Understanding Software-Defined Storage

How it works and why
its time has come

April 2017

In a nutshell

The term "software-defined storage" (SDS) is all over the press, yet sometimes it seems there are as many definitions of it as there are companies promoting it. Once you dig in, however, it becomes clear that it is more than just hype: these companies are trying to solve real problems – problems such as soaring storage management costs, vendor lock-in, proliferating storage silos, keeping up with new technologies, and making disparate storage products work together.

This introductory guide will pull it all together for you. It looks at the reality behind the headlines to draw a map that you can use before you embark on what could otherwise be a journey into uncharted territory.

What is software-defined storage?

The core concept within software-defined storage – as with all the other 'software-defined somethings' – is the physical and logical separation of the hardware and software. It achieves this by moving the command and control element of storage away from the hardware and into a software-based service management interface or layer. Here, routine storage management can be automated for policy-based provisioning and administration, and for end user self-service.

That software will typically be running in a separate server and may also control other available storage on the network or in the data centre. The essential element here is that the storage goes into a shared pool, from which the SDS controller can flexibly carve out storage for applications and services. The high level software control makes it relatively easy to modify things when necessary, such as when more capacity is required by an application or service. It can even happen automatically based on policy triggers. And this storage pooling makes more efficient use of the available capacity.

In essence then, SDS takes all the storage available to it and provides the ability to manage and use it as if it were a simple array, even though that underlying physical storage might be composed of many physically dispersed and disparate elements.

The SDS differentiator

The fact that SDS physically and logically separates the storage hardware and software is vital – yet is also potentially very confusing. This is because in some respects all storage is software-defined and has been ever since the development of the IDE (later ATA) disk drive interface by Western Digital and others in the early 1980s. The IDE controller presented the host with a set of generic 512-byte blocks. The controller then took care of the mechanics of accessing the physical disk, even transparently mapping-out bad sectors without the host computer being aware that anything was missing.

The difference with SDS is that this software-based definition is not fixed in firmware. Instead it can be re-programmed via the management layer, for example to grow or shrink a live storage volume, or to change its service level or data protection characteristics, perhaps even to move the data to a different type of storage to meet the new service requirements as they change.

It seems there are as many definitions of software-defined storage as there are companies promoting it; however, it is more than just hype.

SDS takes all the disparate storage available to it and provides the ability to use it as if it were a simple array.

The anatomy of storage

An enterprise disk array and a microSD card look utterly different and offer very different capabilities, but like all storage systems, they have the same three elements inside:

All storage systems contain the same three basic elements, but with SDS you don't have to buy them as a single, fixed bundle.

- Media, whether this is disk, tape, flash or any combination thereof;
- Controller(s) – the processor and other hardware that drives the media;
- Software, which runs on the controller to glue it all together, add storage services such as read, write and even more sophisticated ones such as data protection, and present the resulting storage to the outside world.

In many cases – most enterprise arrays, for instance – these three elements will arrive as a complete pre-integrated bundle, but this doesn't have to be the case. Today, with SDS it is perfectly possible to disaggregate the process, instead buying storage media, connecting it to an industry-standard server, and then loading the latter with appropriate SDS software.

This disaggregation is almost exactly what many smaller storage vendors do, especially those promoting hyper-converged storage as an alternative to the traditional array. End-users can do it too, although it requires a degree of detailed technical knowledge that is less common outside areas such as academia, R&D, and skilled IT storage specialists.

Today it's perfectly possible to build your own SDS – just buy storage media, connect it to an x86 server and load the latter with SDS software.

The software within modern storage arrays is complex. As well as managing the storage hardware and provisioning storage volumes to users and applications, it will typically implement a degree of storage abstraction or virtualisation. This is how it merges multiple drives into a single, large, logical device that can then be carved into user-volumes, and it is what separates how the data is physically laid out on the media from how it logically appears to a file system or server. It is also what enables arrays to provide features such as virtualised file, block and object interfaces, thin provisioning, RAID and snapshots.

