Since the last-referenced date reflects the time the dataset was opened, and not the time it was actually last used, the dataset might be swept off DASD even though only a few minutes had elapsed since it was closed. APAR OW16706, introduced in DFSMS version 1.2, allows the last-referenced date and the DS1DSCHA (dataset-changed bit) for SMS-managed non-VSAM datasets to be updated at CLOSE time as well as when the dataset is opened. In DFSMS version 1.4, this support was also extended to VSAM datasets.

There are many other subtleties the storage administrator must consider when looking at how the dates associated with datasets are maintained. These include setting the creation date when files are copied, moved, or restored by utilities such as DFSMSdss, FDRDSF, or IDCAMS (it may be left alone, or it may be set to the current date), and the usefulness of the last-reference date for VSAM files (it is maintained only in the data component of the base cluster). A thorough understanding of date processing in DFSMS will aid the storage administrator in bringing the data center efficiently into the next millennium. **ts**

NaSPA member Steve Pryor has more than 15 years of experience in storage management, disaster recovery, software development, and technical support. Steve can be contacted via the Internet at pryor@atlanta.com.