Data Center Evolution

A data center is a physical facility that organizations use to house their critical applications and data. A data center's design is based on a network of computing and storage resources that enable the delivery of shared applications and data.

What defines a modern data center?

Infrastructure has shifted from traditional on-premises physical servers to virtualized infrastructure that supports applications and workloads across pools of physical infrastructure and into a multicloud environment.

The modern data center is wherever its data and applications are. It stretches across multiple public and private clouds to the edge of the network via mobile devices and embedded computing.

Why are data centers important to business?

In the world of enterprise IT, data centers are designed to support business applications and activities that include:

1. Email and file sharing,
2. Productivity applications
3. Customer relationship management (CRM) and enterprise resource planning (ERP)
4. Big data, artificial intelligence, and machine learning
5. Communications and collaboration services

What are the core components of a data center?

- They are routers, switches, firewalls, storage systems, servers, and application delivery controllers.
- Because these components store and manage business-critical data and applications, data center security is critical in data center design.
- Together, they provide Network infrastructure, Storage infrastructure, and Computing resources.

Data centers are at the center of modern software technology and they enable businesses to do more with less.

What is in a data center facility?

Data center components require significant infrastructure to support the center's hardware and software. These include power subsystems, uninterruptible power supplies (UPS), ventilation, cooling systems, fire suppression, backup generators, and connections to external networks.

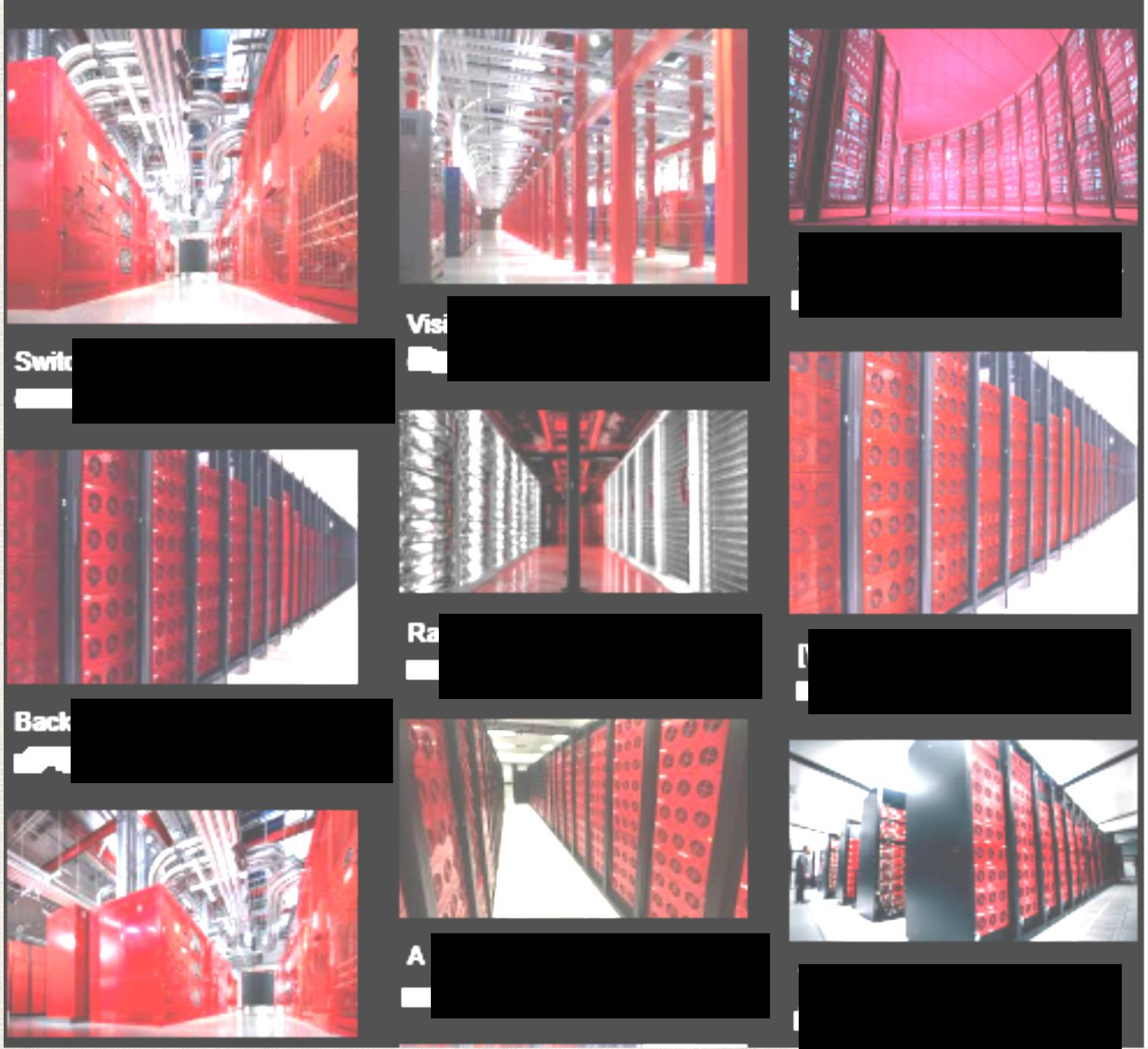
Types of Data Centers

Many types of data centers and service models are available.

