# Dr. D. Y. Patil Institute of Technology, Pimpri, Pune

## Department of Artificial Intelligence & Data Science

# *Unit 2 Data Storage & Cloud Computing*

Prepared By

Prof. Disha Sengupta

# Contents

# Introduction to Enterprise Data Storage

- Enterprise data storage is a hardware solution that manages large volumes of data for organizations, ensuring that data is securely stored, easily accessible, and efficiently managed.
- Enterprise storage is a centralized repository for business information that provides common data management, protection and data sharing functions through connections to computer systems.
- It should be scalable for workloads of hundreds of terabytes or even petabytes without relying on excessive cabling or the creation of subsystems.

- In addition to basic storage functions, enterprise data storage solutions emphasize reliability and data protection.
- They deploy redundancy measures such as RAID (redundant array of independent disks), snapshotting, and replication to safeguard data against hardware failures and data corruption.
- These systems also support data deduplication and compression techniques to maximize storage efficiency and reduce costs.

- Understanding storage system is an important point in building effective storage system. This will yield cost effective, high performance and ease in managing the systems.
- The various types of storage subsystems are:
  - Direct Attached Storage (DAS)
  - Storage Area Network (SAN)
  - Network Attached Storage (NAS)
- DAS is the basic in a storage system and employed in building SAN and NAS either directly or indirectly. NAS is the top most layer, having SAN and DAS as its base. SAN lies between a DAS and a NAS.

# Direct Attached Storage (DAS)

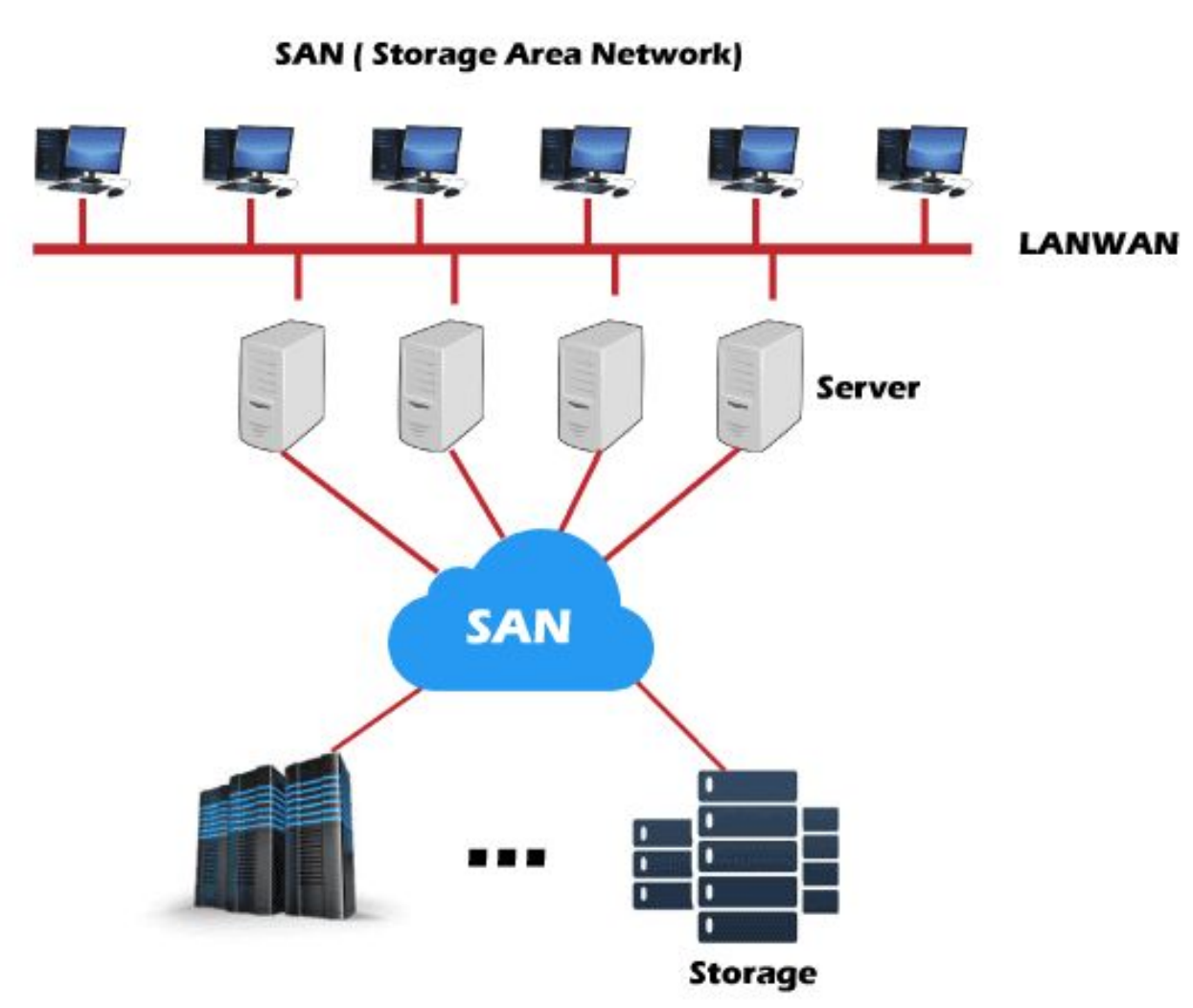
- Direct-attached storage (DAS) is hard disk drives (HDDs) or solid-state drives (SSDs) connected directly inside or outside to a single computer or server that cannot be accessed by other computers or servers.
- Unlike NAS and SAN, DAS is not networked through Ethernet or FC switches.
- Examples include external hard disk drives (HDDs) connected by cable to a desktop or laptop, and solid state drives (SSDs) connected by cable or an M.2 port on the motherboard.
- It is a quick way for users to access storage on a computer, but it's one of the less flexible methods of data storage.

- **Advantage of DAS:**
  - DAS offers low latency and high-speed access.
  - DAS is ideal for localized file sharing in environments with a single server or a few servers.
  - DAS devices can offer block-level access or file-level access.
  - DAS also offers ease of management and administration.
  - DAS can still be used locally to store less critical data.
  - Initial cost of DAS is lower than NAS.
- **Disadvantage of DAS:**
  - DAS is limited in its scalability.
  - DAS is limited to dedicated servers.
  - Unused resources cannot be reallocated easily.
  - If the server the device is attached to it is down for any reason, the data stored on attached DAS device is inaccessible.

# Storage Area Network (SAN)

- **SAN** is a dedicated, independent high-speed network that interconnects and delivers shared pools of storage devices to multiple servers.
- Each server can access shared storage as if it were a drive directly attached to the server.
- A SAN is typically assembled with cabling, host bus adapters, and SAN switches attached to storage arrays and servers. Each switch and storage system on the SAN must be interconnected.
- SANs are typically used to provide high-speed, scalable storage for mission-critical applications, such as databases, email servers, and virtualized environments.

- SANs use specialized hardware and software to provide storage connectivity between servers and storage devices.



## o Types of Storage Area Networks (SAN)

- **Fibre Channel (FC) -** It presents excessive-velocity, low-latency connectivity between servers and storage devices with the use of fibre optic cables.
- **Internet Small Computer System Interface(iSCSI)-** storage protocol that transmits SCSI commands over TCP/IP networks.
- **NVMe over Fabrics (NVMe-oF) -** NVMe over Fabrics extends the NVMe garage protocol over excessive-pace networks, together with Ethernet or Fibre Channel, to offer low latency.
- **Fibre Channel over Ethernet (FCoE) –** It encapsulates Fibre Channel frames into Ethernet packets, allowing Fibre Channel site visitors to be transmitted over Ethernet networks.
- **Serial Attached SCSI(SAS) -** Serial Attached SCSI is a factor-to-point garage protocol designed to attach servers to gadgets using high-pace serial connections.

# Features of SAN

- Centralized Storage
- High-Speed Data  Access
- Scalability
- Redundancy
- High Availability
- Backup and recovery
- Data Protection
- Remote Data Replication

## Advantages of SAN

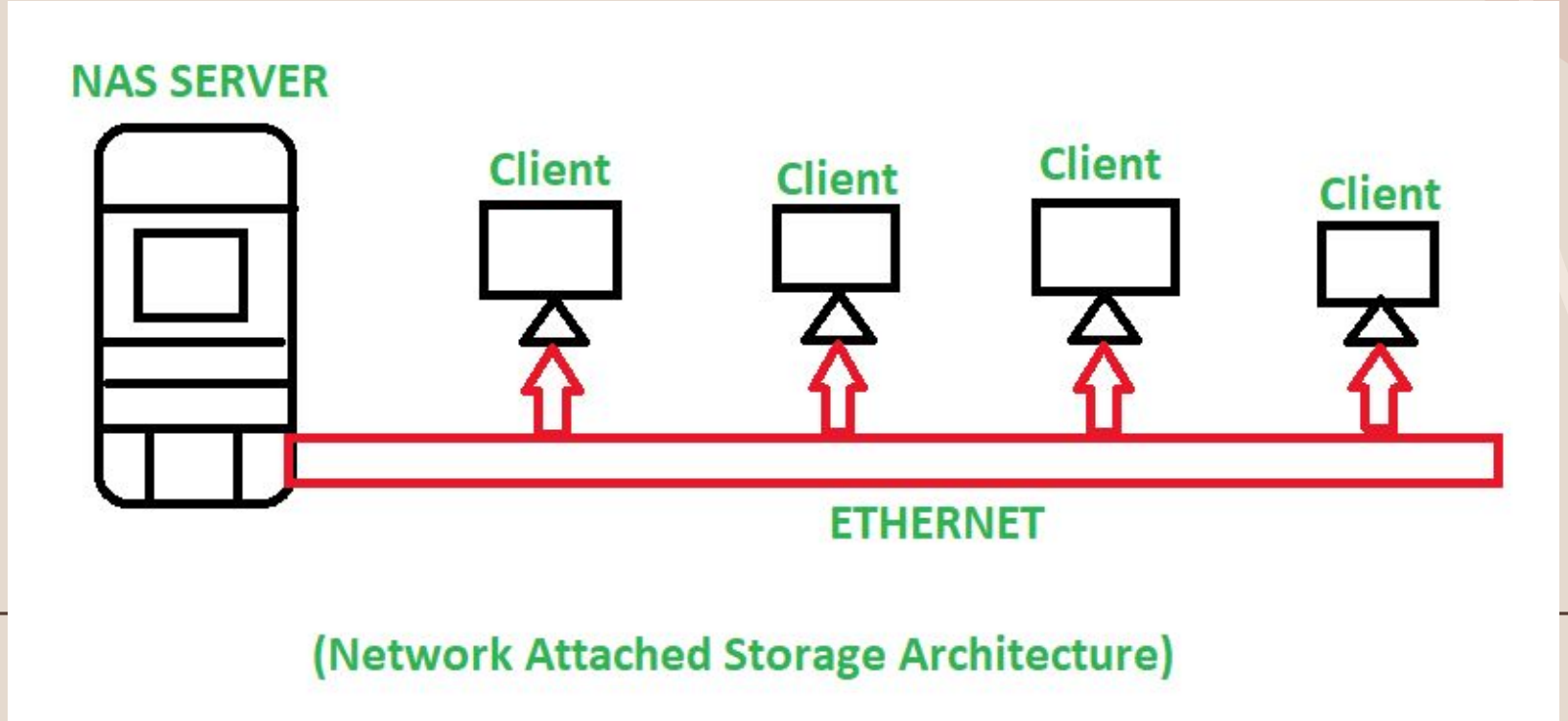
- High performance
- Scalability
- Data protection
- Centralized management

## Disadvantages of SAN

- Complexity
- Cost
- Network dependency
- Security

# Network-attached Storage (NAS)

- NAS is a file storage device which is connected to the network and enables multiple users to access data from the centralized disk capacity. The users on a LAN access the shared storage by the ethernet connection.




