

EVMS and Linux LVM: A Comparison

July 12, 2002

	EVMS	LVM	LVM2 w/ Device-Mapper
Overall Design	A complete, comprehensive volume management system. Modular, plug-in architecture. Supports new volume types and formats by simply adding a new module to the existing infrastructure.	Single-purpose architecture. Supports one style of volume management and one data format.	Modular design. Currently supports LVM format.
	Manages the entire volume stack, from the disk drives up to the filesystems.	Manages a single facet of volume management. Relies on other Linux sub-systems in order to build and manage more complex volumes.	Currently supports LVM volumes.
Supported Features	Disk partitioning <ul style="list-style-type: none"> - DOS-style MBR/EBR - GPT (for IA-64) - CDL/LDL/CMS (for S/390), including fdasd and dasd_format support 	None <ul style="list-style-type: none"> - Must use separate tools for disk partitioning (fdisk, parted). 	None <ul style="list-style-type: none"> - Must use separate tools for disk partitioning (fdisk, parted). - New user tools would be required to allow device-mapper to support disk partitions.
	Volume groups <ul style="list-style-type: none"> - Linux LVM - AIX LVM - OS/2 LVM 	Volume groups <ul style="list-style-type: none"> - Linux LVM 	Volume groups <ul style="list-style-type: none"> - Linux LVM
	Software RAID <ul style="list-style-type: none"> - Linear - RAID-0 (striping) - RAID-1 (mirroring) - RAID-4/5 (striping w/ parity) 	None <ul style="list-style-type: none"> - Linux LVM supports LVs striped across PVs (similar to RAID-0). 	None <ul style="list-style-type: none"> - New user tools would be required to allow device-mapper to support Linear and RAID-0. - New device-mapper modules would be required to support other RAID levels.
	Bad-block-relocation	None	None <ul style="list-style-type: none"> - New device-mapper module would be required to support BBR.

	<p>Snapshotting</p> <ul style="list-style-type: none"> – between any EVMS objects – fast–mode (asynchronous copies) – writeable – rollback (revert the original back to the state of the saved snapshot) 	<p>Snapshotting</p> <ul style="list-style-type: none"> – only between LVM LVs – read–only 	<p>Snapshotting</p> <ul style="list-style-type: none"> – fast–mode – writeable
Filesystem Integration	FSIMs allow easy, integrated filesystem management directly from the EVMS UIs.	Must use separate utilities for managing filesystems (mkfs, fsck).	Must use separate utilities for managing filesystems (mkfs, fsck).
	Expanding and shrinking volumes is coordinated with the filesystems. Ordering is handled automatically.	Expands and shrinks of volumes is separate from expands and shrinks of the filesystems. User must understand ordering issues or risk losing filesystem data.	Expands and shrinks of volumes is separate from expands and shrinks of the filesystems. User must understand ordering issues or risk losing filesystem data.
	<p>Current support:</p> <ul style="list-style-type: none"> – Ext 2 / 3 – JFS – ReiserFS – Swap – XFS (coming soon) 	<p>None</p> <ul style="list-style-type: none"> – Must use separate utilities for managing filesystems 	<p>None</p> <ul style="list-style-type: none"> – Must use separate utilities for managing filesystems
Usability	Several user interfaces available. Easy–to–use graphical interface for use in a windowing environment. Full–screen text–mode interface for console access. Scriptable command–line for automation. Other special–purpose utilities also available.	One set of command–line utilities. Each management task is handled by a different command.	Multiple command–line utilities have been combined into a single command–line for LVM2.
	User–space engine provides an API that allows any number of additional user–interfaces to be written. This API could be used to write an interface specific to a Linux distribution’s installation program.	Includes a library of commonly used routines that are shared among the command–line utilities. Difficult to use from 3rd–party applications.	
	Management of the entire volume stack allows complex volumes to be built and managed intuitively in a single user–interface. Plug–in ordering and interaction issues are handled automatically.	Management of complex volumes requires using a different set of tools for each layer. Presents complex ordering and interaction issues that the user must understand to build the volume correctly.	Management of complex volumes requires using a different set of tools for each layer. Presents complex ordering and interaction issues that the user must understand to build the volume correctly.

Migration	Plug-ins can be written to recognize volume formats from other operating systems. Provides simple migration path from other OS to Linux without any loss of data or cumbersome backup/restore migration. (Requires filesystem support as well.)	Single volume format. No migration options currently.	No migration options currently. Migration would require new device-mapper modules and new user tools.
	Currently offers plug-ins to support AIX and OS/2 volume formats	Not possible.	None currently.
	Could offer support for Sun/Solaris, HP-UX, and Windows volumes if the details of those formats are available.	Not possible.	None currently.
Scalability	Full 64-bit I/O paths. When 64-bit I/O is enabled in the 2.5 kernels, volume size limit will increase from 2 terabytes to 8 million petabytes (8 zetabytes).	Currently supports 32-bit I/O.	Currently supports 32-bit I/O.
Maturity	Production-ready. Continuing to support and enhance.	Production-ready. Support will be completely dropped for 2.5 kernels.	Beta-level. For testing only.
Kernel	2.4 and 2.5 support.	2.4 support only.	2.4 support currently. 2.5 support planned.