Their classification depends on whether they are owned by one or many organizations, how they fit (if they fit) into the topology of other data centers, what technologies they use for computing and storage, and even their energy efficiency.

Types of Data Centers

- Enterprise data centers
- Managed services data centers
- Colocation data centers
- Cloud data centers



[redacted]

[redacted]

[redacted]

[redacted]

[redacted]

[redacted]

[redacted]

[redacted]

Top-Tier Data Center Players (Mega Clouds)

1. Amazon

With Amazon Web Services, users can "launch virtual machines or apps within minutes," a drastic reduction in provisioning time compared to traditional models.

Through a collection of remote computing services, Amazon Web Services creates and offers a cloud computing platform to subscribers.

2. Google Cloud Platform

An obvious contender as one of the largest data center players, Google offers options via Google Cloud Platform and Google Compute Engine.

Google's Cloud Platform enables developers to build, test and deploy applications on the same infrastructure that powers Google's vast capabilities. Google Compute Engine offers 99.95 percent monthly SLAs with 24-7 support.

3. Microsoft – Windows Azure

Microsoft rounds out the big three when it comes to major players in the cloud computing space with Windows Azure, offering the services of Microsoft-managed data centers in 13 regions around the world.

With per-minute billing and built-in auto-scaling, it proves to be a cost-efficient platform to build and deploy applications of any size.

Other Data Center Players: Rackspace, VMWare, Facebook etc.

A History of the Modern Data Center

Data centers play a critical role in the expanding capabilities for enterprises.”

The concept of “data centers” has been around since the late 1950s when American Airlines and IBM partnered to create a passenger reservations system offered by Sabre, automating one of its key business areas.

The idea of a data processing system that could create and manage airline seat reservations and instantly make that data available electronically to any agent at any location became a reality in 1960, opening the door to enterprise-scale data centers.

Since then, physical and technological changes in computing and data storage have led us down a winding road to where we are today.

Let’s take a brief look at the evolution of the data center, from the mainframe of yesterday, to today’s cloud-centric evolution, and some impacts they’ve had on IT decision-making.

The first few decades in the life of the room that eventually became known as the “data center” were characterized by electromechanical computers made from electrical switches and mechanical relays, and later by all electronic computers that used vacuum tubes as switches.

1946

The Electronic Numerical Integrator and Computer (ENIAC) was built in 1946 for the U.S. Army to store artillery firing codes and was dubbed as the first general-purpose electronic digital computer.

Early 1960s

The first transistorized computer (TRADIC) was introduced in 1954 and was the first machine to use all transistors and diodes and no vacuum tubes. The innovation responsible for the data center as we know it today was the transistorized, integrated circuit based microprocessor.

Maturity in this technology eventually led to Intel’s 8086 chip, and all of its Successors.

1971

Intel introduced its 4004 processor, becoming the first general-purpose programmable processor on the market. It served as a “building block” that engineers could purchase and then customize with software to perform different functions in a wide variety of electronic devices.

1973

The Xerox Alto becomes the first desktop computer to use a graphical UI and included a bit-mapped high-resolution screen, large internal memory storage, and special software.

1977

ARCnet is introduced as the first LAN.

1980s

Personal computers (PCs) were introduced in 1981, leading to a boom in the microcomputer industry.
Sun Microsystems developed the network file system protocol, allowing a user on a client computer to access files over a network in a manner similar to how local storage is accessed.

Early 1990s

Microcomputers began filling old mainframe computer rooms as "servers," and the rooms became known as **data centers.** Companies then began assembling these banks of servers within their own walls.

Mid 1990s

The enterprise construction of server rooms, lead to much larger facilities (hundreds and thousands of servers). The **data center as a service model** became popular at this time.

2002

Amazon Web Services begins development of a suite of cloud-based services, which included storage, computation and some human intelligence through “Amazon Mechanical Turk.”

2006

Amazon Web Services begins offering IT infrastructure services to businesses in the form of web services, now commonly known as cloud computing.

2007

Sun Microsystems introduces the modular data center, transforming the fundamental economics of corporate computing.

2011

Facebook launches Open Compute Project, an industry-wide initiative to share specifications and best practices for creating the most energy efficient and economical data centers. About 72% of organizations said their data centers were at least 25 percent virtual.

2013

Telcordia introduces generic requirements for telecommunications data center equipment and spaces.

Google invested a massive $7.35 billion in capital expenditures in its Internet infrastructure during 2013. The spending was driven by a massive expansion of Google's global data center network, which represented perhaps the largest construction effort in the history of the data center industry.

Today and Beyond

Today's datacenters are shifting from an infrastructure, hardware and software ownership model, towards a subscription and capacity on demand model.

Two constantly developing technologies — the microprocessor/x86 architecture and disk-based storage medium — form the foundation for the modern data center. The modern data center provides Software Defined Storage to serve I/O to the applications, thereby offering IT organizations greatest storage flexibility and ease of use. It is also possible to develop projects that once seemed like pure science fiction.

