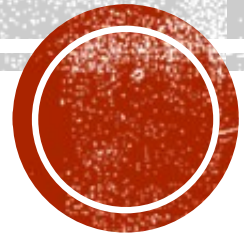
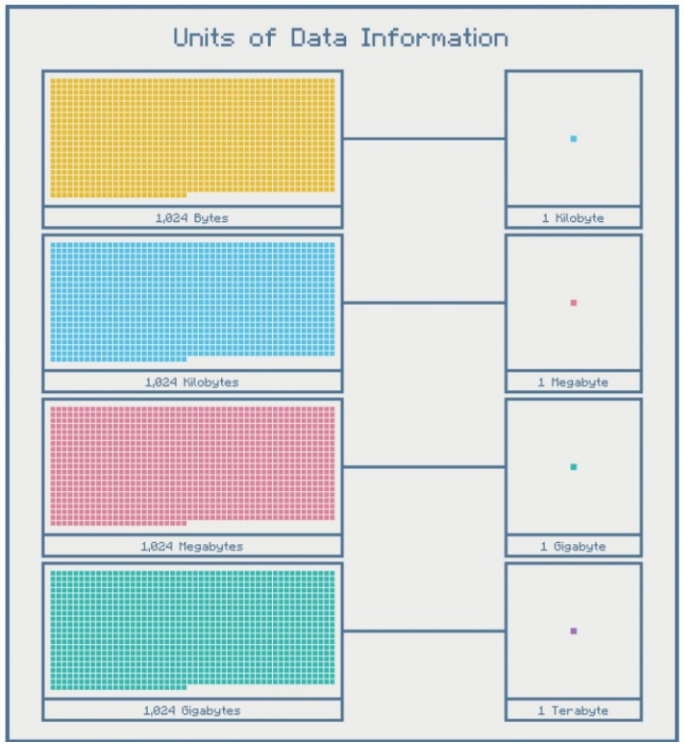
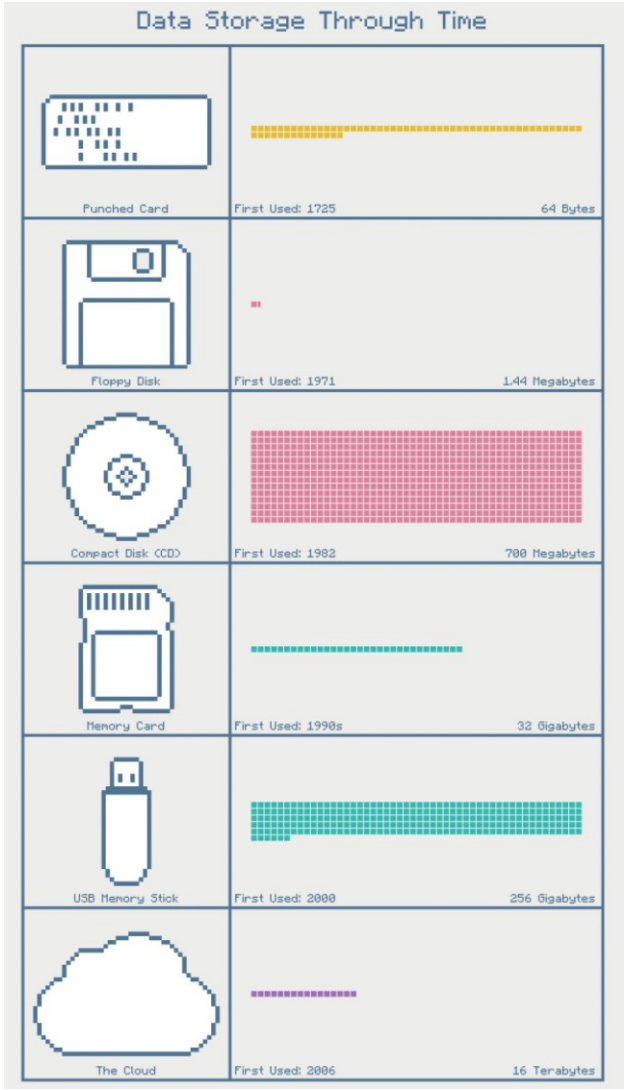
CLOUD DATA STORAGE