(Network Attached Storage Architecture)

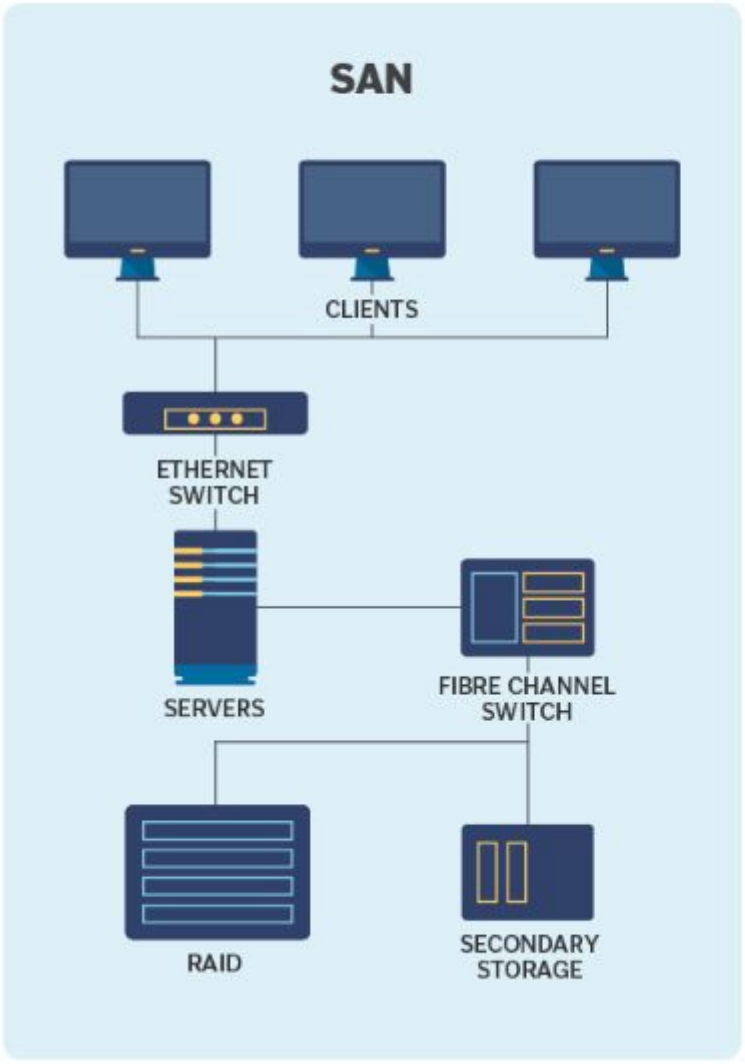
- This storage is fast, low-cost and offers all the advantages of a public cloud on the site. It uses file access protocols such as NFS, SMB, NCP, or AFP.
- It is basically designed for those network systems, which may be processing millions of operations per minute.
- If you have both UNIX and windows users on your network and you want both groups to be able to share files, NAS devices are most suitable. NAS devices can make use of existing directories of user accounts from Windows, Netware or UNIX server.

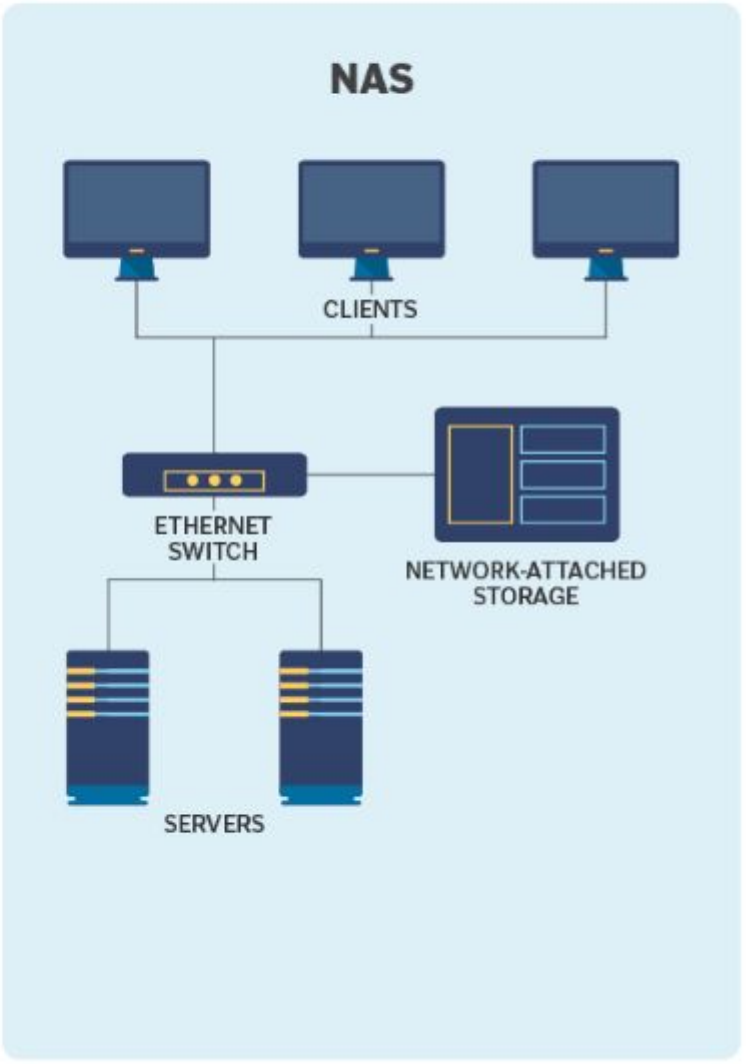
## What is network-attached storage used for?

- The purpose of network-attached storage is to enable users to collaborate and share data more effectively.
- It is useful to distributed teams that need remote access or work in different time zones.
- NAS connects to a wireless router, making it easy for distributed workers to access files from any desktop or mobile device with a network connection.
- Organizations commonly deploy a NAS environment as a storage filer or the foundation for a personal or private cloud.

- Components of a NAS device
  - Physical storage drives
  - Central processing unit (CPU)
  - Operating System
  - Networking Interface
- Basic storage principles of NAS devices
  - File Storage
  - Block storage
  - Object Storage
  - File vs block vs Object storage

# How SAN and NAS compare



# Enterprise Data Storage Trends

1. **Hyper-Converged and Converged Infrastructure -** Both HCI and CI aim to simplify IT operations and improve resource utilization, but HCI offers further integration and ease of use.
2. **NVMe -** Non-volatile memory express (NVMe) is a protocol optimized for high-performance SSDs, offering lower latency and higher throughput compared to traditional storage interfaces. NVMe can significantly reduce data access times, improving overall application performance and user experience.

3. **AI and Analytics Integration -** Enterprise data storage systems are increasingly being designed to support AI and data science initiatives within companies. Modern storage solutions also offer advanced analytics capabilities directly within the storage infrastructure.
4. **Disaggregated and Composable Storage -** Disaggregated storage separates compute and storage resources, allowing each to scale independently. This approach contrasts with traditional, tightly coupled systems where upgrades must be coordinated.

# Key Considerations When Choosing Enterprise Data Storage Solutions

1. **Capacity**
2. **Performance -** Performance metrics like IOPS (input/output operations per second), throughput (GB/s), and latency are crucial when evaluating storage solutions.
3. **Reliability -** Technologies like RAID, erasure coding, and hardware redundancy help mitigate risks associated with hardware failures and data corruption.
4. **Security**
5. **Data Recovery**

# Data Storage Management

- Data storage is expensive; therefore, storage administrators are trying to use tiered storage.
- Using fibre channel for storing data for a network user gives better performance but storage devices used are small and are expensive.
- Today IT organizations are implementing tiered storage as a mix of storage technologies that meet the performance needs and are cost effective.
- Data storage management involves the monitoring of software and hardware assets, such as storage arrays, physical servers, and cloud storage services.

- A tiered approach to data management utilizes different types of storage media to create multiple tiers for accommodating different types of data.
- The exact approach that organizations take to tiering depends on their specific storage, data and application requirements.
- Today's IT teams might support anywhere between two and five tiers, sometimes even more.
- A tiered storage architecture categorizes data hierarchically based on its business value, with data ranked by how often it's accessed by users and applications.
- The data is then assigned to specific storage tiers that are defined by their performance, availability and media costs.