More with Less

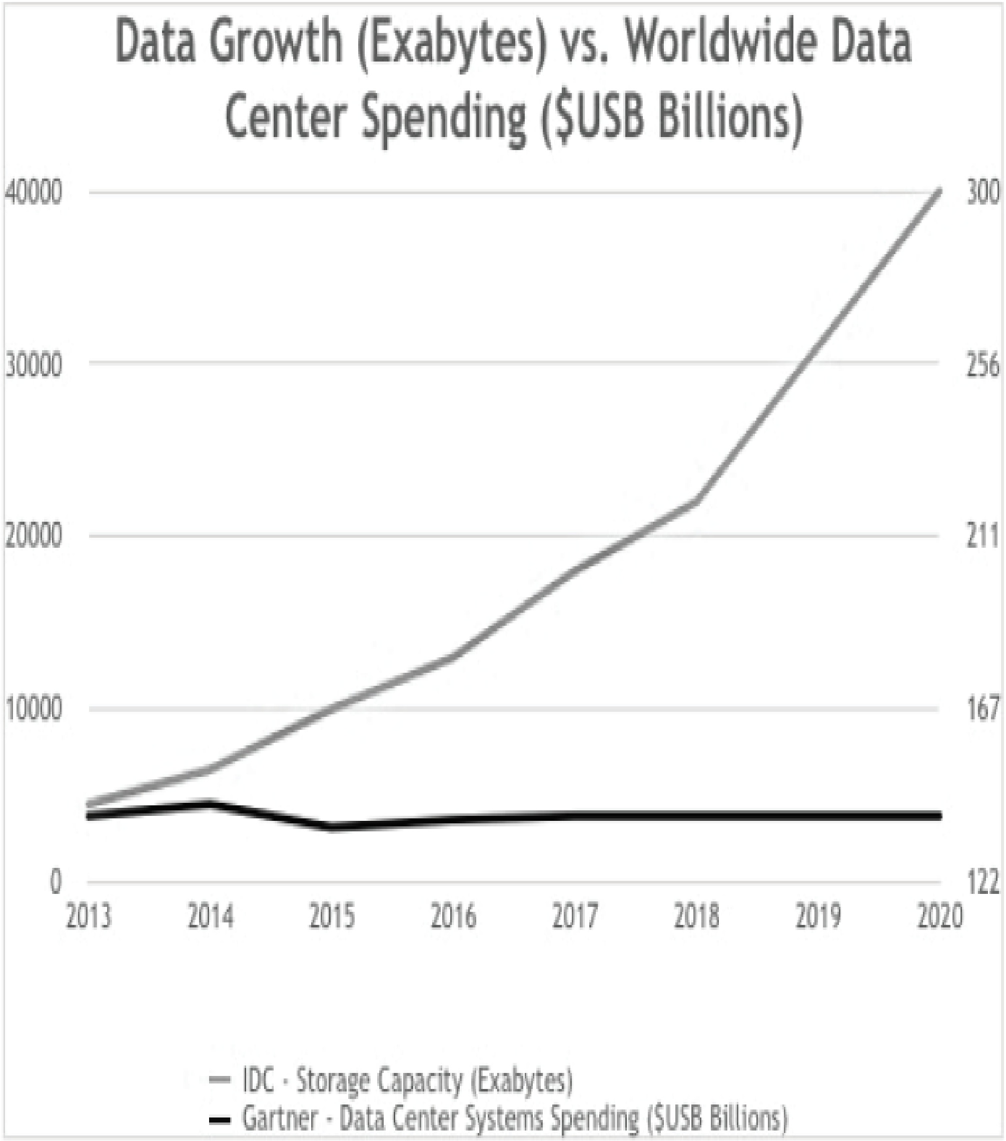


Figure Data Growth vs. Data Center Spending

Technology in the data center tends to be Cyclical and the technology improves with each new Cycle.

In the 1990s, the prevailing data center design had each application running on a server, or a set of servers, with locally attached storage media.

As the quantity and criticality of line-of-business applications supported by the data center grew, this architecture began to show some dramatic inefficiency when deployed at scale.

Plus, the process of addressing that inefficiency has characterized the modern data center for the past two decades.

The Rise of the Monolithic Storage Array

The inefficiency at scale actually had two components:

1. Servers very commonly used only a fraction of the computing power. Typically the server ran at 10% CPU utilization, thus wasting massive amounts of resources.

2. The available storage was under utilized. There were islands of storage created by placing direct attached storage with every server which could not be fully utilized. There came a great inefficiency caused by the need to allow room for growth.

For ex: If an enterprise had 800 servers in their data center. If each of those servers had 60 GB of unused storage capacity to allow for growth. That would mean there was 48 TB of unused capacity across the organization.

Paying for 48 TB of capacity to just sit on the shelf seems absurd, but until this problem could be solved, that was the accepted design.

Solution to this problem: Storage Area Network

- In a storage area network (SAN), rather than providing direct-attached storage for each server, disks are pooled and made accessible via the network.

- This allows many devices to draw from one capacity pool and increase utilization across the enterprise dramatically.

- It also decreased the management overhead of storage systems, because it meant that rather than managing 800 storage silos, perhaps there were only 5 or 10.

- These arrays of disks (“storage arrays”) were connected on a network segregated from the local area network.

- The SAN made use of a different network protocol more suited for storage networking called Fibre Channel Protocol.

- It was more suited for delivering storage because of its “lossless” and high-speed nature.

