

## Why Data Lake?

Why Data Warehouse is failing today?



# Once upon a time ☺



Total Capacity – 1.44 MB

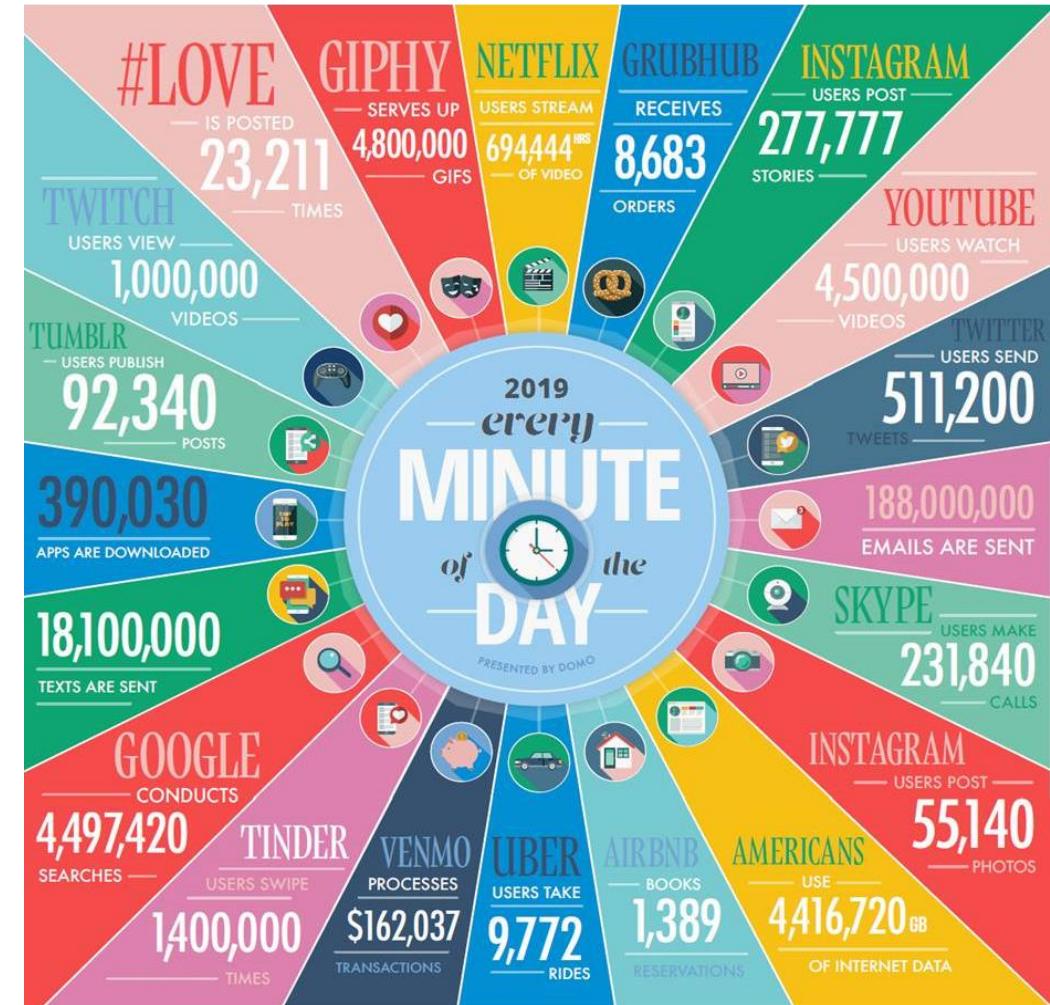
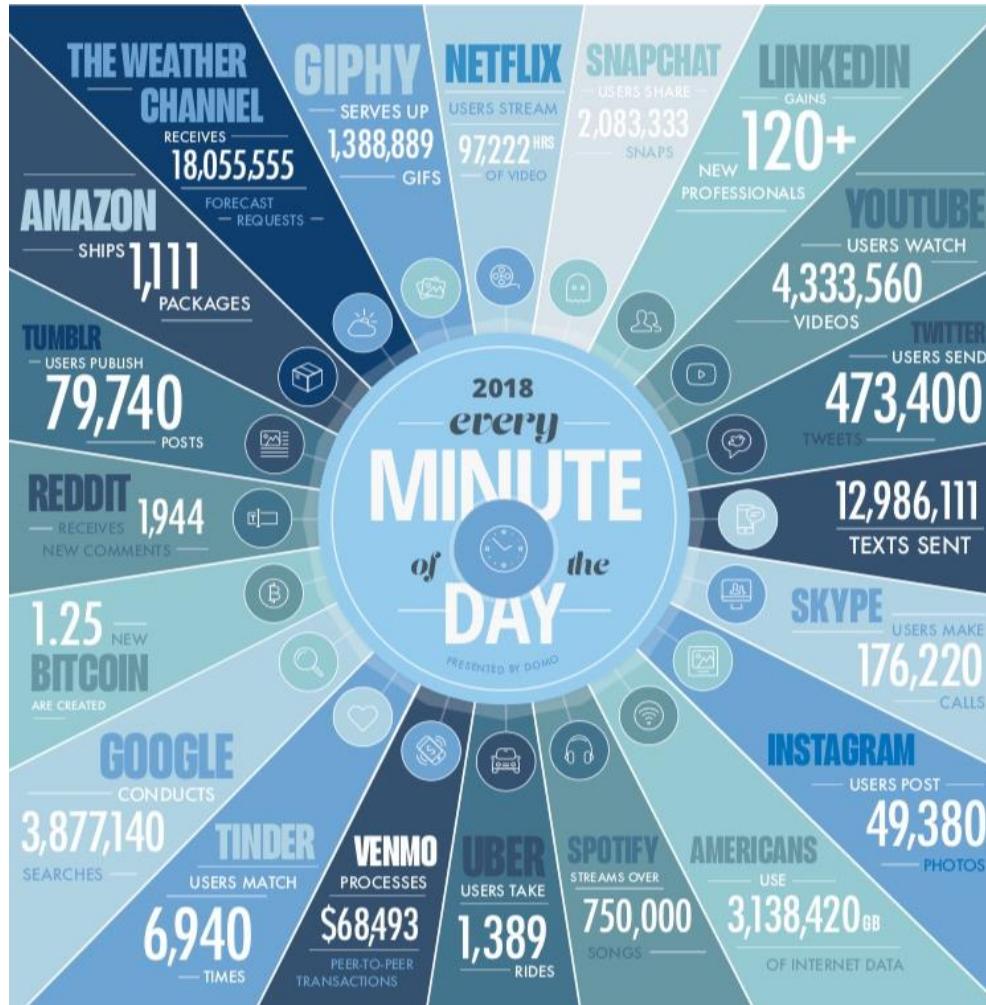
1990

