

IT Infrastructure Architecture

Infrastructure Building Blocks
and Concepts

Storage

Network Attached Storage (NAS)

- A NAS, also known as a File Server, is a network device that provides a shared file system to operating systems over a standard TCP/IP network
 - NFS (UNIX and Linux)
 - SMB/CIFS (Windows)
- A NAS is often an appliance that implements the file services and holds the disks on which data is stored
- A NAS appliance could also use external disk storage provided by a SAN
- Can provide snapshot and clone technology at a file level, enabling features like “un-erasing” deleted files by end users



Network Attached Storage (NAS)

- The difference between a SAN and NAS:
 - SAN:
 - Offers disk blocks (unformatted disks called LUNs) that can be used by only one server
 - Uses iSCSI, Fibre Channel or FCoE as the communication layer
 - NAS:
 - Offers a shared filesystem to store files that can be used by multiple servers
 - Connects to for instance to an LDAP or Active Directory service in order to set file and/or folder permissions
 - Uses SMB/CIFS or NFS over TCP/IP as the communication layer

Network Attached Storage (NAS)

- A clustered NAS is a NAS that uses a distributed file system running simultaneously on multiple servers
 - Distributes data and metadata across storage devices
 - Still provides unified access to the files from any of the cluster nodes, unrelated to the actual location of the data

Object Storage

- Object storage is a storage architecture that manages data as objects, where an object is defined as a file with its metadata, and a globally unique identifier called the object ID
- Examples of metadata:
 - Filename
 - Date and time stamps
 - Owner
 - Access permissions
 - The level of data protection
 - Replication settings to for instance a different geography
- Object storage stores and retrieves data using a REST API over HTTP, served by a webserver, and is designed to be highly scalable

Object Storage

- A traditional file system provides a structure that simplifies locating files
 - For example, a log file is stored in `/var/log/proxy/proxy.log`
- In object storage, a file's object ID must be administered by the application using it
 - Using the object ID, the object can be found without knowing the physical location of the data
 - For example, an application has administered that its log file is stored in object ID 8932189023
- Using object IDs enables simplicity and massive scalability of the storage system
 - The object ID is a link to an object that can be stored anywhere

Object Storage

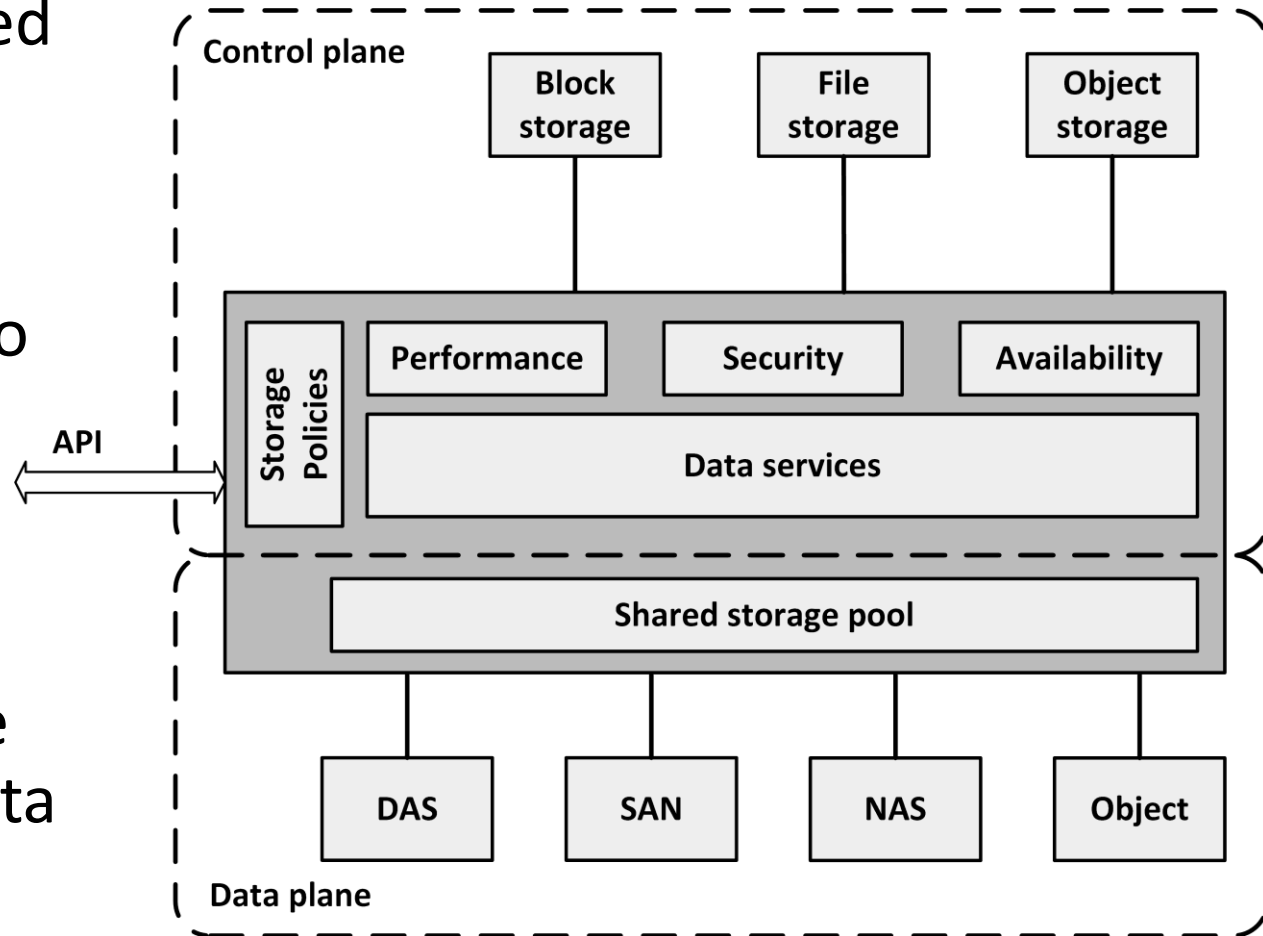
- Data in object storage can't be modified
 - The original file must be deleted, and a new file must be created, leading to a new object ID
- This makes object storage unsuitable for frequently changing data
- It is a good fit for data that doesn't change much, like:
 - Backups
 - Archives
 - Video and audio files
 - Virtual machine images