- The purpose of the SAN is to direct and store data, and therefore the loss of transmissions is unacceptable.

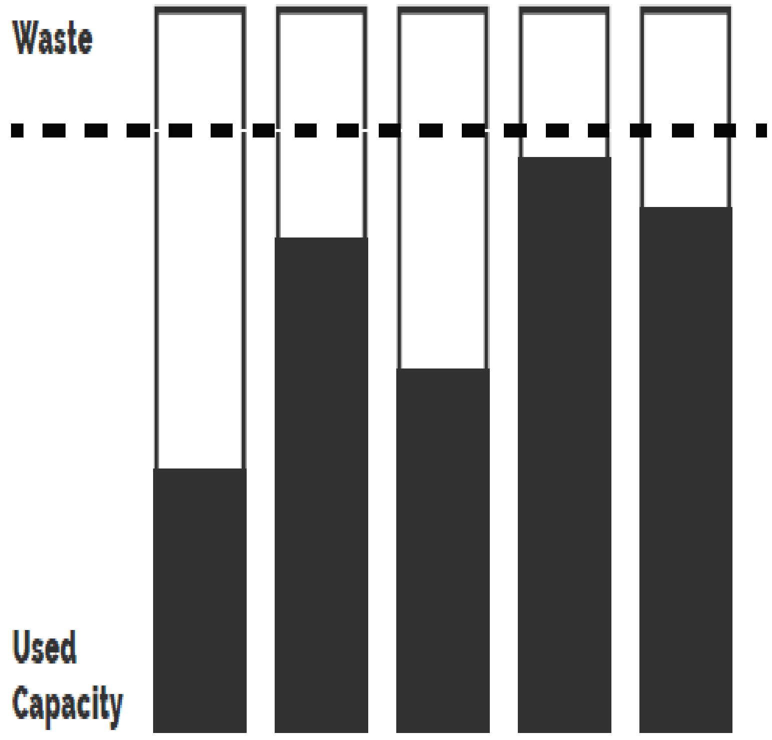


Figure Inefficiency of the microprocessor/x86 architecture & Disk-based storage

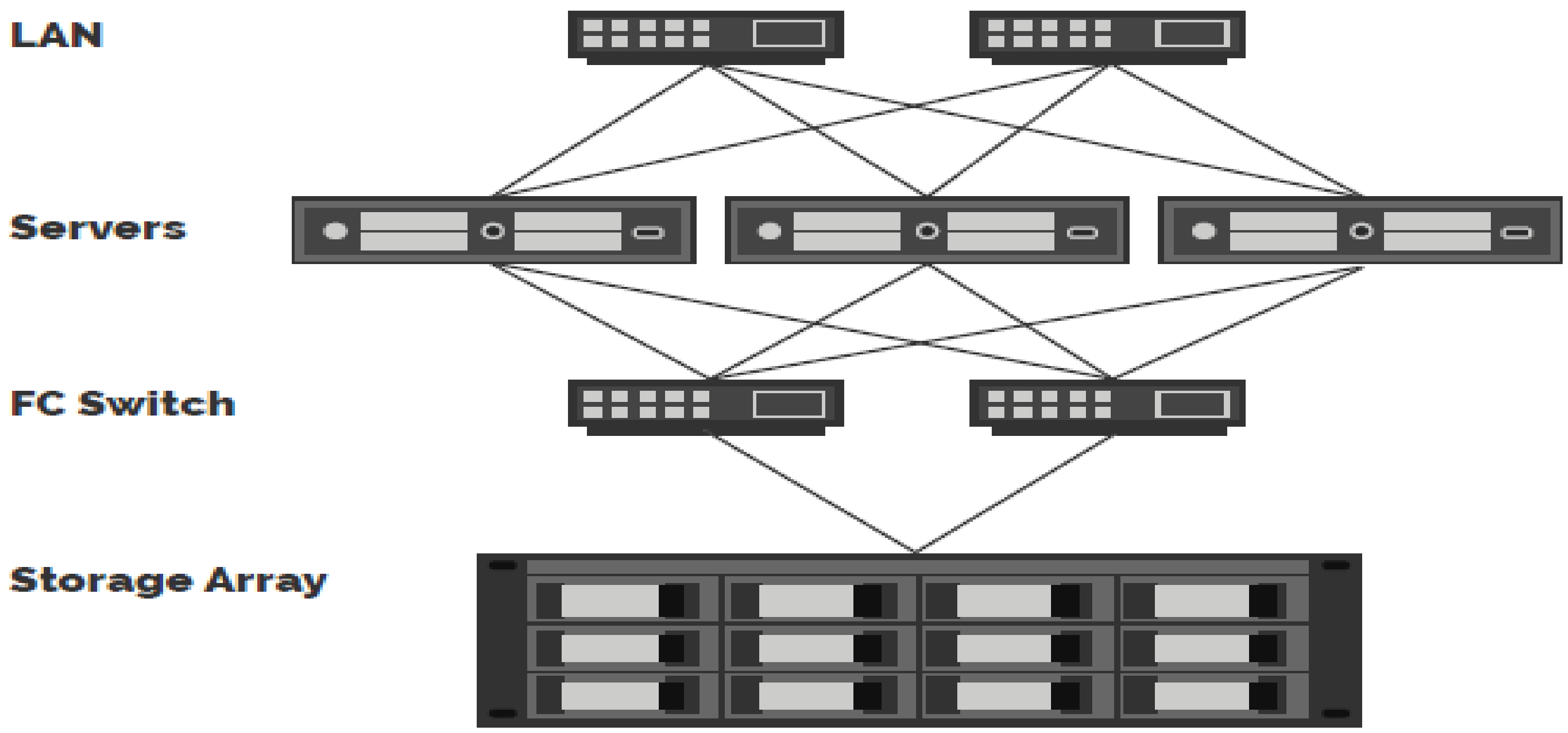


Figure : A Simple SAN

File vs. Block Storage

- Data stored on a shared storage device is typically accessed in one of two ways: at the block level or at the file level.

- In file level access the granularity of access is a full file. Protocols used are NFS, and SMB. File-based protocols may be used where an end user or application will be accessing the files directly —for example a network share.

For block storage the protocols used are: Fibre Channel (block), iSCSI (block). Block level access sends SCSI commands directly from the initiator (client side) to the target (storage array side).

Block-based protocols are more likely to be used when an operating system or hypervisor is accessing the storage, as direct access to the disk is preferred.

Data Services: Allow the administrator to manipulate and protect the stored data.

Types:

a. **Snapshot:** A storage feature that allows an administrator to capture the state and contents of a volume or object at a certain point in time. A snapshot can be used later to revert to the previous state. Snapshots are also sometimes copied off site to help with recovery from site-level disasters.

b. Replication: A storage feature that allows an administrator to copy a duplicate of a data set to another system. Replication is the most commonly used data protection method; copies of data are replicated off site and are available for restore in the event of a disaster.

Replication can also have other uses, however, like replicating production data to a testing environment.

c. Data Reduction: There is a large amount of duplicate data in enterprise environments. Virtualization compounds this issue.

Many storage platforms are capable of compression and deduplication, which both involve removing duplicate bits of data.

Compression happens to a single file or object, whereas deduplication happens across an entire data set. By removing duplicate data, often only a fraction of the initial data will be stored.

As the industry matured and more organizations adopted a shared storage model, the value of the architecture continued to increase.

In addition to file system snapshots, administrators could make use of volume-level snapshots also. This created new possibilities for backup and recovery solutions to complete backups faster and more efficiently.

Storage systems also contained mechanisms for replicating data from one storage array to another. This meant that a second copy of the data could be kept up-to-date in a safe location, as opposed to backing up and restoring data all the time.

Perhaps one of the greatest efficiencies achieved by adopting the shared storage model was the potential for global deduplication of data across the enterprise.

By the mid-2000s, average data centers had the efficiency of using shared storage across servers and applications, combined with the added efficiency of being able to globally deduplicate that data.