*By 2020, it's estimated that 1.7MB of data will be created every second for every person on earth.“*

— Domo report (6<sup>th</sup> edition)

2020

# How much data is generated every min?



**EVERY DAY WE CREATE  
2,500,000,  
000,000,  
000,000**

**(2.5 QUINTILLION) BYTES OF DATA**

*This would fill 10 million blu-ray discs,  
the height of which stacked, would measure  
the height of 4 Eiffel Towers on top of one another.*



**1 Quintillion = 1 million terabyte**

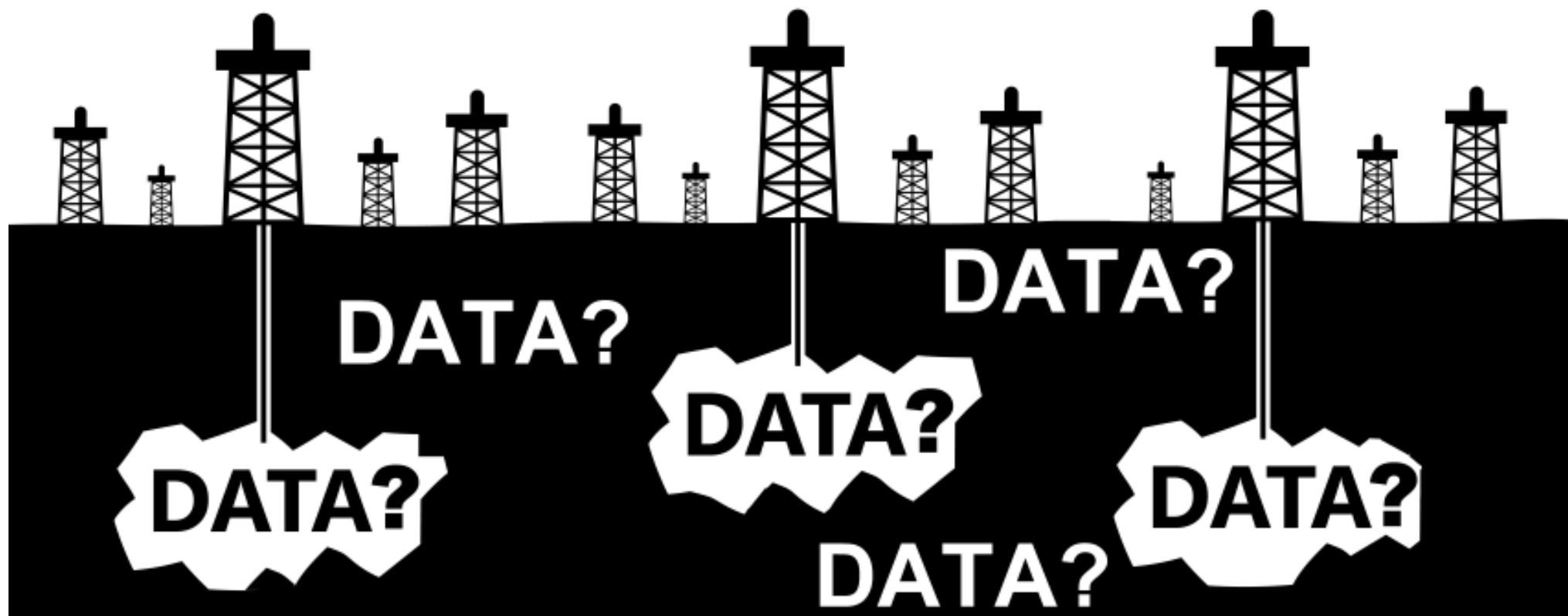


**LearnCloud.Info**

# Data is new oil

We need to find it, extract it, refine it, distribute it and monetize it

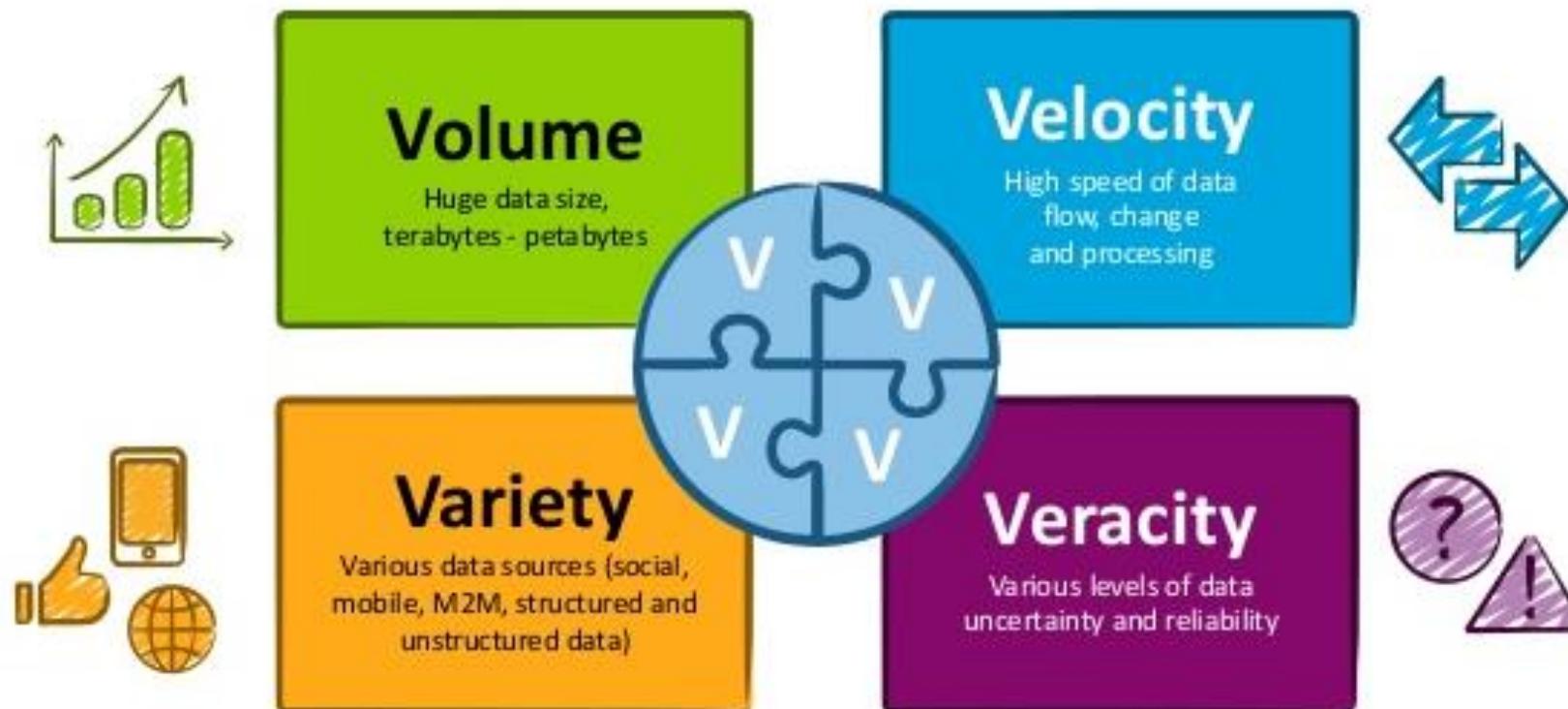
– David Buckingham, Big data expert

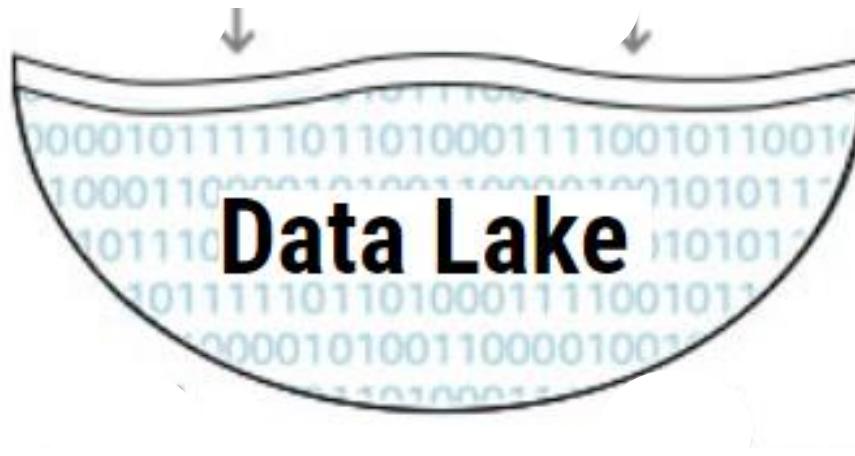




# Problem statement

- Need a solution which can handle below 4 V's of data.





Data Lake is a big container to store data.

