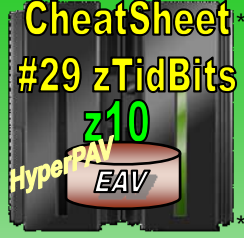

* Past architectural limit of disk volumes was 54 GB per volume, *now* Extended Address Volume (**EAV**) is planned to initially support 223 GB per volume on z/OS V1.10 and IBM System Storage DS8000 (262,668 cylinders per volume).
  - There is full compatible access to data residing on cylinders below 65,520.
  - The existing 3390 device geometry[1] (track length and number of tracks per cylinder) is maintained on EAV (ESDS, KSDS, RRDS and LDS), at any location on an extended address volume.
  - Non-VSAM data sets, catalogs, page data sets, and VSAM data sets with the KEYRANGE or IMBED attribute are restricted to the first 65,520 cylinders.
  - With this initial support, space after the first 65,520 cylinders is intended to provide constraint relief for applications using large VSAM data sets, such as those used by DB2, CICS, zFS file systems, SMP/E CSI data sets, and NFS mounted data sets.
  - In z/OS V1.10, support is planned for non-SMS managed VSAM data sets as well.
* There is a *new* dynamic volume expansion function designed to eliminate the need to copy volumes to increase their size.
  - This new function is expected to provide substantial, immediate constraint relief for installations with a hugh number of large VSAM data sets.
  - This is also expected to help improve storage management administration over time, as a relatively small number of large volumes are thought to be simpler to manage than a larger number of smaller ones.
* IBM recommends the **HyperPAV**[2] on the IBM System Storage be leveraged to help manage the number of paths to devices defined as EAV.
  - With HyperPAV technology, z/OS uses pools of aliases by Logical Subsystem[3] (LSS).
  - As each application I/O is requested, if the base volume is busy with another I/O, z/OS selects (and removes) a free alias from the pool, and starts the I/O to the base address (UCB) through the selected alias. When the I/O completes, the alias device is used for another I/O on the LSS or is returned to the free alias pool. If *too many* I/Os are started simultaneously, z/OS will queue the I/Os at the LSS level and not the device level.

NOTE1: The 3390 geometry contains 56,664 bytes per track, 15 tracks per cylinder and 849,960 bytes per cylinder. The terms 'track' and 'cylinder' comes from old pre-raid disks, which were like 8 old fashioned vinyl records stacked in a pile, with a set of fixed read/write heads which moved in and out of them. The disks had recording surfaces on both sides of the 16 surfaces, one surface was used for control information which left 15 for data.

NOTE2: The DS8000 series offers enhancements to Parallel Access Volumes (PAV) with support for **HyperPAV**, which is designed to enable applications to achieve equal or better performance than PAV alone using the same or fewer operating system resources. ***See #18 zTidBits for PAV overview.***

NOTE3: A logical subsystem (LSS) is a logical structure internal to Enterprise Storage System that consists of up to 256 addresses to be used for base volumes (*UCB*s) and aliases.
- An alias can only be associated with a base within the same LSS.
- ESS supports up to 16 logical subsystems for System z. Bases and aliases are managed within the scope of each LSS. Thus, the administrator designates a certain number of devices based on the amount of data that needs to be held by the LSS.

☞ ***UCB** = Unit Control Block represents a device to the LSS.*

* An important distinction using dynamic[4] PAV is that it requires the WLM to monitor the workload and goals. It took some time until the WLM detected an I/O bottleneck and then the WLM had to coordinate the reassignment of alias addresses within the sysplex and the DS8000. All of this incurred latency and if the workload was fluctuating or had a burst demand, the job that caused the overload of one volume could have ended before the WLM had reacted. In these cases, the IOSQ time was *not* eliminated completely.
* With **HyperPAV**, the WLM is *no longer involved* in managing alias addresses.
  - For each I/O, an alias address can be picked from a pool of alias addresses within the same disk control unit (DS8000).

HyperPAV has been designed to:
- Provide an even more efficient Parallel Access Volumes (PAV) function
- Help customers, who implement larger volumes, to scale I/O rates without the need for additional PAV alias definitions
- Exploit FICON architecture to reduce overhead, improve addressing efficiencies, and provide storage capacity and performance improvements:
  > More dynamic assignment of PAV aliases improves efficiency.
  > Number of PAV aliases needed is reduced, taking fewer from the 64 K device limitation and leaving more storage for capacity use.
- Enable a more dynamic response to changing workloads
- Simplified management of aliases

Attribute	PAV	HyperPAV
Alias to Base ratio	Complex: One alias per 9 GB depending on work load and I/O rates	Simple: (peak I/O rate * avg respone time *2) = 10 x reduction in the # of PAV-aliases.
Workload Management	Sluggish: WLM adjusts every 10 seconds when work is not meeting goals and adjustment can be late missing alias change.. This incurs multi-system (sysplex) overhead	On Demand: Instantaneous response to changing workloads with no multi-system (sysplex) overhead
Improved Efficiency	Alias used for specific base: PAV-Aliases bound to the same base across all OS images.	Alias used for any base, any time on any system; Each OS image uses PAV-alias for a different base at the same time – uses multiplier for the effective number of aliases.
Virtual Storage Constraint Relief	PAV-aliases reside in in 31 bit storage	10x reduction in PAV-aliases and device related data structures.

NOTE4: Static PAV: Aliases are always associated with same base addresses. Dynamic PAV: Aliases are assigned up front, but can be reassigned to any base address as need dictates by means of the Dynamic Alias Assignment function of Workload Manager - reactive alias assignment.