Performance of the shared storage systems grew as manufacturers continued to improve the networking protocols, the physical disk media, and the file systems that governed the storage array.

Using shared storage allowed more agility and flexibility with servers than was known with direct-attached storage.

Many organizations chose to provision the operating system disk for a server on the storage array and use a “boot from SAN” model.

The benefit of deploying operating systems this way was this: if one physical server failed, a new server could replace it, be mapped to the same boot volume, and the same operating system instance and applications could be back up and running in no time.

All these efforts brought down the cost of data centers. However there was still the problem of compute resources.

CPU resources were still generally configured far above the actual utilization of the application the server was built for.

Eliminating this problem was the second frontier in solving inefficiency in the modern data center.

The Virtualization of Compute — Software Defined Servers

Virtualization as a concept is not a new development. Virtualization has been around since the 1960s when the technology was developed to allow multiple jobs to run simultaneously on a mainframe.

Virtualization allowed for multiple workloads to run in tandem on shared hardware, yet be isolated from one another.

But the true power of modern virtualization came in 2001, when VMware released ESX, a bare-metal hypervisor capable of virtualizing server workloads in the data center. Later Microsoft came up with Hyper-V.

The **hypervisor**, a software that abstracts physical resources like CPU and memory from the virtual machines, is capable of running multiple workloads simultaneously and effectively isolated from one another.

It was estimated that the data center industry in 2006 consumed 61 billion kilowatt hours of electricity due to the large no. of physical servers. There was a great need to cut down the numbers without affecting the businesses. Virtualization was the answer.

Rather than having 10 physical servers running at 10% utilization, there were now two servers running at 50% or higher utilization. This brought down administrative as well as networking(cable) costs.

As hypervisor and virtual machine performance increased the demands on related infrastructure components also increased. There was a need for higher bandwidth, higher disk performance and lower latency. Spinning disks have served as primary storage, and tape-based storage systems have served higher capacity longer term storage needs.

The speed by which data on a spinning disk can be accessed cannot be increased beyond a certain limit as it will lead to the damage of the disk. There’s also the issue of latency. Due to the mechanical nature of a spinning disk drive, latency (the time it takes to retrieve or write the data in question) can’t be pushed below a certain threshold.

Tiny bits of latency added together across many drives becomes an issue at scale. This led to the replacement of spinning disk by flash storage in the data centers so as to increase the speed of data access.