# Data Lake Sources

Web logs, JSON, XML, csv



Applications



Traditional databases



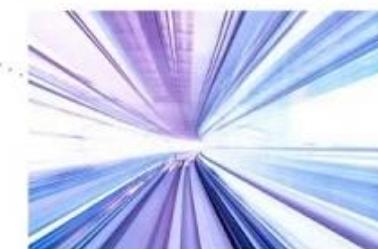
Data Lake

Data Lake

Sensor data, social media



Streaming data



# What is Data Lake?

“If you think of a DataMart as a store of bottled water – clean and packaged and structured for easy consumption – the data lake is a large body of water in a more natural state. The contents of the data lake stream in from a source to fill the lake, and various users of the lake can come to examine, dive in, or take samples.”



James Dixon  
CTO, Pentaho



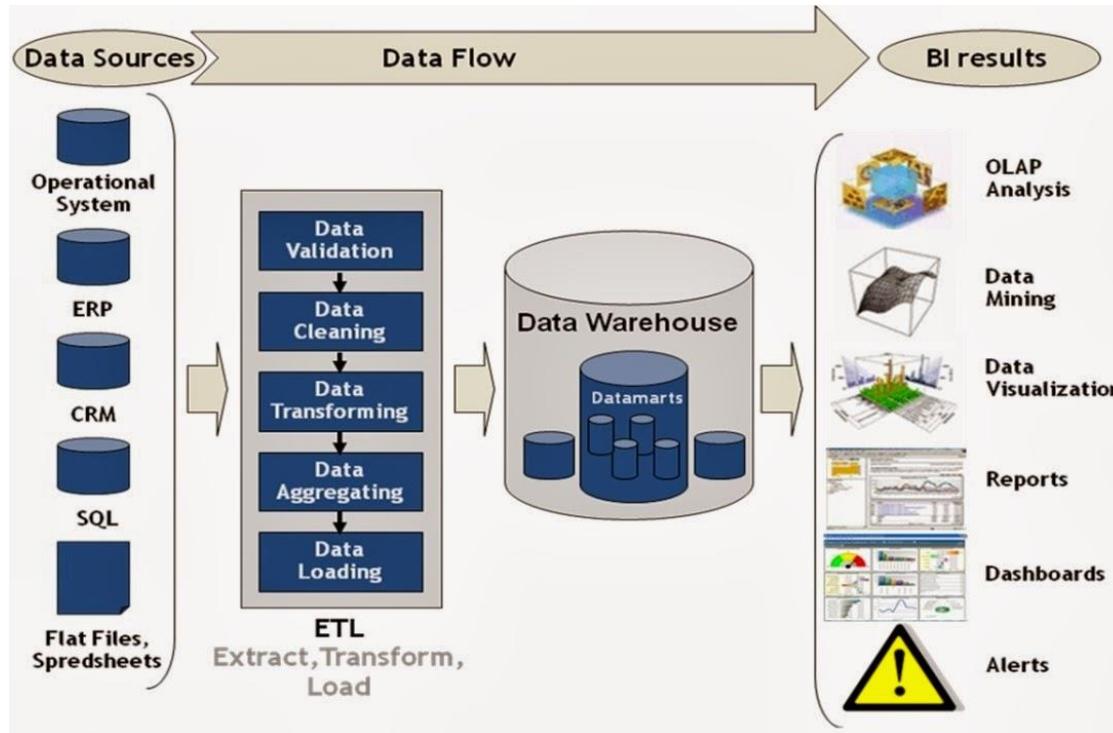
Data Warehouse



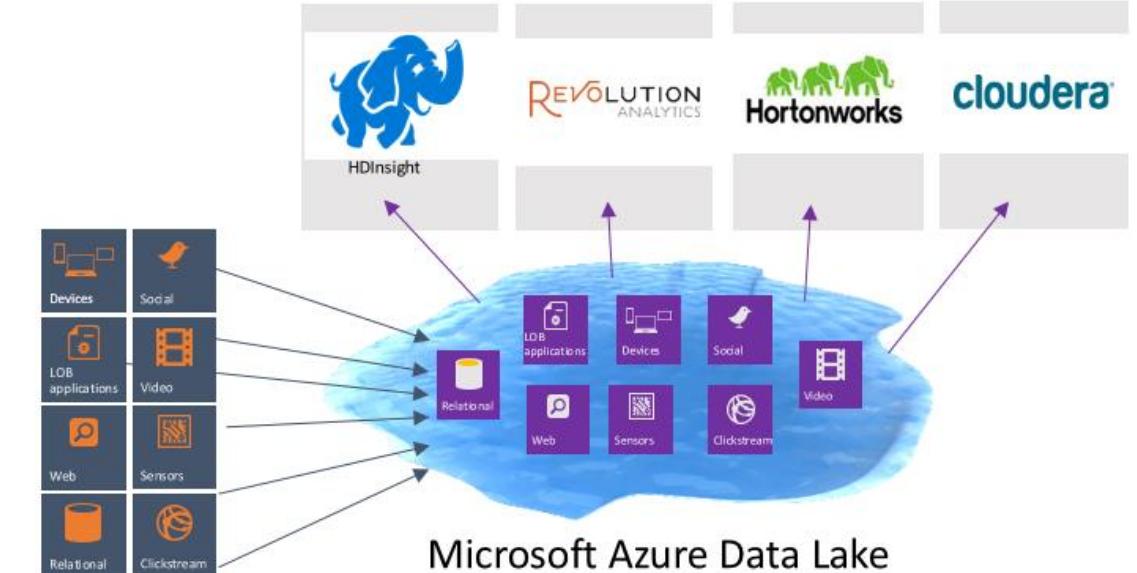
Data Lake



# Data Warehouse vs Data Lake



Data Warehouse



Data Lake





## Warehouse vs. Data Lakes

A word cloud centered around the theme of Big Data, containing terms like Storage, Volume, Management, Analysis, and Search.

The word cloud includes the following words:

- BIG
- DATA
- ANALYSIS
- RESEARCH
- INTERNET
- SEARCH
- MANAGEMENT
- INFORMATION