# Data storage tiering hierarchy

How many storage tiers an organization has largely depends on how it classifies data.

| Tier | Data category | Data description | Example storage media |
|---|---|---|---|
| 0 | Mission critical | ▪ Data that supports critical, high-performing workloads that cannot afford delays or disruptions in service<br>▪ Data requirements outweigh storage costs | ▪ NVMe SSDs<br>▪ RAM<br>▪ Storage-class memory (e.g., Optane) |
| 1 | Hot data | ▪ Data that is used continuously to maintain day-to-day business operations<br>▪ Data needs are balanced against storage costs | ▪ SSDs<br>▪ High-performing HDDs<br>▪ Hybrid storage systems |
| 2 | Warm data | ▪ Data that's accessed infrequently or not in constant use but might still be required on occasion<br>▪ Cost considerations are given greater priority | ▪ SATA HDDs<br>▪ Backup appliances<br>▪ Tape storage<br>▪ Cloud storage |
| 3 | Cold data | ▪ Data that is rarely accessed or updated, if at all, or stored only for archival purposes<br>▪ Uses the least expensive storage | ▪ Slow-spinning HDDs<br>▪ Optical discs<br>▪ Tape storage<br>▪ Archival cloud storage |

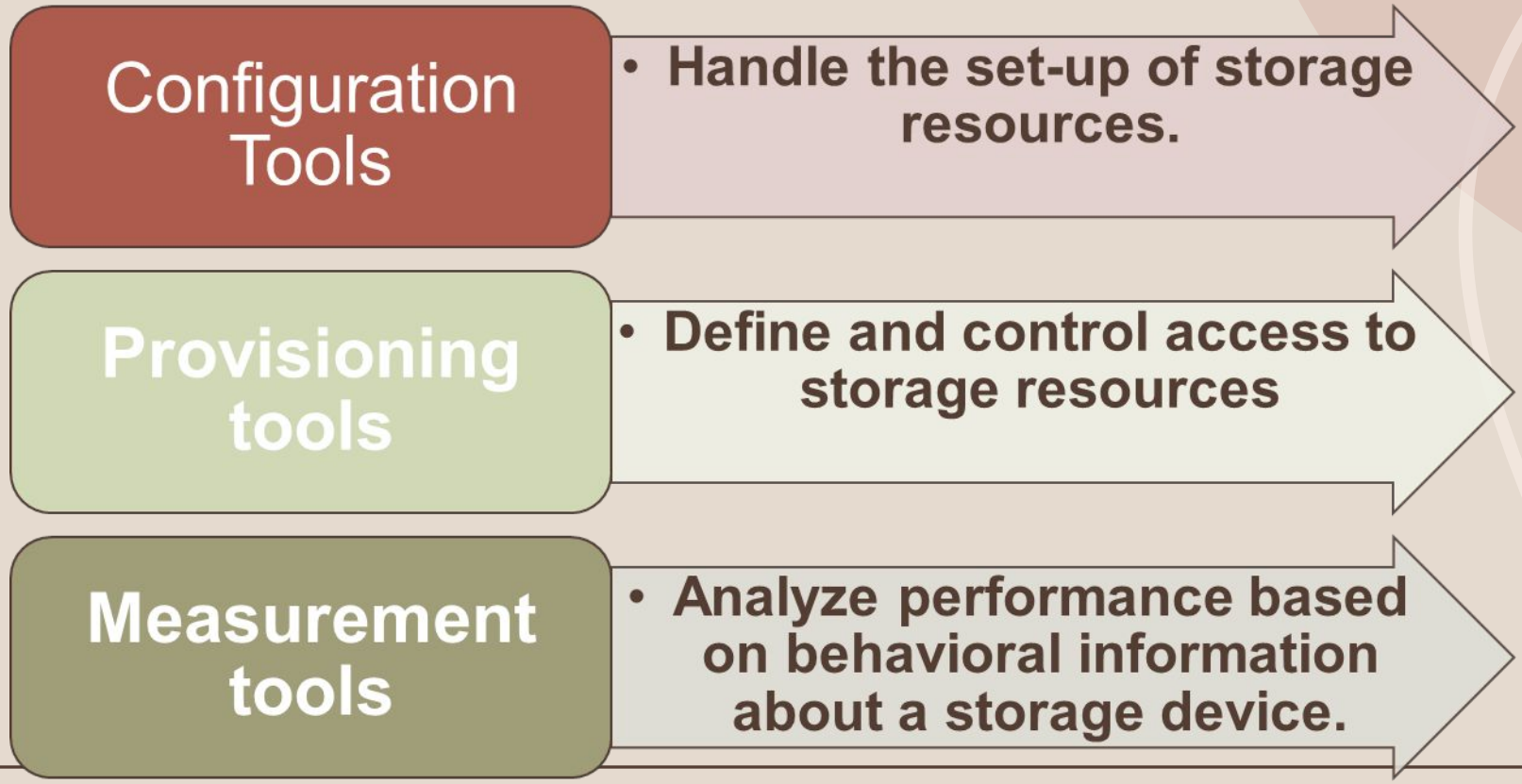
- Data storage management can also include
  - traffic analysis,
  - process automation,
  - memory management,
  - network virtualization,
  - replication, and
  - storage provisioning.
- Using reliable data storage management software, organizations can more easily configure and track storage and report related storage activities.

- **Functions of data storage management are:**

1. **Performance and reliability**
2. **Security and data protection**
3. **Control and Compliance**

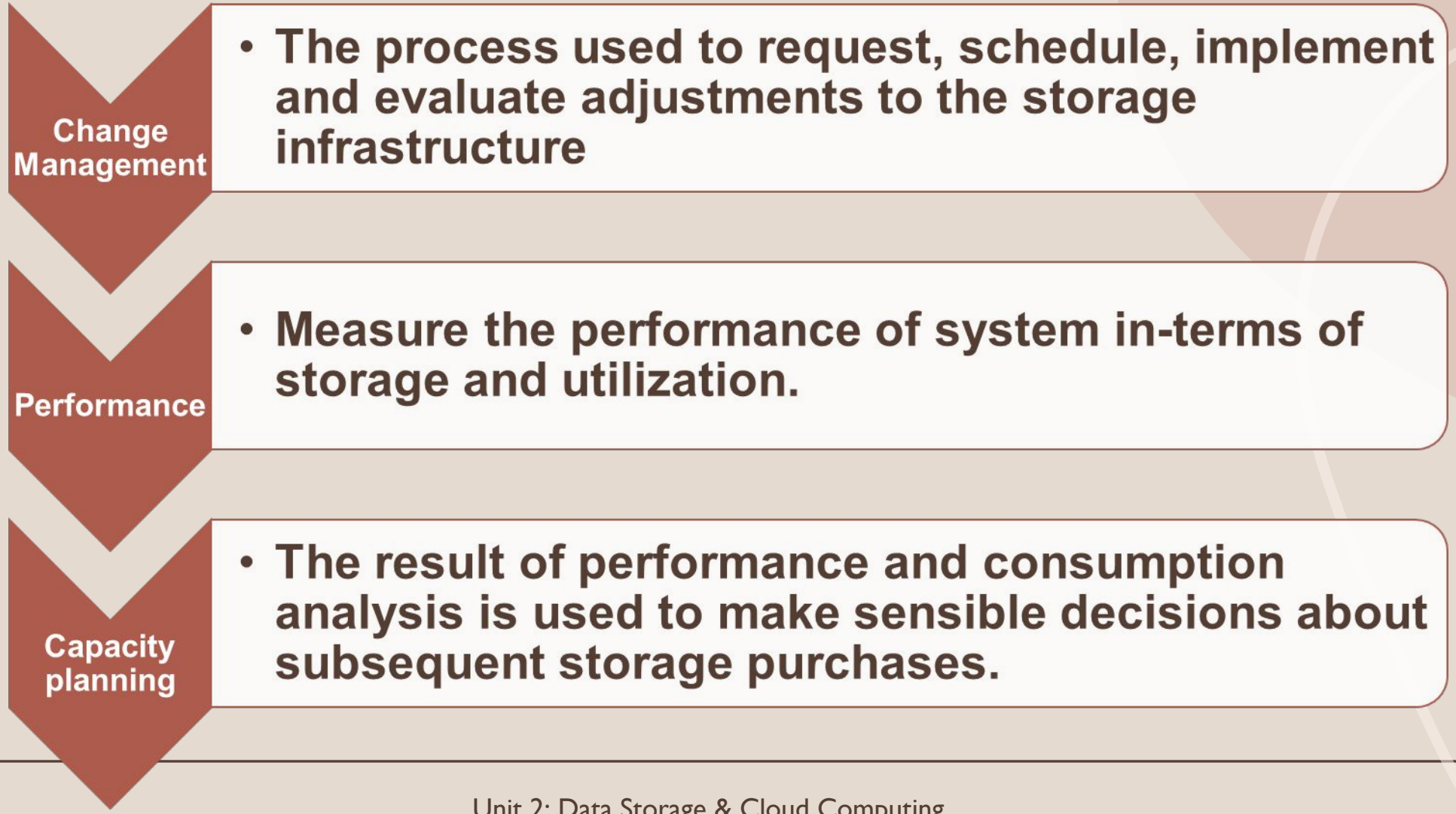
## Data Storage Management Tools:

- Management level tasks are configuration, migration, provisioning, archiving and storage monitoring/reporting.
- Storage Resource Management (SRM) include following tools:

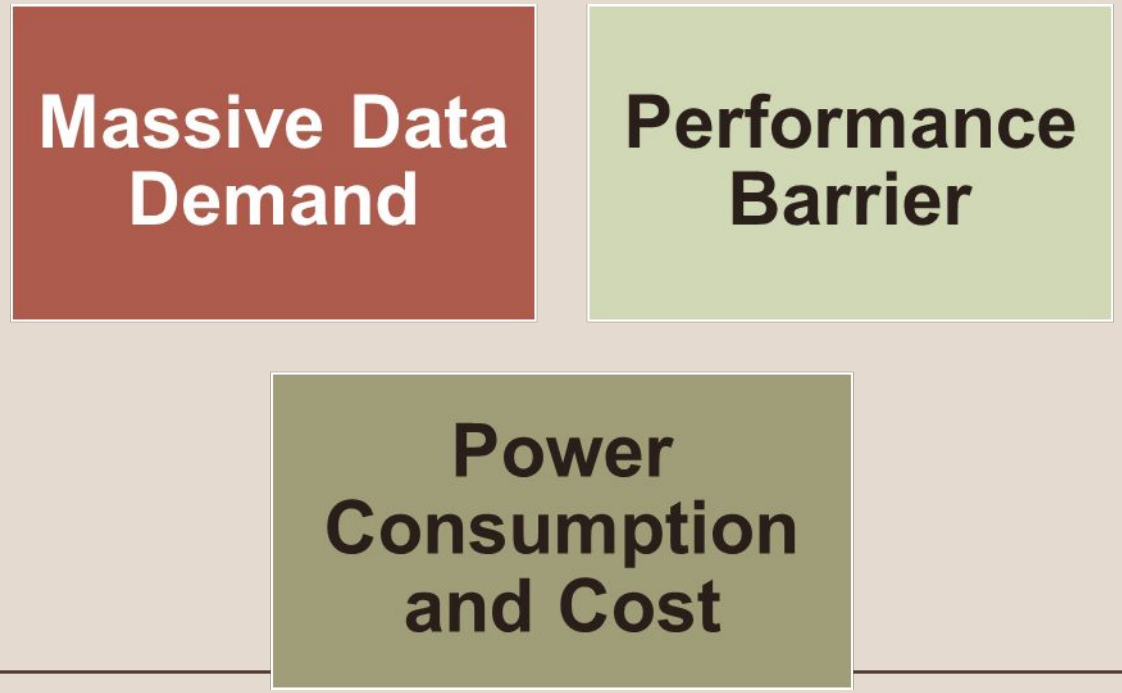


## Storage Management Process:

- Storage management encompasses three areas—change management, performance and capacity planning and tiering (tiered storage).