Because flash storage is not mechanical in nature, it doesn’t suffer from the same limitations as spinning disks. Flash storage is capable of latency on the order of microseconds as opposed to spinning disk’s multiple milliseconds. It’s also capable of far more I/O operations per second than a handful of spinning disks. Also flash storage durability has improved over time.

Lastly, because of the non-mechanical (or “solid state”) nature of flash storage, it requires much less power to operate when compared to spinning disk. As data center power bills move northwards any way to reduce power consumption is welcome.

The Fall of the Monolithic Storage Array

Monolithic storage arrays solved many of the data center’s problems and allowed IT to achieve greater efficiencies and scale. Unfortunately, the things that made this architecture so attractive also eventually became its downfall. The virtualization of compute led to densities and performance requirements that storage arrays have struggled to keep up with ever since.

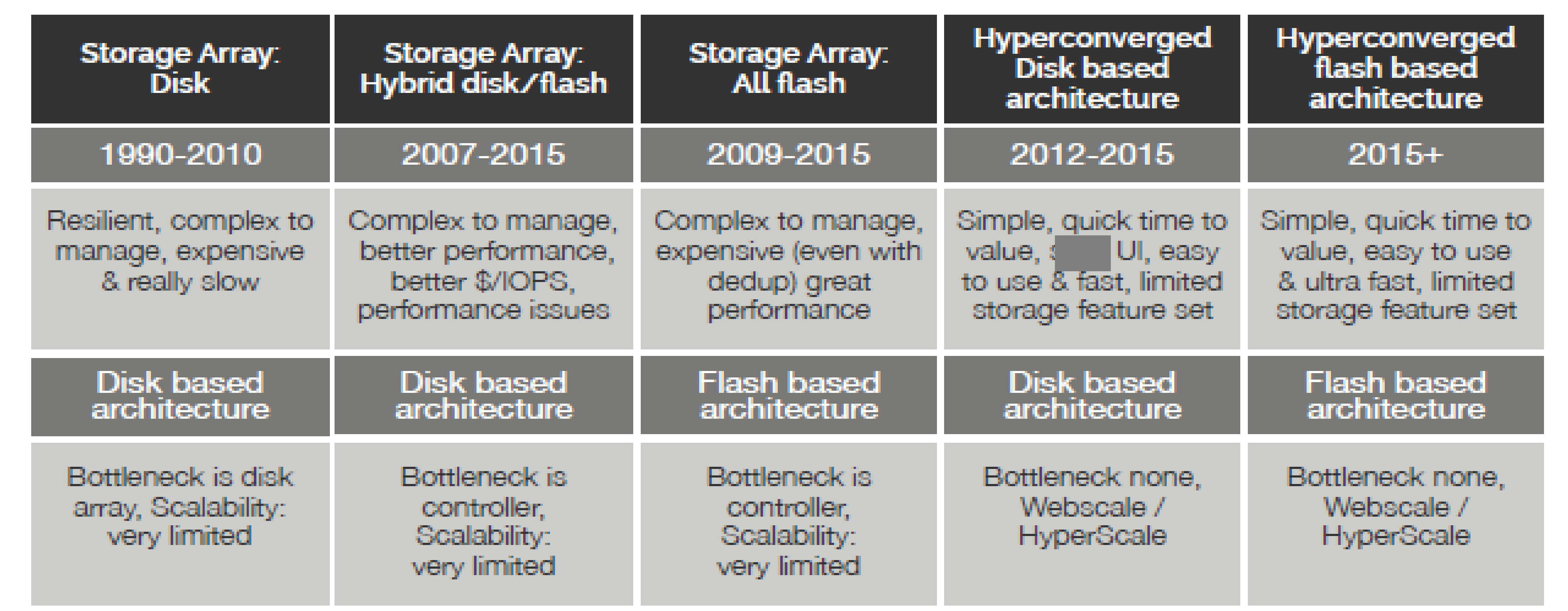
One of the primary challenges here is “mixed workload.” By the nature of virtualization, many different applications and operating systems share the same physical disk infrastructure on the back end.

The challenge with this architecture is that operating systems, & especially applications, have widely varying workload requirements and characteristics. For example, attempting to deploy virtual desktop infrastructure (VDI) on the same storage platform as the server virtualization has been the downfall of many VDI projects.

Due to the drastically different I/O characteristics of a desktop operating system versus a server operating system and the applications running on them, they require almost completely opposite things. An average Windows server might require 80% reads and 20% writes, whereas on the exact same storage array, with the same disk layout, same cache, and so on, a virtual desktop might require 20% reads and 80% writes.

As application performance requirements go up, it has also become increasingly important to provide very low latency. So which storage model is likely to have lower latency: the one where storage is accessed across a network and shared with all other workloads, or the one where storage is actually inside the server.

The answer is the model where the storage is local to the workload. Some new ideas have started popping up in the data center storage market over the past few years. Figure 2-5 shows the progression of storage design over time.



