How Do Things Stack Up? DFSMShsm Dump Tape Stacking

BY STEVE PRYOR

One of the unmistakable features of modern data processing is the rapidly increasing capacity of storage devices. Only a few years ago, data centers with more than a terabyte of mainframe DASD storage were rare. Today, installations of 10TB, 20TB or even larger are not uncommon. In these large-scale environments, storage administrators depend upon large-capacity tape storage hardware, such as STK tape silos and IBM 3590 Magstar tape drives, to ensure that all of the critical DASD data is regularly backed up. Since the survival of the corporation can depend upon the recoverability of data, efficient strategies are critical for backup and restore of these large volumes of information.

For storing very large amounts of data in a small space, tape is usually the medium of choice. The vast capacity of this media (a single Magstar cartridge can hold anywhere from 10GB to 30GB of data) makes it ideal for storage management functions such as Tape Mount Management or unreferenced dataset migration, where datasets that are no longer needed on DASD are "swept" off to tape in bulk.

One of the keys to the successful use of tape is the ability to fill each tape cartridge to capacity. It is extremely wasteful to put only a few bytes of data on a cartridge with a capacity that can be measured in gigabytes, and more importantly, each cartridge may take up a valuable slot in a robotic tape silo. Data center backups are one of the major consumers of tape space. The storage administrator who manages the backups, therefore, should look for every opportunity to improve the level of backup tape utilization.

IMPROVING TAPE UTILIZATION

One place in which improvements can be made is in DFSMShsm automatic dump processing. While the migration and incremental backup functions of DFSMShsm have been able to fully utilize tapes, dump processing, which invokes DFSMSdss to perform full-volume backups, has been forced to devote an entire tape to each DASD volume dumped. Even if completely full, most DASD volumes will not nearly fill a backup tape, and in many installations, DASD utilization is often restricted in order to ensure that sufficient disk space is available for production needs. The result can be a large number of dump tapes with most of their capacity wasted.

Fortunately, beginning with DFSMS releases 1.3 and 1.4 a new capability is provided to stack multiple DFSMShsm dumps to a single cartridge. Stacking multiple dumps on the same tape significantly decreases the number of tapes required and therefore the number of tape mounts performed. The number of dump tasks required can also be decreased, with fewer tape drives needed for DFSMShsm automatic dump processing since the disk volumes belonging to each dump class are dumped sequentially.

AUTODUMP stacking is implemented via 3 APARS, OW27973, OW29535, and OW29537. These APARS implement the new STACK(nn) parameter on the DFSMShsm DEFINE DUMPCLASS command. The value in the STACK parameter indicates how many DASD volumes may be stacked on a single output tape. The default, of course, is to place only one volume on a single tape. Given average values for 3390 occupancy, Magstar tape capacity, and compression ratios, IBM recommends that the value for STACKing be between 10 and 12 to minimize the number of tape volumes used. Each dump class may have its own value for the amount of stacking to be performed.

Most DFSMShsm parameters that apply to dump volumes containing only a single backup dataset apply in exactly the same manner when backups of multiple disk volumes are stacked on the dump tapes. Each DUMP operation, for example, always starts with a new dump tape. Stacking does not continue from one dump operation to the next but ends at the completion of each DUMP. Similarly, any retention period or expiration date associated with the dump class applies to all dumps, whether stacked or not. This means that all of the dumps that are stacked on a tape will expire on the same day. If VTOC copies are to be created for dumps in a dump class, they are created for all of the dumps stacked on a tape.

With the STACK parameter specified, the order in which DFSMShsm selects disk volumes for dump processing changes somewhat. Volumes eligible for stacking are dumped first before volumes that are ineligible for stacking, such as those which are currently being processed by DFSMShsm migration. Each stacked dump tape may contain both SMS-managed and non-SMS data, but will not mix data from disk volumes that have affinity to the DFSMShsm host with those that do not.

The implementation of DFSMShsm automatic dump stacking involves changes to several DFSMShsm commands, usually in order to identify which dump (i.e., which file sequence number) is of interest. For example, the LIST DUMPCONTENTS command, used to display information about the datasets on the disk volume at the time the dump was taken, requires the source disk volume serial number be specified in order to indicate which stacked dump to use. The RECOVER command also has a new parameter, SOURCEVOLUME, to identify which of the stacked dumps to use when recovering a dataset that has become uncatalogued.

As one might expect, there are a number of considerations that the storage administrator must take into account before implementing dump tape stacking. For instance, the stacking of dumps applies only to the automatic dump function, not to dumps taken with the BACKVOL DUMP command. When deleting dump volumes, the storage administrator must be aware that the DELVOL command will fail for stacked dump volumes, unless the PURGE parameter is specified. Changes have been made to several of the DFSMShsm record types and control blocks (the DCL, DGN, DVL, and the MWE), so any installation-written or vendor programs that use this data may need to be updated.

Implementing automatic dump stacking can reduce the number of dump tapes required However, more time might be required for all of the dumps to complete if the number of concurrent dump tasks is reduced, since the volumes within a dump class will be dumped sequentially rather than concurrently. Another potential source of surprise occurs if the last dump file on a tape extends to a second volume, something that was unlikely to have occurred without stacking. In this case, DFSMShsm will issue message ARC0654I STACK nn for DUMP CLASS WENT to mm VOLUMES.

In addition, if the maximum number of dump generations is exceeded, DFSMShsm may roll off some dump copies as new ones are created. This will leave some dump tapes partially used, containing both good (current) dumps and bad (rolled-off) dumps.

Compatibility APARS (OW28855 and 28857) provide support for the stacked volumes back as far as DFHSM version 2.6. Before implementing stacking of DFSMShsm dump tapes, it is important to ensure that the stacking support is applied to all systems (including any systems at potential disaster recovery sites). If stacking support is not installed on these systems, they will not be aware of the stacked dumps, which could then be inadvertently deleted.

Another consideration is the amount of time required to restore a dataset from a tape containing multiple stacked dumps. Since a single tape may hold dumps from many volumes, it may take somewhat longer to position to the beginning of the dump dataset. To minimize this problem, DFSMShsm stores the file sequence number and block id of the first block of the dump in the BCDS so that the high-speed block search feature can be used to quickly position the tape.

In today's fast-growing data center, the storage administrator must always look for new ways to manage the flood of information that makes businesses run. Dump tape stacking is another facility that the storage administrator can take advantage of to ensure that the available storage hardware is used to its full capacity and that backup and recovery procedures are as efficient and effective as possible. 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.