- TECHNOLOGIES
- GOVERNMENT
- PROCESS
- VOLUME
- STORAGE
- SOFTWARE
- SANAS
- RESULTS
- DRIVE
- REPLICATION
- HANDLE
- SOURCES
- LARGE
- COMPLEX
- PAST
- ANNOUNCED
- CHALLENGES
- INCLUDE
- NETWORKS
- HYPOTHESIS
- SCIENCE
- FUTURE
- MOST
- NEARLY
- MAPREDUCE
- CURRENTLY
- BILLION
- NEW
- CITIQUES
- DEVELOPMENT
- TECHNOLOGY
- APPROACH
- INITIATIVE
- ECONOMIC
- TERABYTES
- MARKET
- VARIETY
- RELATED
- TIME
- STATE
- CRITIQUE
- USE
- PER
- DISTRIBUTED
- RELEVANT
- PROJECT
- ANNUAL
- NOW
- USED
- PARALLEL
- STATE
- CRITIQUE
- TRAFFIC
- TIME
- SENSOR
- WORLD
- FLOW
- NATIONAL
- INTELLIGENCE
- PERMISSIONS
- APPROACHES
- PARADIGM
- FUNDING
- INSIGHT
- TOOLS
- CENTER
- ARCHITECTURE
- TERABYTES
- SET
- DEPARTMENT
- DEGREES
- HIGH
- USERS
- ALGORITHMS
- HIGHER
- COMPANIES
- REAL
- RATE
- STRUCTURE
- WORK
- APPLICATIONS
- INFRASTRUCTURE
- EXACTLY
- DETERMINE
- WELL
- DEFINITION
- STATISTICS
- BUSINESS
- EFFECTIVE
- VISUALIZATION
- DATABASES
- PROCESSING
- SIZE
- ANALYTICS
- SIMULATIONS
- MASSIVE
- FACTOR
- SYSTEMS
- MANAGE
- ONLINE
- DAY
- WORLDWIDE