Storage Array: Disk	Storage Array: Hybrid disk/flash	Storage Array: All flash	Hyperconverged Disk based architecture	Hyperconverged flash based architecture
1990-2010	2007-2015	2009-2015	2012-2015	2015+
Resilient, complex to manage, expensive & really slow	Complex to manage, better performance, better $/IOPS, performance issues	Complex to manage, expensive (even with dedup) great performance	Simple, quick time to value, [redacted] UI, easy to use & fast, limited storage feature set	Simple, quick time to value, easy to use & ultra fast, limited storage feature set
Disk based architecture	Disk based architecture	Flash based architecture	Disk based architecture	Flash based architecture
Bottleneck is disk array, Scalability: very limited	Bottleneck is controller, Scalability: very limited	Bottleneck is controller, Scalability: very limited	Bottleneck none, Webscale / HyperScale	Bottleneck none, Webscale / HyperScale

Figure 2-5: Storage design timeline

The data center of the future looks (physically) a lot more like the data center of the past, in which a number of servers all contain their own direct attached storage.

The difference is that all of this locally attached storage is pooled, controlled, accelerated, and protected by a storage management platform running on the hypervisor. The performance and scale implications of this model are massive: because each node added to the cluster with local storage contributes to the pool, this means that the storage pool can grow to virtually limitless heights.

Each server that is added has its own storage controller, meaning that throughput never becomes an issue. Increasing capacity of the pool is as easy as adding disks to existing servers or adding more servers overall. The control of all of this is done by either virtual machines (VSAs) or by kernel-level software, and the administrator typically manages it from the hypervisor’s existing management interface.

Software Defined Storage (SDS) is changing the data center in tangible ways, and as more organizations begin to adopt this architecture, vendors of monolithic storage arrays will have to innovate in order to stay relevant and survive.

The Emergence of Convergence

As the challenges for IT have grown in equal proportions with the ever-increasing scope of their responsibilities, IT decision makers have often looked to outsource parts of their operation.

A notable trend for data center “outsourcing” of sorts is now referred to as convergence.

Convergence is multiple pieces of the infrastructure assembled prior to delivery to the customer. Convergence saves time and frustration during the deployment phase and provides decreased time-to-value after procurement.

An example of a common form of convergence might look like this: a rack is delivered to the data center already containing a storage array, a blade chassis populated with blades, and a few top-of-rack switches. Everything is cabled up, and all the configuration of the switching and storage has been done prior to delivery.

At the moment the converged stack is delivered, the data center team can roll it into place, deliver power and upstream network connectivity, and the pod will be up and running.

This model of growing the infrastructure is substantially faster than the traditional model of having parts delivered, assembling them, hiring consultants, troubleshooting, and so on.

The value in convergence comes not only from the fact that the solution comes pre-assembled, but also from the fact that it includes all the pieces necessary. Half the challenge in traditional piecemeal solution-building is getting all the right parts and ensuring interoperability.

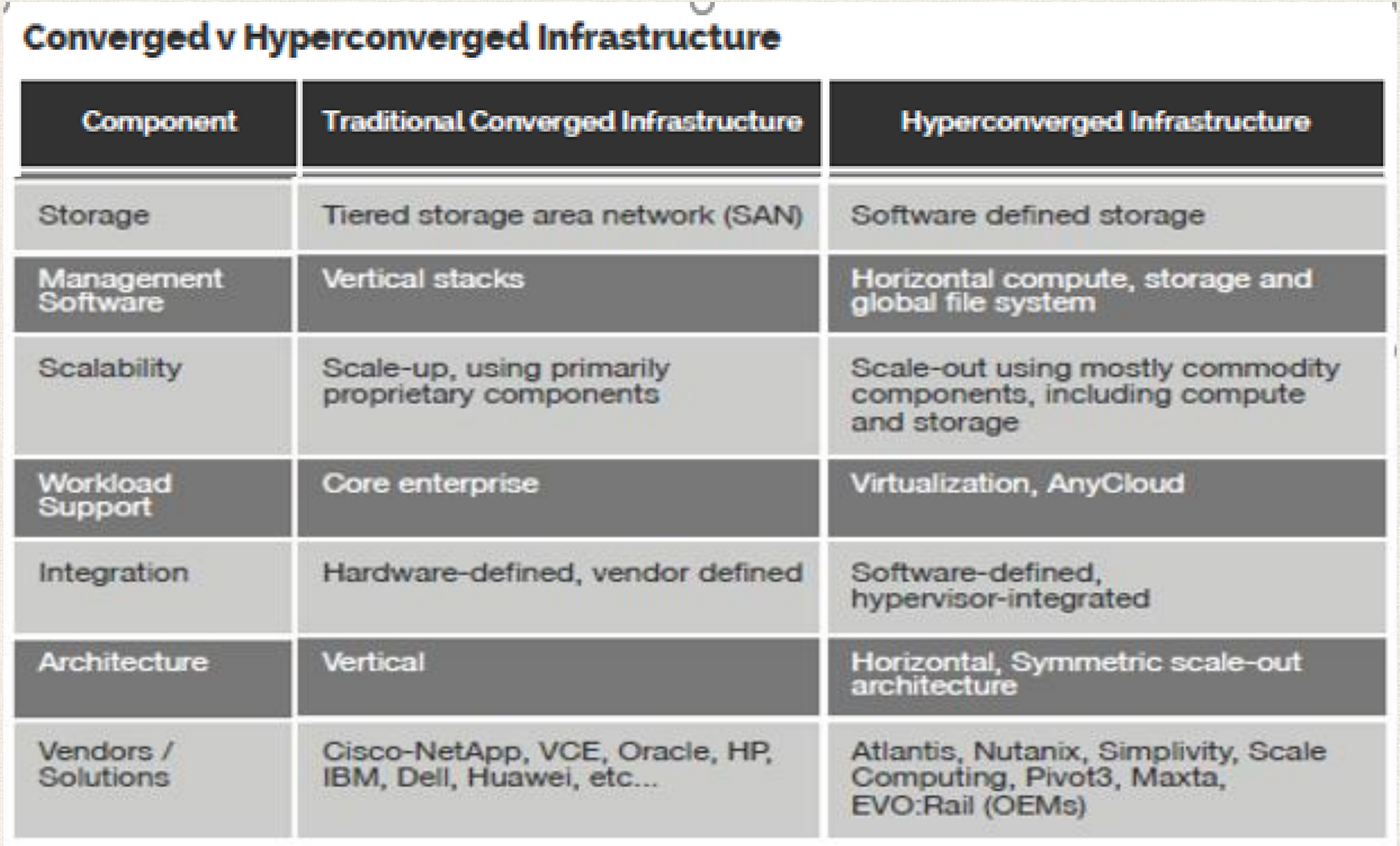
Convergence guarantees that with the purchase of a certain SKU, all the components contained within it will be compatible with one another, and all the necessary parts will be included. This has helped many organizations realize project objectives faster, and has saved a multitude of headaches over time.