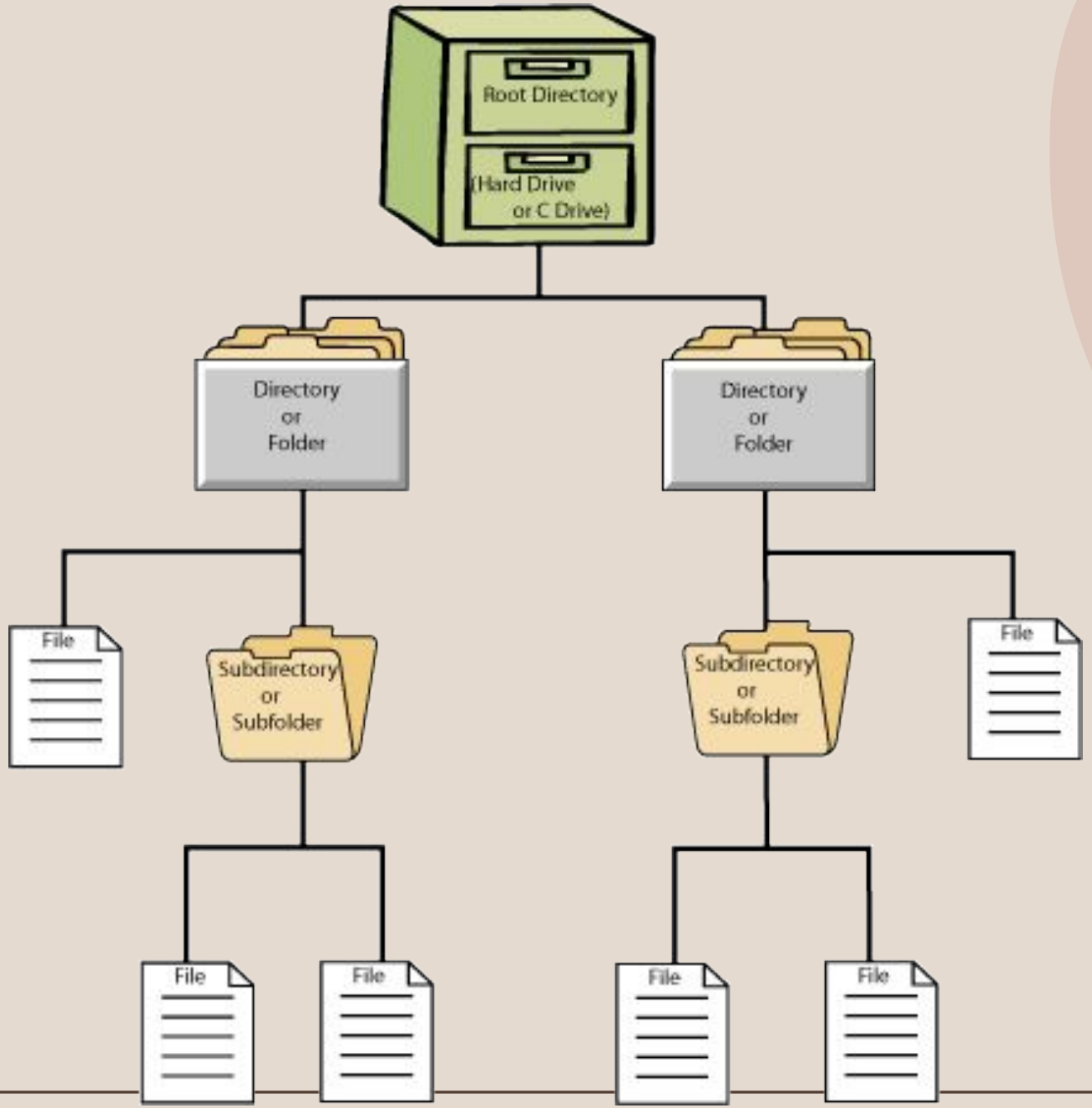


- **Data Storage Challenges are:**
  - Understanding of storage devices will minimize the risks, and an administrator can easily handle challenges like finding out the reason for performance degrading, cost check, etc.
  - Managing traditional storage devices is a complicated task because of high operations cost, performance and scalability issues. Some challenges are:



# File System in Cloud

- File system is an interface between secondary storage device like hard disk and user application.
- The purpose of file systems is to maintain a consistent view of storage so that we can effectively manage it.
- A file system in the cloud is a hierarchical storage system that provides shared access to file data.
- Users can create, delete, modify, read, and write files and can organize them logically in directory trees for intuitive access.


Root Directory
(Hard Drive
or C Drive)
Directory
or
Folder
Directory
or
Folder
File
Subdirectory
or
Subfolder
Subdirectory
or
Subfolder
File
File
File
File
File

- Types of File system
  - **FAT (File Allocation Table):** An older file system used by older versions of Windows and other operating systems. Examples are: FAT16, FAT32
  - **NTFS (New Technology File System):** A modern file system used by Windows. It supports features such as file and folder permissions, compression, and encryption.
  - **ext (Extended File System):** A file system commonly used on Linux and Unix-based operating systems.
  - **HFS (Hierarchical File System):** A file system used by macOS.
  - **APFS (Apple File System):** A new file system introduced by Apple for their Macs and iOS devices.

# Cloud File System

- It is a file system that creates a hub and spoke method of distributing data.
  - The “hub” is the central storage area, typically located at a public cloud provider like Amazon AWS, Microsoft Azure or Google Cloud.
  - The “spokes” are the organization’s local locations (data centers, branch offices, remote offices).
- Considerations are:
  - Must sustain basic file functionality
  - Should be open source
  - Should be grown up enough so that users trust for storing data
  - Should be shared
  - Should be scalable
  - Honest data protection

- A cloud file system should be scalable enough to adopt large organizations file systems under different workloads with good performance requirements.
- Cloud file systems should have high throughputs then local file systems.
- Cloud file system should have minimal operation latency.
- The system should also be scalable to multiple hosts operating in parallel.
- Transparency and backwards compatibility is important to facilitate migration to the cloud with less effort.

- Following are some of the famous cloud file systems:
  - Google File System (GFS)
  - Ghost File System
  - Gluster File System
  - Hadoop File System

- Few lesser known cloud file systems are:
  - XtreemFS: A Distributed and Replicated File System
  - Kosmos File System
  - CloudFS

# 1. Google File System

- Google Inc. developed the Google File System (GFS), a scalable distributed file system (DFS), to meet the company's growing data processing needs.
- GFS is a file system designed to handle batch workloads with lots of data. The system is distributed: multiple machines store copies of every file, and multiple machines try to read/write the same file.
- GFS handle the following issue:
  - Fault Tolerance
  - Large Files
  - Optimize for Reads + Appends
  - High and Consistent Bandwidth

- *Features of GFS*
  - Namespace management and locking.
  - Fault tolerance.
  - Reduced client and master interaction because of large chunk server size.
  - High availability.
  - Critical data replication.
  - Automatic and efficient data recovery.
  - High aggregate throughput.

- *Advantages of GFS*
  - High accessibility Data is still accessible even if a few nodes fail. (replication) Component failures are more common than not, as the saying goes.
  - Excessive throughput. many nodes operating concurrently.
  - Dependable storing. Data that has been corrupted can be found and duplicated.
- *Disadvantages of GFS*
  - Not the best fit for small files.
  - Master may act as a bottleneck.
  - Unable to type at random.
  - Suitable for procedures or data that are written once and only read (appended) later.

# 2. Ghost File System

- Ghost cloud file system is used in Amazon Web Services (AWS).
- GFS (Ghost File System) run over Amazon's S3, EC2 and SimpleDB web services.
- When using GFS, user can have complete control of the data and can be accessed as a standard network disk drive.
- Benefits of Ghost CFS
  - Elastic and cost efficient:
  - Multi-region redundancy:
  - Highly secure:
  - No administration:.
  - Anywhere:

- Features of Ghost CFS
  - Mature elastic file system in the cloud.
  - All files and metadata duplicated across multiple AWS availability regions.
  - WebDav for standard mounting on any Linux, Windows or Mac server or client in the world.
  - FTP access.
  - Web interface for user management and for file upload/download.
  - File name search.
  - Side-loading of fi les from torrent and from URL.

# 3. Gluster File System

- GlusterFS is an open source, distributed file system capable of handling multiple clients and large data.
- GlusterFS gives users the ability to deploy scale-out, virtualized storage, centrally managed pool of storage.
- GlusterFS is based on a stackable user space design and delivers good performance for even heavier workloads.
- Attributes of GlusterFS include scalability and performance, high availability, global namespace, elastic hash algorithm, elastic volume manager, gluster console manager, and standards-based.

# 4. Hadoop File System (HDFS)

- HDFS is an open-source distributed file system developed by the Apache Software Foundation as part of the Hadoop ecosystem.
- It follows a similar master-slave architecture with a single NameNode (Master) and multiple DataNodes (similar to Chunk Servers in GFS).
- Data is divided into fixed-size blocks (typically 128 MB or 256 MB).
- HDFS uses a hierarchical directory structure and provides a standard POSIX-like file system interface.

- **Replication and Fault Tolerance**:
  - HDFS replicates data blocks across DataNodes, with a configurable replication factor (usually three).
  - The NameNode manages metadata and block locations and can handle failover using a standby NameNode.

- **Data Access**:
  - HDFS is designed for both read-heavy and write-heavy workloads and supports a wide range of access patterns.
  - It is the primary file system used in Hadoop for big data processing, including MapReduce and Apache Spark.

- **Consistency Model**:
  - HDFS provides a strong consistency model, ensuring that data consistency is maintained across replicas.

- **Use Case**:
  - HDFS is widely used in various organizations and cloud platforms as a scalable and reliable storage system for big data processing and analytics.

# 5. XtreemFS: A Distributed and Replicated File System

- XtreemFS is a distributed, replicated and open source. XtreemFS allows users to mount and access files via WWW.
- Engaging XtreemFS a user can replicate the files across data centres to reduce network congestion, latency and increase data availability.
- Installing XtreemFS is quite easy, but replicating the files is bit difficult.

# 6. Kosmos File System

- Kosmos Distributed File System (KFS) gives high performance with availability and reliability.
- For example, search engines, data mining, grid computing, etc. It is deployed in C++ using standard system components such as STL, boost libraries, aio, log4cpp.
- KFS is incorporated with Hadoop and Hypertable.