DATA STORAGE EVOLUTION





CLOUD DATA STORAGE

2006

2008

2012

2014

CLOUD STORAGE MODELS OVERVIEW

- Attached File Stores
  - A traditional model of organizing data into folders and directories
  - Usually accessed by attaching a virtual disk to a virtual machine
- Object Stores
  - Store unstructured binary objects, referred to as blob (Binary Large Object)
- Databases
  - Structured data collections
  - Three well-known types
    - Relational databases
    - NoSQL databases
    - Graph databases
- Data warehouses
  - Designed to support search over massive amounts of data

ATTACHED FILE STORE

- Good
  - Easy to understand – files are organized around a tree of directories or folders
  - Use standard POSIX (Portable Operating System Interface) API
  - Allow direct use of many existing programs without modification

- Bad
  - Not scalable
    - Limit in file size, number of files, number of folders
    - Slow search when the number of file is large
  - No support for data model

AMAZON FILE STORES

- Amazon **E**lastic **B**lock **S**tore (EBS)
  - Designed to be attached to a single Amazon EC2

- Amazon **E**lastic **F**ile **S**ystem (EFS)
  - General purpose file storage service service
  - A file system interface for one or more Amazon EC2 instances

	File-Stores	Block-Stores
Visibility to OS	OS gets a network share, i.e. sees a directory with files.	OS gets a block device, i.e. sees the volume as a disk
Protocols	NFS (Linux) and CIFS (Windows)	iSCSI/ iSER/ Vendor Specific
Cons	Relatively slow	No built-in file system

GOOGLE ATTACHED FILE STORES

- Persistent Disks
  - Cheapest, up to 64 TB
  - Can be accessed anywhere in a zone.
- Local SSD (solid state disk)
  - More Expensive, and better performance
  - Up to 3 TB
- RAM disk – in memory
  - Most expensive, and fastest
  - Up to 208 GB

AZURE ATTACHED FILE STORES

- Managed Disks
  - Ultra SSD Managed Disks
    - Highest performance
    - Up to 64TB
  - Premium SSD Managed Disks
    - I/O intensive workloads with significantly high throughput and low latency
    - Up to 8TB
  - Standard SSD Managed Disks
    - Entry-level production workloads requiring consistent latency
    - Up to 32TB
  - Standard HDD Managed Disks
    - Cheapest
    - Up to 32TB
- File Share
  - Can be mounted by multiple instances via Server Message Block (SMB) protocol

OBJECT STORES

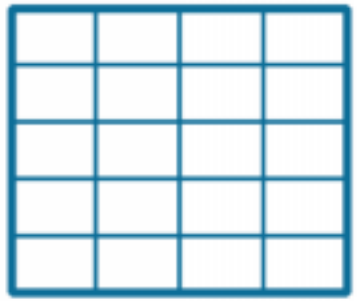
- General term that refers to the way in which we organize and work with units of storage, called objects
- Use information dispersal (erasure coding) algorithms to place object
- Every object contains three things:
  - The data itself
  - An expandable amount of metadata.
  - A globally unique identifier
- Access via API at application-level, rather than via OS at filesystem-level
  - Each object gets a HTTP URL that can be publicly accessible via REST
- Flat object model which support a two-level folder-file hierarchy that allows for the creating of object containers
  - Each can hold zero or more objects
- Objects cannot be modified once created/uploaded
  - They can only be deleted or replaced