But if a little convergence was good, does that mean a lot of convergence is great?

The successor to convergence is known as “hyperconvergence,” and it takes the idea of simplicity to the customer to new heights.

Hyperconvergence is so called because of the scope of what is being converged. In a converged infrastructure, many infrastructure components are brought together into one rack (or a few racks).

In a hyperconverged infrastructure (HCI), those same components are brought together within a single server node. Hyperconvergence is born from cloud data centers that pioneered and leveraged this technology to operate at the massive scale they require.



Converged v Hyperconverged Infrastructure

Component	Traditional Converged Infrastructure	Hyperconverged Infrastructure
Storage	Tiered storage area network (SAN)	Software defined storage
Management Software	Vertical stacks	Horizontal compute, storage and global file system
Scalability	Scale-up, using primarily proprietary components	Scale-out using mostly commodity components, including compute and storage
Workload Support	Core enterprise	Virtualization, AnyCloud
Integration	Hardware-defined, vendor defined	Software-defined, hypervisor-integrated
Architecture	Vertical	Horizontal, Symmetric scale-out architecture
Vendors / Solutions	Cisco-NetApp, VCE, Oracle, HP, IBM, Dell, Huawei, etc...	Atlantis, Nutanix, Simplivity, Scale Computing, Pivot3, Maxta, EVO:Rail (OEMs)

Figure 2-6: Converged vs. Hyperconverged Infrastructure

The Role of Cloud

Cloud technology is being leveraged all over the world by even small companies. Cloud computing is a model of delivering infrastructure or application resources in a flexible, rapid, and on-demand manner. This is why purchasing infrastructure from Amazon Web Services (AWS), for example, would be classified as cloud. It's on-demand, takes about two minutes to provision, & has tons of options. Because cloud is a model and not a thing, there are a number of different ways in which cloud infrastructure can be implemented.

Cloud Types

A. Based on Deployment

Different cloud deployment models fit different organizations. There are certain cases where an application has been developed to run in a cloud. In this case, it may make sense to use the Cloud. Cloud is simply a method of offering & provisioning on-demand services.

A ***private cloud*** deployment is simply an on-premises deployment of a tool like OpenStack that allows for rapid, on-demand provisioning of resources that can easily be created & destroyed. It can also be remotely managed. Private cloud offers higher control and security.

A ***public cloud*** model is one where all resources are provisioned in a third party data center provided by the likes of AWS, Microsoft, VMware, Google, or a friendly neighborhood cloud provider. Especially for some small businesses, being entirely public-cloud-based allows for an extremely light IT footprint in the office or storefront, resulting in less overhead.

Public cloud can be very affordable. It also offloads risk and overhead in terms of compliance, patching, equipment failure, hardware refreshes, and so on. Public cloud offers less or no control and security.

The next possible choice is a combination of on-premises cloud and public cloud; it's known as ***hybrid cloud***. Using this model, IT resources run in the corporate data center as usual, but an extension to a public cloud data center is in place. This means that based on certain requirements, constraints, or other design decisions, a workload can be provisioned either to the private data center or to the public one.
Ex1: An example of how hybrid cloud might work is that of a retailer. If Black Friday is coming up, the retailer may be able to spin up an extra 20 instances of their website and shopping cart application in the public cloud. The back end databases still exist in the on-premises data center and need not be migrated. This is commonly referred to as "bursting" to the cloud.

Ex2: A hybrid cloud model could work out well is in an organization that has a heavy in-house development workload. If developers are constantly creating and destroying test environments, it can require lots of horsepower to keep things running fast enough that developers are happy, and project scopes can change with a moment's notice. A much easier way to handle this situation would be to run production workloads in the on-premises data center, but have development and testing workloads provision to the public cloud. This can also save on cost as opposed to running the resources locally.

Ex3: In one of the surveys roughly 10% of all survey respondents said that they're managing over a petabyte of storage at their primary site.