Depending on the implementation, SDS will be able to do some of that, maybe all of it, and possibly a lot more. That could mean scaling the entire storage infrastructure seamlessly and without disruption, say, or transparent remote replication and encrypting data at rest, and so on. And of course in many cases SDS can do this on disparate commodity-grade hardware.

Disaggregation – why now?

The exponential growth in data volumes seen in recent years has rendered the existing ways of managing storage systems unsustainable. As the joke goes, you may love running your first two or three arrays or filers, but by the time you install your 4th or 5th, storage management has become a pain in the neck.

Separating out the storage software layer brings great advantages in flexibility and management.

Add the fact that in many organisations, it won't just be filers from a single vendor – there may be block-based SAN systems, backup systems, filers from other vendors for other purposes, and so on. That means silos, and that often leads to wasted storage and considerable additional management overhead.

Combining server virtualisation with all the associated tools for the policy-based creation, movement and management of VMs has transformed the business of server

administration, making it possible to run many more servers on fewer physical machines and manage them with fewer people. Now we need the same to happen with storage. To do that we must disaggregate storage, abstracting the hardware layer and centralising the software layer.

How do you disaggregate storage?

In the late 1990s, the first network-based storage virtualisation tools showed what was possible by abstracting disparate storage devices into a network-wide pool of generic blocks, and then building new hardware-independent logical (or virtual) storage volumes from that pool.

The difference is that back then the management tools were less sophisticated and the typical storage volumes were smaller, whereas today SDS offers a more functionally rich solution to significant storage problems. SDS today is much more than storage virtualisation, just as vSphere® and its ilk are now far more than merely an x86 server hypervisor. Indeed, some storage virtualisation tools have now evolved into full SDS solutions.

What capabilities does SDS deliver?

In addition to the basic virtualisation of storage, it is essential that SDS provides a management and policy-based automation layer. Routine storage administration can then be performed at a higher level, with tasks automated wherever feasible. Policy-based automation also allows you to offer new services to your users, such as self-provisioning to support DevOps or dynamic QoS management. SDS solutions can also bring greater transparency, with some implementations allowing users to monitor and cost-manage their storage usage.

The disaggregation inherent in SDS can also bring other opportunities. A key one for many will be that you can now upgrade the three elements of the storage infrastructure – that is, media, controllers and software – separately, to take advantage of new technologies and capabilities. For instance, a simple software update might add totally new storage management functionality.

From the capacity and performance perspectives, SDS means seamless and non-disruptive scalability, for example by adding faster media such as flash. And because of the layer of abstraction over the physical storage, many SDS infrastructures can support a combination of file, block and object data access from the same storage pool.

Taken together, all this flexibility means that you can more readily adapt the storage infrastructure to meet changing business demands. For example, some SDS implementations can tailor the storage infrastructure to specific requirements, such as prioritising storage cost, performance or density. And of course you can use lower-cost commodity storage hardware, with the SDS layer adding the intelligence that used to cost a lot of money when it was bought as part of an enterprise storage array or filer.

Storage needs to go through the same transition that turned application servers into virtual machines.

Policy-based automation of pooled storage can simplify operations, enable new business services, and bring greater transparency.

SDS can implement enterprise-grade storage features on commodity hardware.

Potential triggers for SDS

However useful SDS may be, few organisations will be in a position to take a 'Big Bang' approach and deploy site-wide SDS for everything. Instead, the most common approach is to pick a particular project for a first SDS deployment.

When it comes to where and why to start, potential triggers for an initial SDS deployment include, but are not limited to, the following:

Simplifying storage management: Simplifying storage management is an obvious reason to take the SDS route. While simpler management was not enough on its own to drive mainstream acceptance of storage virtualisation, this time it is different. First, because the management load is rising almost exponentially, and second because as discussed above, SDS is today a much more practical solution and some SDS solutions have paid considerable attention to making routine storage management far more straightforward and automatable than ever before.