# 7. CloudFS

- CloudFS is a distributed file system to solve problems when file system is itself provided as a service.
- CloudFS is based on GlusterFS, a basic distributed file system, and supported by Red Hat and hosted by Fedora.

# Cloud Data Stores

- A data store is a digital repository that stores and safeguards the information in computer systems.
- A data store can be network-connected storage, distributed cloud storage, a physical hard drive, or virtual storage.
- It can store both structured data like information tables and unstructured data like emails, images, and videos.
- Organizations use data stores to retain, share, and manage information across business units.

- Data stores can be of different types:
  - Relational databases (Examples: MySQL, PostgreSQL, Microsoft SQL Server, Oracle Database)
  - Object-oriented databases
  - Operational data stores
  - Schema-less data stores, e.g. Apache Cassandra or Dynamo
  - Paper files
  - Data files (spread sheets, flat files, etc)

# Types of Data Stores

1. BigTable:
   - BigTable is a compressed, high performance and proprietary data storage system construct on Google File System.
   - Cloud Bigtable is a sparsely populated table that can scale to billions of rows and thousands of columns, enabling you to store terabytes or even petabytes of data.
   - A single value in each row is indexed; this value is known as the row key.
   - It supports high read and write throughput at low latency, and it's an ideal data source for MapReduce operations.
   - BigTable was developed in 2004 and is used in number of Google applications such as web indexing, Google Earth, Google Reader, Google Maps, Google Book Search, MapReduce, Blogger.com, Google Code hosting, Orkut, YouTube and Gmail.

- Advantage for developing BigTable includes scalability and better performance control.
- BigTable charts two random string values (row and column key) and timestamp into an associated random byte array.
  - Row Key:- maintains data in lexicographic order by row key. Each row range is called a tablet, which is the unit of distribution and load balancing.
  - Column Key:- Column keys are grouped into sets called column families. A column family must be created before data can be stored under any column key in that family.
- BigTable is designed to scale into the petabyte range across multiple machines and easy to add more machines and automatically start using resources available without any configuration changes.

- Other similar softwares are as follows:
  - Apache Accumulo: Construct on top of Hadoop, ZooKeeper and economy. Server-side programming mechanism deployed in Java environment.
  - Apache Cassandra: Dynamo's distributed design and BigTable's facts and numbers form adds simultaneously in Apache Cassandra, which uses Java.
  - Hbase: Supports BigTable and Java programming language.
  - Hypertable: Designed for cluster of servers especially for storage and processing.
  - KDI: Kosmix stab to make a BigTable clone and is written in C++.

# 2. DynamoDB

- Amazon DynamoDB is a fully managed NoSQL database service that provides fast and predictable performance with seamless scalability.
- With DynamoDB, you can create database tables that can store and retrieve any amount of data and serve any level of request traffic.
- You can scale up or scale down your tables' throughput capacity without downtime or performance degradation.
- DynamoDB provides on-demand backup capability. It allows you to create full backups of your tables for long-term retention and archival for regulatory compliance needs.

## Advantage of DynamoDB:

- It has fast and predictable performance.
- It is highly scalable.
- It offloads the administrative burden operation and scaling.
- It offers encryption at REST for data protection.
- Its scalability is highly flexible.
- AWS Management Console can be used to monitor resource utilization and performance metrics.
- It provides on-demand backups.
- It enables point-in-time recovery for your Amazon DynamoDB tables. Point-in-time recovery helps protect your tables from accidental write or delete operations. With point-in-time recovery, you can restore that table to any point in time during the last 35 days.
- It can be highly automated.

- **Limitations of DynamoDB –**
  - It has a low read capacity unit of 4kB per second and a write capacity unit of 1KB per second.
  - All tables and global secondary indexes must have a minimum of one read and one write capacity unit.
  - Table sizes have no limits, but accounts have a 256 table limit unless you request a higher cap.
  - Only Five local and twenty global secondary (default quota) indexes per table are permitted.
  - DynamoDB does not prevent the use of reserved words as names.
  - Partition key length and value minimum length sits at 1 byte, and maximum at 2048 bytes, however, DynamoDB places no limit on values.

# Using Grids for Data Storage

- Grid Computing can be defined as a network of computers working together to perform a task that would rather be difficult for a single machine.
- It provides users and applications to use shared pool of resources.
- The compute grid connects computers both desktops and servers and storage across an organization.
- It virtualizes heterogeneous and remotely located components into a single system.


Grid computing allows sharing of computing and data resources for multiple workloads and enables collaboration both within and across organizations.

- Storage for grid computing requires a common file system to present as a single storage space to all workloads.
- Presently grid computing system uses NAS type of storage.
- NAS provides transparency but limits scale and storage management capabilities.

# Grid Oriented Storage (GOS)

- It is a dedicated data storage architecture connected directly to a computational grid.
- It supports and acts as a data bank and reservoirs for data, which can be shared among multiple grid clients.
- GOS is a successor of Network-Attached Storage (NAS) products in the grid computing era.
- A GOS system contains multiple hard disks, arranged into logical, redundant storage containers like traditional file servers.
- GOS-FS can be used as an underlying platform to utilize the available bandwidth and accelerate performance in grid-based applications.

# Cloud Storage

- Cloud storage is a cloud computing model that enables
  - storing data and files on the internet through a cloud computing provider that you access either through the public internet or a dedicated private network connection.
- The provider securely
  - stores,
  - manages, and
  - maintains the storage servers, infrastructure, and network

  to ensure you have access to the data when you need it at virtually unlimited scale, and with elastic capacity.

- Cloud storage removes the need to buy and manage your own data storage infrastructure, giving you agility, scalability, and durability, with anytime, anywhere data access.
- Resources that are exposed to clients are called as **functional interfaces**, that is, data paths.
- Resources maintained by the service providers are called as **management interfaces**, that is, control paths.
- It is important that any provider providing storage as a service should also provide following attributes to the consumer.

- Following are some additional cloud storage attributes:
  - Pay-as-you-use
  - Elasticity
  - Simplicity
  - Resource pooling and multi-tenancy
  - Scalable and elastic
  - Accessible standard protocols
  - Price based on usage
  - Shared and collaborative
  - On-demand self-service
- Cloud storage can be accessible through web-based applications or through web services Application Programming Interfaces (APIs), and using this data are stored.

# Data Management for Cloud Storage

- Data management encompasses
  - acquiring,
  - storing,
  - protecting and
  - processing data
- across an organization or business unit.
- It helps ensure that data is validated and fully accessible to stakeholders when needed.
- With access to large volumes and different data types, organizations invest significantly in data storage and management infrastructure.

- They use data management systems to run business intelligence and data analytics operations more efficiently.
- Some benefits of data management below:
- Organizations require a data management system that is fair, transparent, and confidential while still maintaining accuracy.
- For cloud storage, a standard document is placed by SNIA (Storage Networking Industry Association) Storage Industry Resource Domain Model (SIRDM).
- It states the importance of simplicity for cloud storage.

Figure below shows the SIRDM model which uses CDMI (Cloud Data Management Interface) standards.

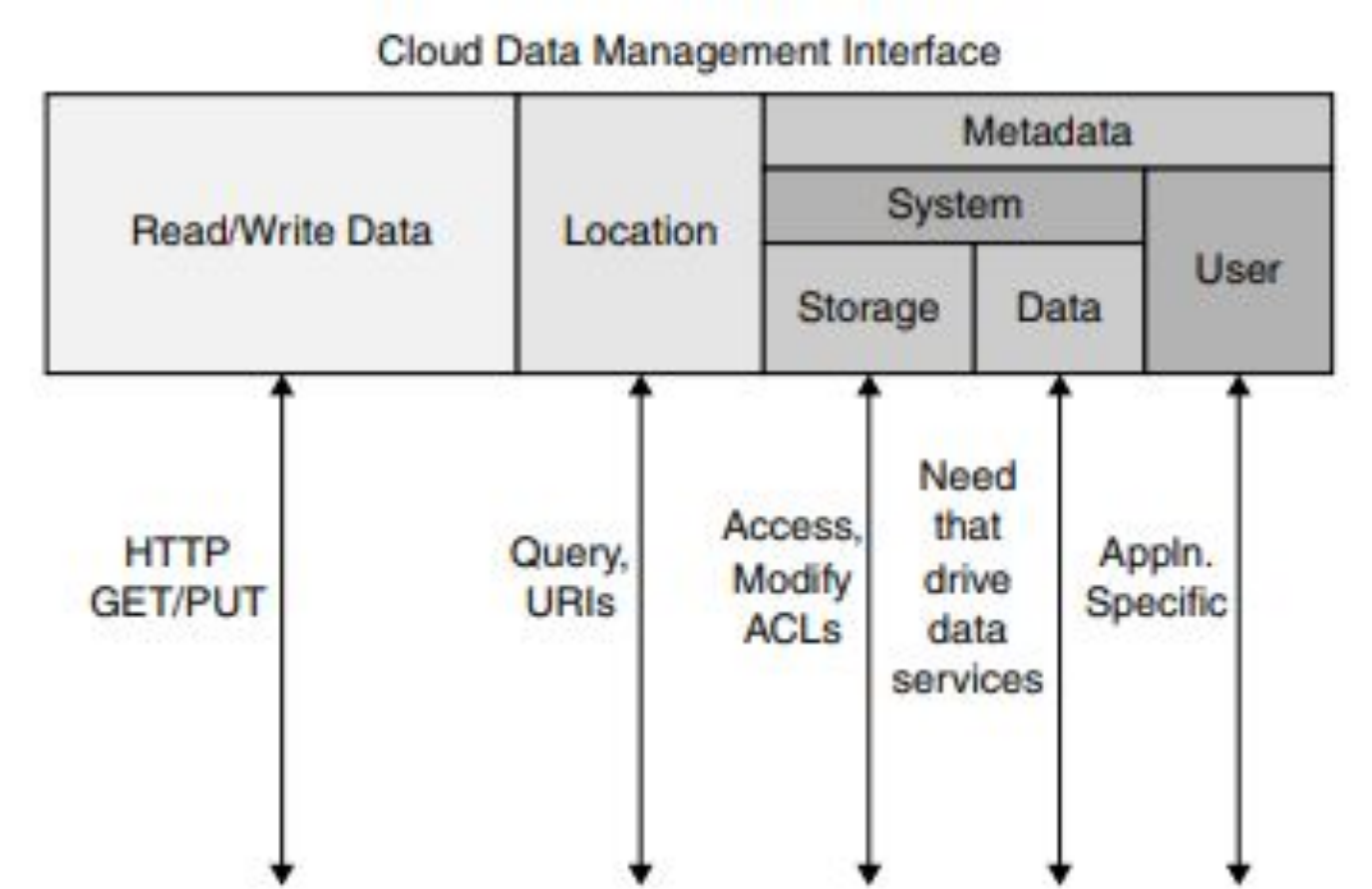



Figure 12.1 Cloud Storage Usage of SIRDM Model

- SIRDM model adopts three metadata: system consisting of storage metadata, data metadata and user metadata.
- By using these metadata, cloud storage interface can offer services without adding unnecessary complexity in managing the data.
  - User metadata is used by the cloud to find the data objects and containers.
  - Storage system metadata is used by the cloud to offer basic storage functions like assigning, modifying and access control.
  - Data system metadata is used by the cloud to offer data as a service based on user requirements and controls the operation based on that data.