FILE SYSTEM VS DATA BASE VS OBJECT STORAGE

File System

```
C:\folder\music.m4a
```

sysadmin required for integrity and scale



Database / Structured Data

```
SELECT * FROM table;
INSERT INTO table;
```

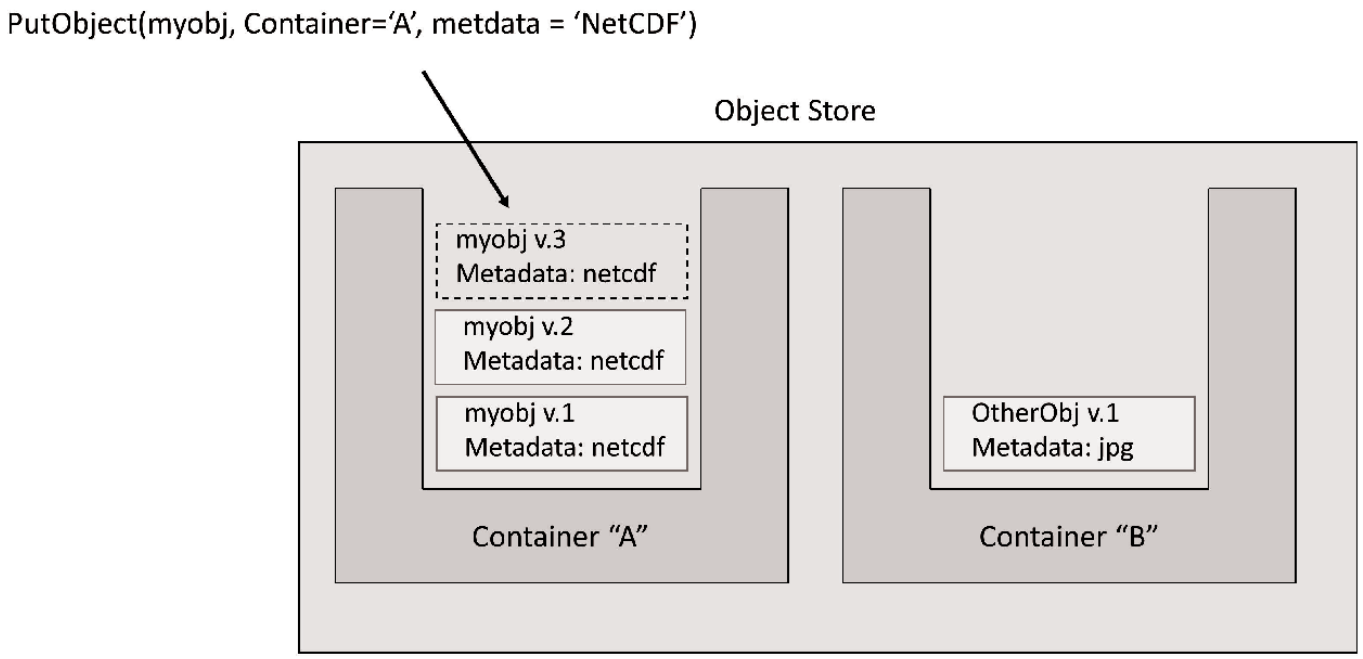
sysadmin and DBA required for scale, integrity and performance

Object Storage

```
GET /object/KbglBn7qepo
PUT /object/KbglBn7qepo
```

sysadmin not required

OBJECT STORES EXAMPLE



Object storage with versioning

- Each NetCDF file is stored in a separate container
- All versions of the same NetCDF file are stared in the same container

OBJECT STORES

- Good
  - Scalable – can grow as large as needed
  - Simplify provisioning – flat namespace with metadata
  - Ease of Use - Each object gets a unique ID and a HTTP URL that can be publicly accessible
  - Agility – sysadmin is not required to maintain it

- Bad
  - No support for search – need to know the object identifier to access, or create a complex metadata index
  - Eventual Consistency - there is no guarantee that a read request returns the most recent version of the data

OBJECT STORES

- Amazon
  - Simple Storage Service (S3)
    - First to come out
    - 2016, reportedly holds trillions of objects in billions of containers (buckets)
  - Glacier
    - Designed for long-term, secure, durable, and low cost data archiving
- Google – three storage tiers
  - Standard – multi-regional
  - Regional
  - Nearline
- Azure
  - Blob as a part of storage account

RELATIONAL DATABASES

- A structured collection of data about entities and their relationships
- Models real-world objects

- Normally managed through a **database management system** (DBMS), such as Oracle, MySQL and PostgreSQL
  - Query via SQL (Structured Query Language)

```
select experiment-id from Experiments, People
where Experiments.person-id = People.person-id
      and People.name = "Smith";
```

  - Support **ACID** semantics
    - Atomicity, Consistency, Isolation, and Durability

RELATIONAL DATABASES

- Good
  - Best for structured data
  - Moderate size of database

- Bad
  - No support for unstructured data
  - Not scalable in the cloud
    - Require a single server to host the whole database