Simplifying storage management is an obvious reason to take the SDS route.

Reduced hardware cost: Traditional enterprise storage is relatively expensive to acquire and maintain. This can act as a trigger for investigating SDS, which can use cheaper commodity hardware. In some smaller-scale cases, simple cost reduction may be enough of a driver on its own, for example where SDS is delivered as a pre-packaged system that replaces an enterprise array. However, in most cases cost will help to open the door but then will combine with other factors to drive adoption, because SDS can be so much more than just a different way of building an enterprise array.

SDDC component: Another reason for adopting SDS is as a precursor to, or essential foundation for, a wider SDDC (software-defined data centre) strategy. Games of 'Buzzword Bingo' aside, there are good reasons why you might want to do this. The most fundamental is that storage is essential to the data centre infrastructure, so if you are moving to a virtualised and software-driven data centre, SDS and also of course SDN (software-defined networking) will necessarily be parts of that.

Software-defined storage and software-defined networking are also essential for a software-defined data centre.

Cloud foundation: For much the same reason, SDS will also be a good fit for a private or hybrid cloud project. The automation that SDS can offer is a key part of building the sort of self-service and policy-based provisioning systems that are essential to achieving the cloud's economies of administration and scale, never mind the resource flexibility cloud deployments require.

Hyper-converged infrastructure (HCI): Like private clouds, these systems gain their simplicity of management from high degrees of underlying abstraction and automation. Many HCI systems are therefore built upon a foundation of SDS, which is to say a storage virtualisation layer plus tools to automate the provisioning of storage, compute and networking when a new virtual machine or service is defined.

Many HCI systems are built upon a foundation of SDS.

Storage efficiency: SDS provides an opportunity to greatly reduce 'wasted storage' and improve storage flexibility. For example, storage resources that have been allocated but never used, or which without SDS or storage virtualisation would only be available to a single application or service.

The bottom line

SDS lets you start small but plan to grow big.

Software-defined storage takes 60-plus years of data storage evolution to its next logical step. Like virtualised computing, it first abstracts the useful representation of storage from its physical manifestation, and then adds a management layer that makes the resulting pooled storage resource simpler and faster to plan, deploy, administer, redeploy and automate.

Done appropriately – SDS can be implemented in several different ways – the automation element can improve IT agility and flexibility, and therefore business agility. At the same time the use of commodity hardware can reduce storage costs while the abstraction element allows both elastic data scaling and more efficient utilisation of storage pools.

Abstraction also enables many SDS infrastructures to support file, block and object data access, and SDS is a necessary precursor for a full software-defined data centre.

Considered together there could be a lot of things to add in to an SDS business case.

Finally, each of the improvements or innovations that SDS can offer is interesting in its own right, especially to hard-pressed storage administrators and to organisations struggling to cope with and pay for exponential data growth. However, it is when all those capabilities are considered together – and after all, they do typically arrive as a package – that the business case for SDS can get really interesting.

Further reading

The following document is available for free download from the Freeform Dynamics or Western Digital websites.

Getting into Software-Defined Storage
What you need to know before you start

About Freeform Dynamics

Freeform Dynamics is an IT industry analyst firm. Through our research and insights, we aim to help busy IT and business professionals get up to speed on the latest technology developments, and make better-informed investment decisions.

For more information, and access to our library of free research, please visit www.freeformdynamics.com.

About Western Digital

Western Digital Corporation (NASDAQ: WDC) is an industry-leading provider of storage technologies and solutions that enable people to create, leverage, experience and preserve data. The company addresses ever-changing market needs by providing a full portfolio of compelling, high-quality storage solutions with customer-focused innovation, high efficiency, flexibility and speed. Our products include storage drives, platforms, and systems marketed to OEMs, distributors, resellers, cloud infrastructure providers and consumers.

Terms of Use