# Cloud Data Management Interface (CDMI)

- To create, retrieve, update and delete objects in a cloud the cloud data management interface (CDMI) is used. The functions in CDMI are:
  - Cloud storage offerings are discovered by clients
  - Management of containers and the data
  - Sync metadata with containers an objects'
- CDMI is also used to manage containers, domains, security access and billing information.
- CDMI defines how to manage data and also ways of storing and retrieving it. 'Data path' means how data is stored and retrieved. 'Control path' means how data is managed. CDMI standard supports both data path and control path interface.

# Cloud Storage Requirements

- Multi-tenancy
- Security
- Secure Transmission Channel
- Performance
- Quality of Service (QoS)
- Data Protection & Availability
- Metering and Billing

# Provisioning Cloud Storage

- Cloud provisioning means allocating a cloud service provider's resources to a customer.
- It refers to how a client gets cloud services and resources from a provider. The cloud services that customers can subscribe to include
  - infrastructure-as-a-service (IaaS),
  - software-as-a-service (SaaS), and
  - platform-as-a-service (PaaS) in public or private environments.
- Provisioning in cloud computing involves allocating, configuring, and enabling access to IT resources to address the dynamic needs of an organization.

- Aim of Cloud Provisioning:-
  - To ensure that an organization can seamlessly access the required resources in an optimized and efficient way.
  - Configures various components, such as operating systems, middleware, and applications.
  - Implementation of security initiatives, such as firewalls, threat detection, and encryption, to ensure the safety, confidentiality, and integrity of critical information and data.

# Types of Provisioning

- Provisioning incorporates the procedures and policies involved in sourcing cloud services. The following are the significant types of provisioning in cloud computing.
  1. Server Provisioning
  2. Cloud Provisioning
  3. User Provisioning
  4. Network Provisioning
  5. Service Provisioning

# The Benefits of Cloud Provisioning

- Scalability
- Costs Savings
  - There are fewer employees required, leading to more cost-cutting.
  - The duty of upgrading cloud provisioning software and hardware falls to the cloud service providers.
  - With limited IT hardware around, organizations consume less energy.
  - Employees execute projects faster and spend more of their company time working on productive things.

# Challenges of Cloud Provisioning

- Complex Management and Monitoring
- Service Enforcement
- Service and Resource Dependencies
- Cost Controls

# Data-Intensive Technologies for Cloud Computing

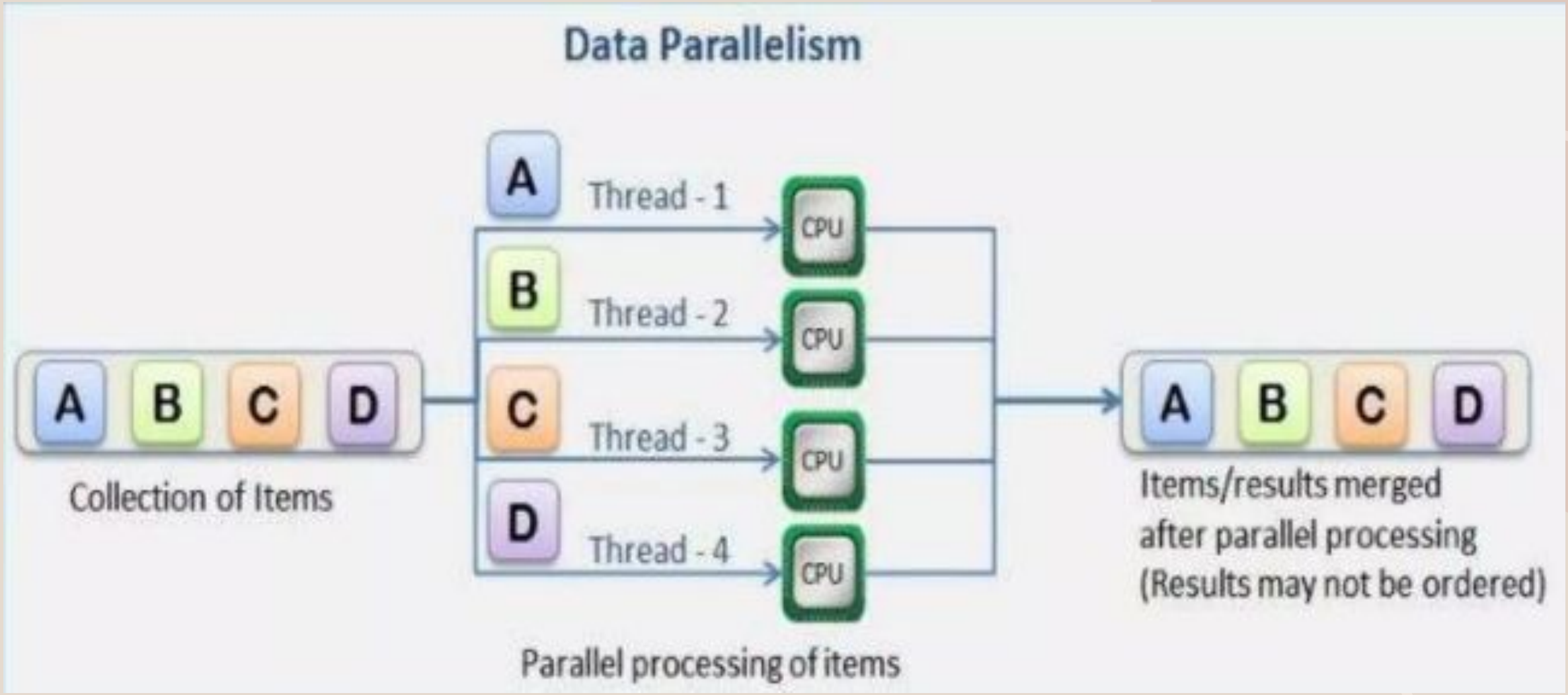
- Data-intensive computing is concerned with production, manipulation, and analysis of large-scale data in the range of hundreds of megabytes (MB) to petabytes (PB) and beyond.
- The storing, managing, accessing, and processing of this vast amount of data represents a fundamental need and an immense challenge in order to satisfy needs to search, analyze, mine, and visualize this data as information.
- Data-intensive computing is intended to address this need.

- Parallel processing approaches can be generally classified as either **compute-intensive**, or **data-intensive**.
- Compute-intensive is used to describe application programs that are compute-bound. Such applications devote most of their execution time to computational requirements as opposed to I/O, and typically require small volumes of data.
- While Parallel processing of data-intensive applications typically involves parallelizing individual algorithms within an application process, and decomposing the overall application process into separate tasks, which can then be executed in parallel on an appropriate computing platform to achieve overall higher performance than serial processing.

# Processing Approach

- Data-intensive is used to describe applications that are I/O bound or with a need to process large volumes of data.
- Such applications devote most of their processing time to I/O and movement and manipulation of data.
- Parallel processing of data-intensive applications typically involves partitioning or subdividing the data into multiple segments which can be processed independently using the same executable application program in parallel on an appropriate computing platform, then reassembling the results to produce the completed output data.

- The fundamental challenges for data-intensive computing are managing and processing exponentially growing data volumes, significantly reducing associated data analysis cycles to support practical, timely applications, and developing new algorithms which can scale to search and process massive amounts of data.


Data Parallelism
A
B
C
D
Thread - 1
Thread - 2
Thread - 3
Thread - 4
CPU
CPU
CPU
CPU
A B C D
Collection of Items
A B C D
Items/results merged
after parallel processing
(Results may not be ordered)
Parallel processing of items

# Characteristics of Data Intensive Computing

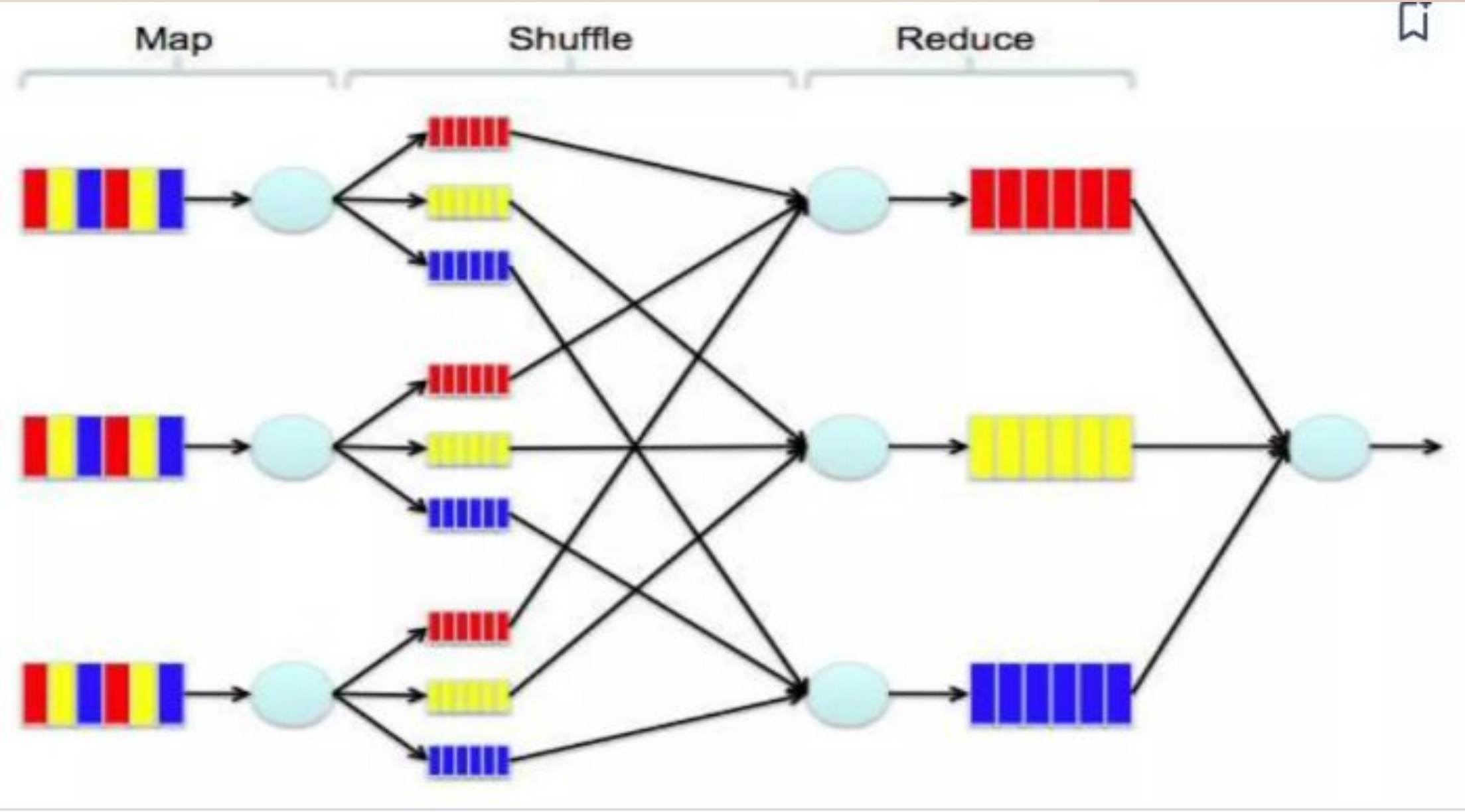
- Several common characteristics of data-intensive computing systems distinguish them from other forms of computing:
  1. To achieve high performance in data-intensive computing, it is important to minimize the movement of data.
  2. It utilize a machine-independent approach in which applications are expressed in terms of high-level operations on data, and the runtime system transparently controls the scheduling, execution, load balancing, communications, and movement of programs and data across the distributed computing cluster.
  3. A focus on reliability and availability.
  4. Data-intensive computing systems can typically be scaled in a linear fashion to accommodate virtually any amount of data, or to meet time-critical performance requirements