RELATION DATABASE SERVICES IN THE CLOUD

- Running a DBMS, ie. MySQL, on one of the virtual machines
  - Limit in scale
- Relational database services
  - Amazon's Aurora
  - Google's Cloud SQL and Spanner
  - Azure's SQL Database

NOSQL DATABASES

- "Not only SQL or Non SQL" (invented by Johan Oskarsson #nosql for a meet)
  - Most of SQL is supported, but other properties are available
- Opensource software designed to address scalability issues
  - Grow as needed on commodity hardware
- Scheme-less
  - Support unstructured data that are not easily represented in tabular form
  - Key-value store where key and value can be arbitrary value
  - New column can be introduced over time
- Aggregate-oriented - related items stored together for efficiency
- In the cloud, normally the database are distributed over multiple servers
- Not satisfy ACID
  - Eventual consistency

NOSQL DATABASES

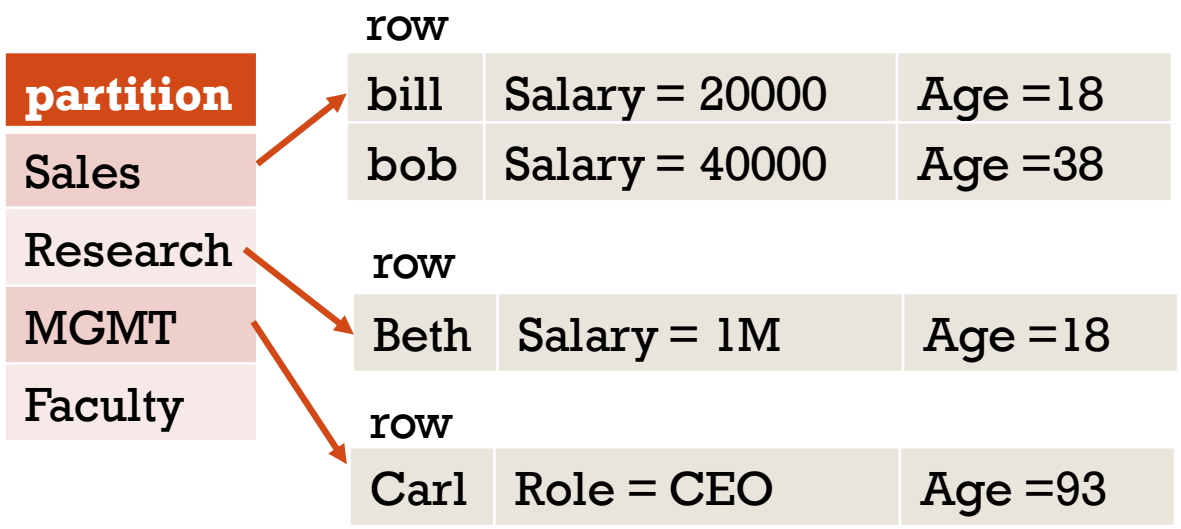
- Amazon's DynamoDB
  - Based on key-value model
    - For each row, the primary key column is the only required attribute
    - Any number of additional columns can be defined, indexed and made searchable
- Google's
  - BigTable
    - The same database behind Google search, analytics, maps, and Gmail
    - Maps two arbitrary strings, row and column key, and a timestamp to arbitrary byte array
    - Designed for sparse and large datasets, and to support large workloads
  - Datastore
    - Similar to BigTable + ACID
- Azure Table
  - Similar to DynamoDB, but quite limited

RELATIONAL VS NOSQL DATABASES

Relational table

Name	Job	Salary	age
Bill	Sales	$20000	18
Beth	Research	$1m	35
Carl	CEO	0	93
Jill	Prof	$100000	24

NoSQL Key-value system



DATA WAREHOUSES

- Data management systems optimized to support analytic queries from reading large data sets
- Designed to support many concurrent requests to read and update

- Example
  - A data warehouse for a medical center that stores clinical data of patients
    - Demographic data
    - Daily detailed information of the visits
    - Medications and treatments
    - Query - "What factors are correlated with the length of a stay?"

DATA WAREHOUSE SERVICES

- Amazon Redshift
- Google BigQuery
- Azure Data Lake

Note that all provide REST (**RE**presentation **S**tate **T**ransfer) API interface