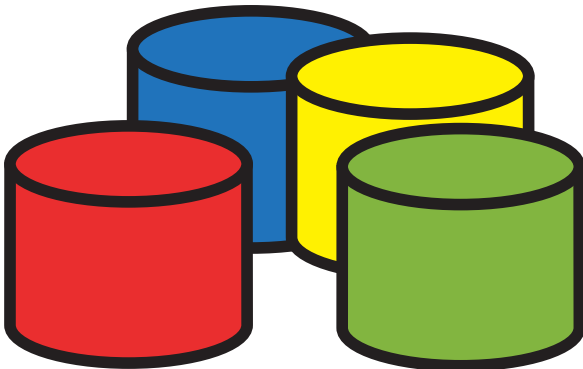# P V S
dynamic Disk Management System

PVS is a dynamic disk management system for the operating system VSE resp. z/VSE with or without VM. It offers assistance for the following functions:

- dynamic assignment of free dasd space

- dynamc adminstration of dasd files in a catalog

- dasd sharing among any number of virtual or real machines
- fast backup/restore procedures for user data

Present situation:

due to the availability of large disk capacities it is reasonable to write sequential datasets intermediately to dasd. This allows also, to hold multiple generations of a dataset.

Under the operating systems VSE resp. z/VSE this is not possible, as

1) sequential disk datasets need an EXTENT statement with an exact definition of start track and size. Besides, an ASSGN statement must be provided to the appropriate dasd.

2) if multiple generations of a dataset are to be hold, the label in the DLBL statement must be adjsted accordingly.

A solution for these problems is offered by

P V S

the dynamic disk management system.

Design and mode of operation

The user defines, which dasds or areas of a dasd are to be controlled by PVS. These areas are called pools. Any number of pools with any number of dasd areas can be defined. These pools may be shared by one or multiple real or virtual machines.

PVS owns a catalog in which all controlled sequential dasd files, together with expiration date and generation number, are stored.

PVS can be initialized in a partition during IPL of VSE. Afterwards this partition is free and available. PVS uses the possibility of vendor exits in VSE. Thus PVS gets control whenever a dasd open, close, end of extent or end of task occurs.

Output of sequential dasd files

A sequential output file is managed by PVS if either no EXTENT statement or an EXTENT statement with relative start track 1 is supplied.

PVS needs a pool name for the assignment of a dasd area. Optionally, a poolname may be defined either globally or explicitly by defining the poolname as VOLSER in the EXTENT statement.

PVS administrates the allocation of the pools by itself. So it can complete or add the appropriate extent information in the EXTENT statement. Thus, any manual change of jobcontrol is avoided.

Optionally, a file may be created as so called single file. This means, there is always only one single output of this file. On the other side, there may be multiple generations of a dataset (under z/OS these are called generation-data-group / GDG).

In the case of output of a GDG, PVS completes the label of 44 characters in the last 4 to 6 positions by a generation number. The form is "G#' followed by a 2 to 4 digit number (01 to 9999). Each later output of such a dataset increments the generation number by one. Each file holds its generation number as long as

iit exists. The highest, possible number is 9999.

Each dasd file written to a pool is managed by PVS without any preceeding definitions.

Optionally additional definitions can be set either for a dataset or system wide. Those definitions may be the size of the very first extent (primary extent), the size and the maximum number of the sequence extents. Furthermore a retention period and a number of generations to be hold may be defined.

At close PVS releases the unused dasd space, so that it is available for other files.

When normal "end of task" is reached, all output files created in this task are permanently recorded into the PVS catalog. In case of an abnormal end of jobstep, all files, even those which were already closed, are deleted and the used dasd space is released.

Input of sequential dasd files

While opening a sequential dasd file, PVS always gets control.

PVS checks, if the input file defined only by a DLBL statement is controlled by its catalog. If yes, PVS completes the DLBL statement by Extent Statements for all extents and creates, if necessay, an assignment to the appropriate dasd(s).

For input files with with multiple generations (GDG) PVS usually forces the latest generation to be opened. If an effective (nnnn) or relative generation number (latest generation - n) is specified in the file-id field of the DLBL statement the appropriate file is opened.

Backup of user data

PVS offers a fast and flexible tool to backup the user data stored in one or more pools. Backup media is always a tape. Due to the high capacity of tapes (30 GB an more) a restore of an even small file may take quite a while due to the fact, that the tape must be read to find the data.

To solve this problem, PVS creates an index file while backing up which holds the physical position of all dasd files on tape. This file may be created as dasd or tape file, even a sequence tape file, or both.

In the case of a restore of a single dasd file, PVS can determine the position of this file on tape by its index file and can position the tape quickly to the appropriate location. Then, the restore of the user data begins

Advantages of PVS

- easy and flexible administration of sequential dasd datasets, as no EXTENT statements must be used.
- any number of generation of a dataset can be hold.
- unused dasd space is released at close.
- datasets need not to be defined in advance
- fast Restore of PVS controlled datsets due to indexed tape positioning of the backup tape.

Miscellanous

As PVS is a new product development all hard- and software standards could be respected.

Therefore PVS does not support ISAM and Direct Access files, as these have been overtaken by the VSE operating system as VSAM files.

As dasd units nowadays usually have an own cache, PVS only works with the regular VTOC without an index. Thus, the overhead in cpu and i/os is avoided.

Conversion of other Dasd Management Systems

PVS has a conversion program to convert an existing Katalog of other dasd management systems into a PVS catalog.