Object Storage

- Some systems emulate a file system using object storage
 - For instance, Amazon's S3FS creates a virtual filesystem, based on S3 object storage, that can be mounted to an operating system in the traditional way, however, with significant performance degradation
 - A much better solution is to use object storage with applications designed for it

Software Defined Storage

- Software Defined Storage (SDS) abstracts data and storage capabilities (also known as the control plane) from the underlying physical storage systems (the data plane)



Software Defined Storage

- SDS virtualizes all physical storage into one large shared storage pool
 - Data can be stored in a variety of storage systems while being presented and managed as one storage pool to the servers consuming the storage
- Storage can be implemented as software running on commodity x86-based servers with direct attached disks
- Physical storage can also be a SAN, a NAS, or an Object storage system

Software Defined Storage

- From the shared storage pool, software provides data services like:
 - Deduplication
 - Compression
 - Caching
 - Snapshotting
 - Cloning
 - Replication
 - Tiering

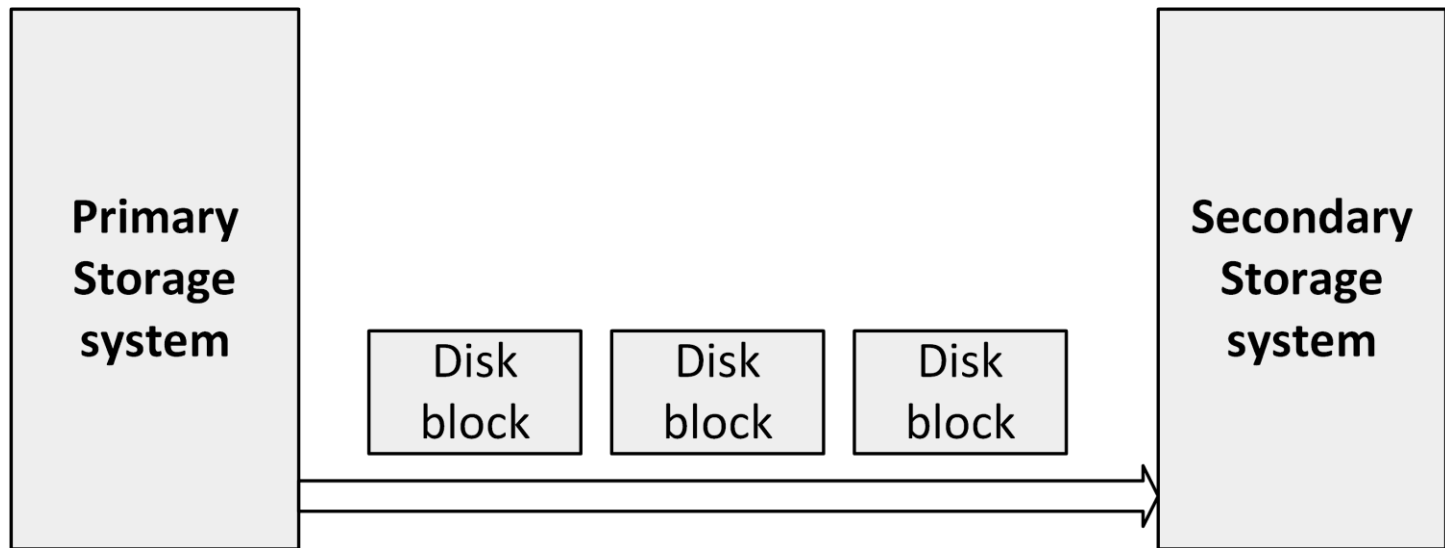
Software Defined Storage

- SDS provides servers with virtualized data storage pools
 - With the required performance, availability and security
 - Delivered as block, file, or object storage
 - Based on policies
- Example:
 - A newly deployed database server can invoke an SDS policy that mounts storage configured to have its data striped across a number of disks, creates a daily snapshot, and has data stored on tier 1 disks
- APIs can be used to provision storage pools and set the availability, security and performance levels of the virtualized storage
- Using APIs, storage consumers can monitor and manage their own storage consumption

Storage availability

Redundancy and data replication

- To increase availability in a SAN, components like HBAs and switches can be installed redundantly
- Using multiple paths between HBAs and SAN switches, failover can be instantiated automatically when a failure occurs
- Multiple storage systems can be used. Using replication, changed disk blocks from the primary storage system are continuously sent to the secondary storage system, where they are stored as well



Redundancy and data replication

- Synchronous replication:
 - Each write to the active storage system and the replication to the passive storage system must be completed before the write is confirmed to the operating system
 - Ensures data on both storage systems is synchronized at all times and data is never lost
 - When the physical cable length between the two storage systems is more than 100 km, latency times get too long, slowing down applications, that have to wait for the write on the secondary storage system to finish
 - Risk: a failing connection between both storage systems a write is never finished, as the data cannot be replicated. This effectively leads to downtime of the primary storage system

Redundancy and data replication

- Asynchronous replication:
 - After data has been written to the primary storage system, the write is immediately committed to the operating system, without having to wait for the secondary storage array to finish its writes as well
 - Asynchronous replication does not have the latency impact that synchronous replication has
 - Disadvantage: potential data loss when the primary storage system fails before the data has been written to the secondary storage system

Backup and recovery

- Backups are copies of data, used to restore data to a previous state in case of data loss, data corruption or a disaster recovery situation
- Backups are always a last resort, only used if everything else fails, to save your organization in case of a disaster
- A well-designed system should have options to repair incorrect data from within the system or by using systems management tools (like database tools)

Backup and recovery

- In general, backups should not be kept for a long time
 - Because the data copies are only relevant in the event of a disaster, organizations will typically have little use to restore a data backup that is more than a few weeks old
 - Restoring a backup takes you back in time
 - Like a time machine, but without the rest of the world – like your business partners and customers – going back in time as well

Backup and recovery

- A common mistake is to mix up backup with archiving
 - Backup is about protection against data loss
 - Archiving deals with long term data storage, in order to comply with law and regulations
- Backups should not be used to view the status of information from the past
 - It should be possible to retrieve these statuses from the system itself
 - No data should ever be deleted in a typical production system
 - Older data could be archived to a secondary system or database

Backup and recovery

- Backups need to be made at a regular basis
 - Usually daily
 - Sometimes more often – every hour, or even continuously in highly critical environments
- 3-2-1 rule:
 - Keep three copies of your data
 - on two different media types
 - with one copy stored at a separate location

Backup and recovery

- Backups must be available at a secondary site for restore
 - Experience with real world disasters shows it is good practice to have a distance of at least 5 km between the main site and the backup data
- Apart from application data, a copy must be available on the secondary site of:
 - Operating system installation disks
 - Printed procedures on how to build up a new system using the backups
 - License keys of the software (including the restore software)

Backup and recovery

- Test the restore procedure at least once a year to ensure restores work as planned
 - Include building up new hardware
 - Have restore procedures tested by a third party, or at least by people that have not performed a restore before
 - In case of a real disaster we cannot assume that systems managers are able to restore data again
- Restore tests should be performed each month to ensure backup media still work as expected
 - Restore some files
 - Do the tapes really contain the expected data?