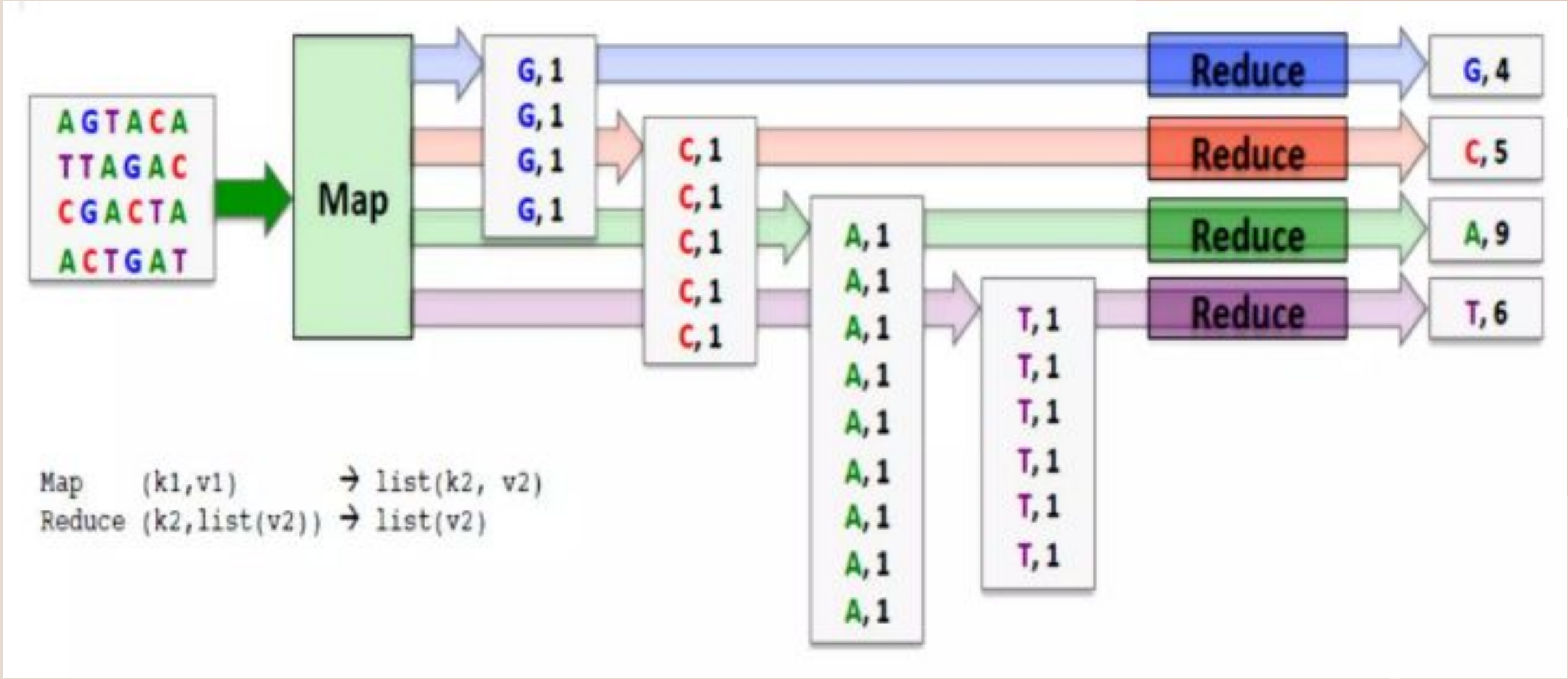
# System Architecture of Data Intensive Computing

- A variety of system architectures have been implemented for data-intensive computing and large-scale data analysis applications.
- However, most data growth is with data in unstructured form and new processing paradigms with more flexible data models were needed.
- Several solutions have emerged including:
  - MapReduce
  - Hadoop
  - HPCC

# 1. MapReduce

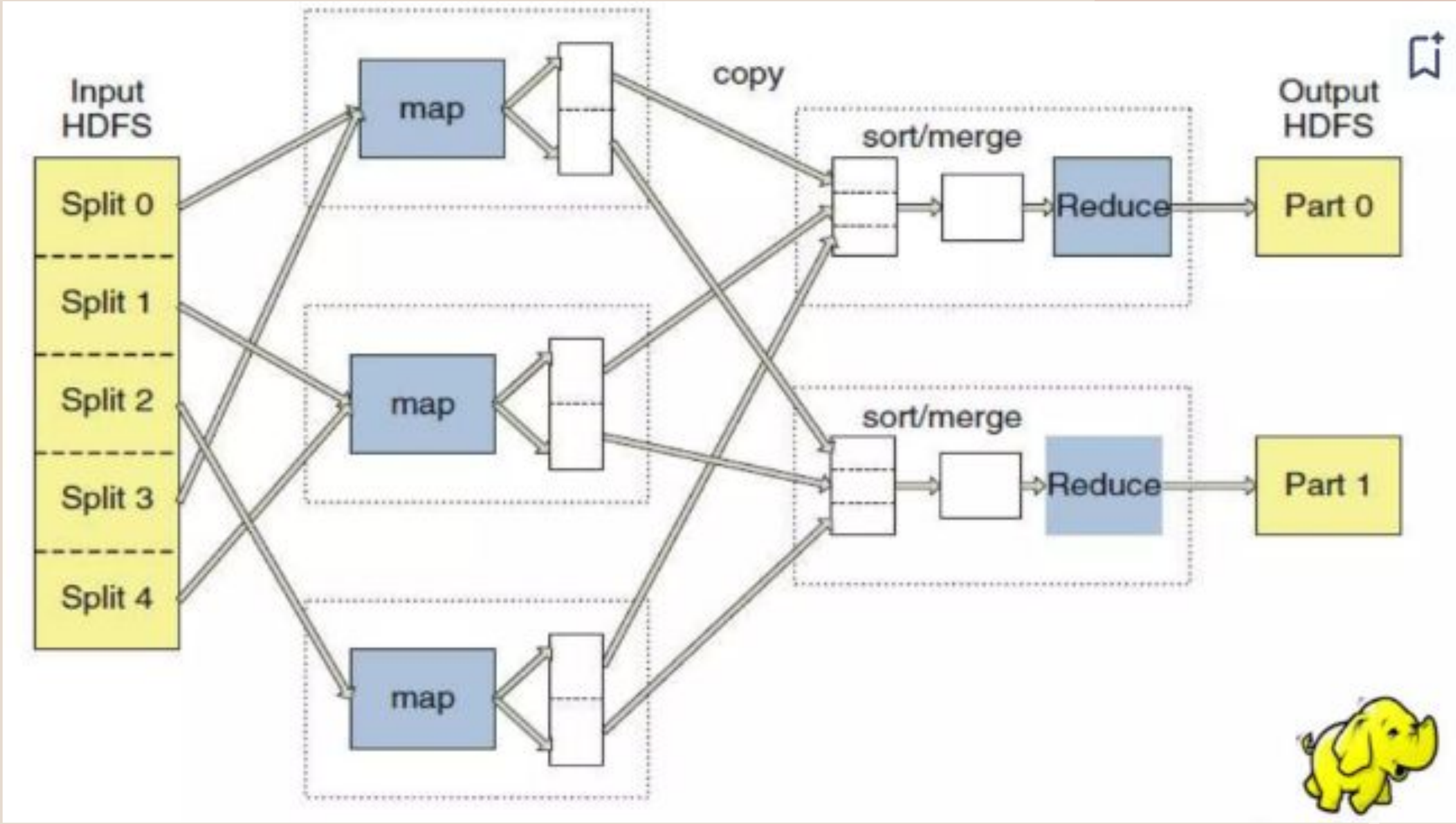
- It is pioneered by Google which an example of a modern systems architecture designed for data-intensive computing.
- The MapReduce architecture allows programmers to use a functional programming style to create a map function that processes a key–value pair The MapReduce architecture allows programmers to use a functional programming style to create a map function that processes a key–value pair associated with the input data to generate a set of intermediate key–value pairs, and a reduce function that merges all intermediate values associated with the same intermediate key.
- It takes a set of input key–value pairs associated with the input data and produces a set of output key–value pairs.


Map
Shuffle
Reduce


AGTACA
TTAGAC
CGACTA
ACTGAT
Map
G, 1
G, 1
G, 1
G, 1
C, 1
C, 1
C, 1
C, 1
C, 1
A, 1
A, 1
A, 1
A, 1
A, 1
A, 1
A, 1
A, 1
A, 1
T, 1
T, 1
T, 1
T, 1
T, 1
T, 1
Reduce
Reduce
Reduce
Reduce
G, 4
C, 5
A, 9
T, 6
Map (k1,v1) → list(k2, v2)
Reduce (k2,list(v2)) → list(v2)

# 2. Hadoop

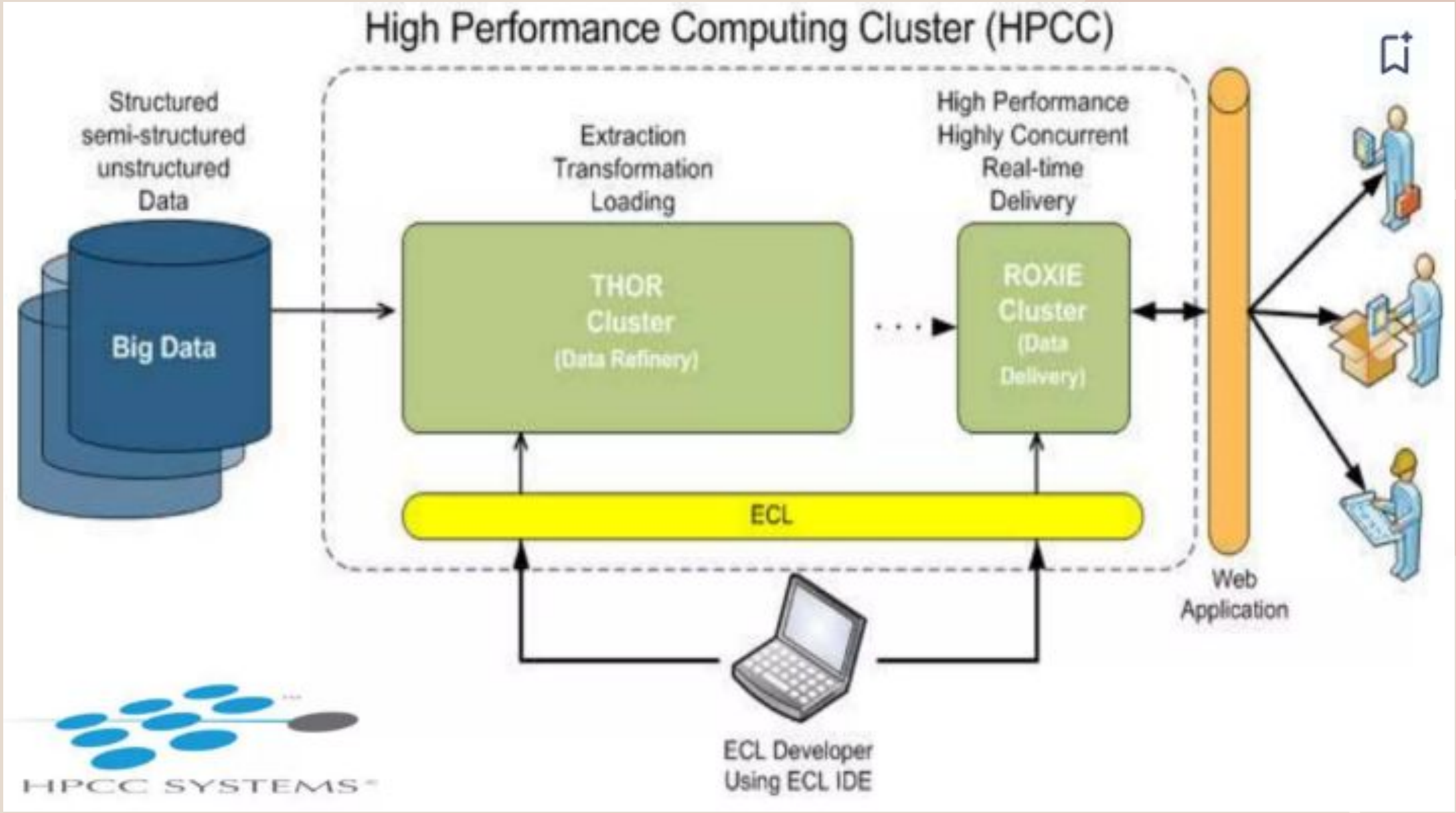
- Apache Hadoop is an open source software project sponsored by The Apache Software Foundation which implements the MapReduce architecture.
- The Hadoop execution environment supports additional distributed data processing capabilities which are designed to run using the Hadoop MapReduce architecture.
- These include
  - Hbase (a column oriented database)
  - Hive (a data warehouse built on top of Hadoop that provides SQL-like query)
  - Pig (a high-level data-flow programming language and execution framework for data-intensive computing)


Input
HDFS
Split 0
Split 1
Split 2
Split 3
Split 4
map
map
map
copy
sort/merge
sort/merge
Reduce
Reduce
Output
HDFS
Part 0
Part 1

# 3. HPCC

- HPCC HPCC (High-Performance Computing Cluster) was developed and implemented by LexisNexis Risk Solutions.
- The HPCC approach also utilizes commodity clusters of hardware running the Linux operating system.
- Custom system software and middleware components were developed and layered on the base Linux operating system to provide the execution environment and distributed filesystem support required for data-intensive computing.
- A new high-level language for data-intensive computing called ECL is also implemented by LexisNexis.

- The ECL programming language The ECL programming language is a high-level, declarative, data-centric, implicitly parallel language that allows the programmer to define what the data processing result should be and the data flows and transformations that are necessary to achieve the result.
- It combines data representation with algorithm implementation, and is the fusion of both a query language and a parallel data processing language.
- The ECL language includes extensive capabilities for data definition, filtering, data management, and data transformation, and provides an extensive set of built-in functions to operate on records in datasets which can include user-defined transformation functions.


High Performance Computing Cluster (HPCC)
Structured
semi-structured
unstructured
Data
Big Data
Extraction
Transformation
Loading
THOR
Cluster
(Data Refinery)
High Performance
Highly Concurrent
Real-time
Delivery
ROXIE
Cluster
(Data
Delivery)
ECL
Web
Application
ECL Developer
Using ECL IDE
HPCC SYSTEMS

# Cloud Storage from LANs to WANs

- Instead of owning, establishing and managing the database programs, cloud computing vendors normally maintain little more than the hardware and give their clients a set of virtual appliances to establish their own software.
- Resource accessibility is normally elastic, with an apparently infinite allowance of compute power and storage accessible on demand, in a pay-only-for-what-you-use model.

# Cloud Characteristics

- There are three characteristics of a cloud computing natural environment that are most pertinent to be considered before choosing storage in cloud.

1. Computer power is elastic, when it can perform parallel operations.
2. Data is retained at an unknown host server.
3. Data is duplicated often over distant locations.

# Distributed Data Storage

- Distributed storage means are evolving from the existing practices of data storage for the new generation of WWW applications through organizations like Google, Amazon and Yahoo.
- There are some reasons for distributed storage means to be favoured over traditional relational database systems encompassing scalability, accessibility and performance.
- The new generation of applications require processing of data to a tune of terabytes and even peta bytes. This is accomplished by distributed services.

- Distributed services means distributed data. This is a distinct giant compared to traditional relational database systems.
- Emerging distributed data storage are
  - Amazon Dynamo,
  - CouchDB and
  - ThruDB.

# 1. Amazon Dynamo

- Amazon Dynamo is a widely used key-value store. It is one of the main components of Amazon. com, the biggest e-commerce stores in the world.
- Amazon DynamoDB is a fully managed NoSQL database service that allows to create database tables that can store and retrieve any amount of data. It automatically manages the data traffic of tables over multiple servers and maintains performance.
- It also relieves the customers from the burden of operating and scaling a distributed database. Hence, hardware provisioning, setup, configuration, replication, software patching, cluster scaling, etc. is managed by Amazon.

- With DynamoDB, you can create database tables that can store and retrieve any amount of data and serve any level of request traffic.
- DynamoDB provides on-demand backup capability.
- DynamoDB allows you to delete expired items from tables automatically to help you reduce storage usage and the cost of storing data that is no longer relevant.
- Benefits of Amazon DynamoDB are:
  - Managed service
  - Scalable
  - Fast
  - Durable and highly available
  - Flexible
  - Cost-effective

# 2. CouchDB

- Apache CouchDB is an open source NoSQL document database that collects and stores data in JSON-based document formats.
- Unlike relational databases, CouchDB uses a schema-free data model, which simplifies record management across various computing devices, mobile phones, and web browsers.
- It is an open-source database that uses various different formats and protocols to store, transfer, and process its data.
- It uses JSON to store data, JavaScript as its query language using MapReduce, and HTTP for an API.
- Documents are the primary unit of data in CouchDB and they also include metadata.

- CouchDB aspires to persuade the Four Pillars of Data Management by these methods:
  1. Save: ACID compliant, save efficiently
  2. See: Easy retrieval, straightforward describing procedures, fulltext search
  3. Secure: Strong compartmentalization, ACL, connections over SSL
  4. Share: Distributed means
- Features of CouchDB
  - Replication
  - Document Storage
  - Security
  - MapReduce
  - Authentication
  - Built for Offline
  - HTTP API

- *Advantages of CouchDB*
  - HTTP API is used for easy Communication.
  - It is used to store any type of data.
  - ReduceMap allows optimizing the combining of data.
  - Structure of CouchDB is very simple
  - Fast indexing and retrieval.
- *Disadvantages of CouchDB*
  - CouchDB takes a large space for overhead, which is a major disadvantage as compared to other databases.
  - Arbitrary queries are expensive.
  - There's a bit of extra space overhead with CouchDB compared to most alternatives.
  - Temporary views on huge datasets are very slow.
  - It doesn't support transactions
  - Replication of large databases may fail.

# 3. ThruDB

- ThruDB is an open source database built on Apache's Thrift framework and is a set of simple services such as scaling, indexing and storage which is used for building and scaling websites.
- It provides flexible, fast and easy-to-use services that simplify the management of the modern web data layer and provides developers with features and tools most web developers need.
- These features can be easily configured or turned off.
- ThruDB aspires to be universal in simplifying the administration of the up-to-date WWW data level (indexing, caching, replication, backup) by supplying a reliable set of services.

- *Features*
  - Multi-master replication
  - Built for horizontal scalability
  - Incremental backups and redo logging
  - Multiple storage back-end client libraries for most languages
  - Simple and powerful search API
- *Services*

  ThruDB provides web-scale data management by providing these services:
  - *Thrucene* - For Lucene-based indexing
  - *Throxy* - For partitioning and load balancing
  - *Thrudoc* - For document storage
  - *Thruqueue* - For a persistent message queue service
  - *Thrift* - For cross-language services framework

# Thank you