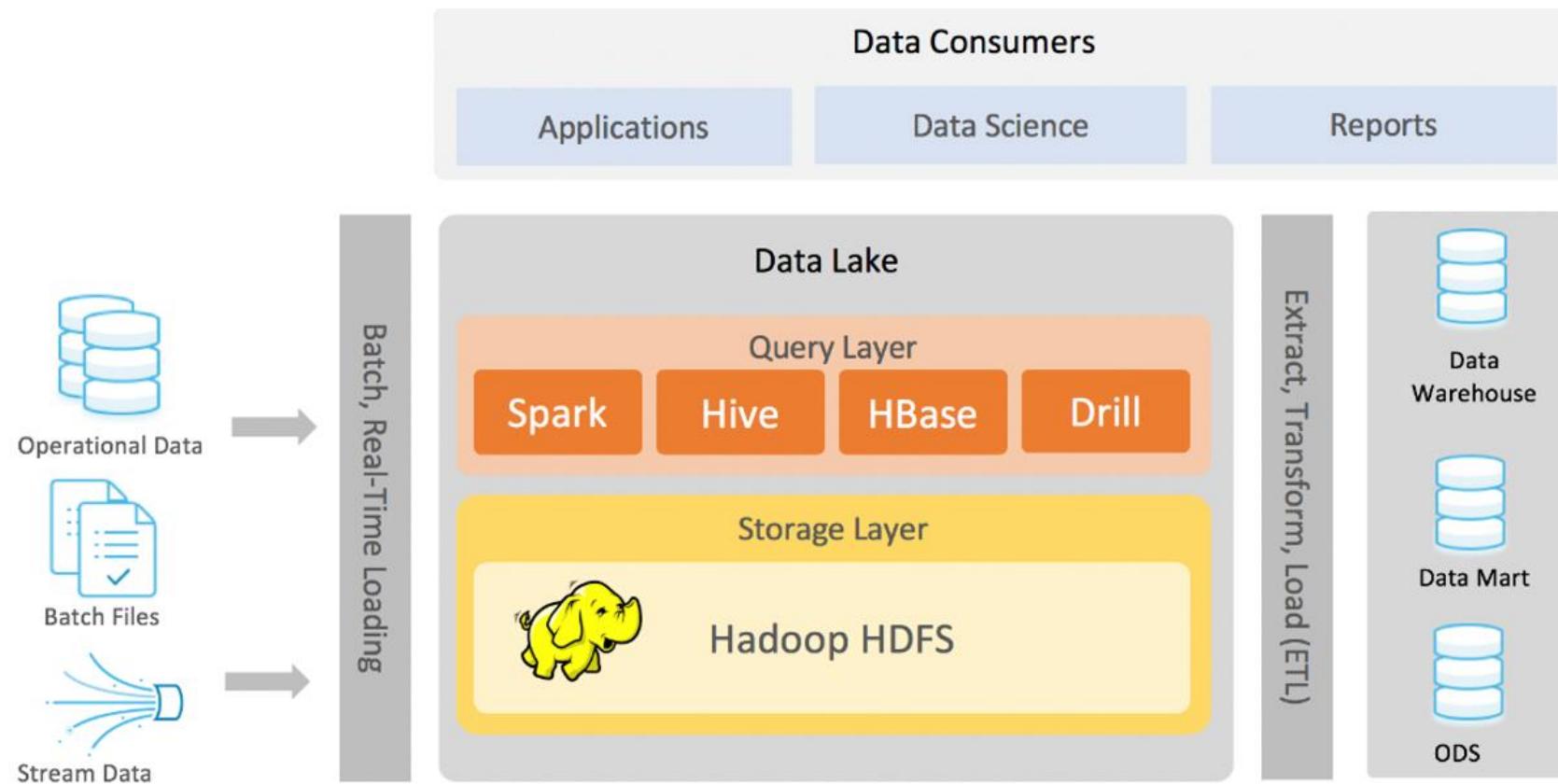
# Data lake vs Hadoop



VS



# Data lake vs Hadoop



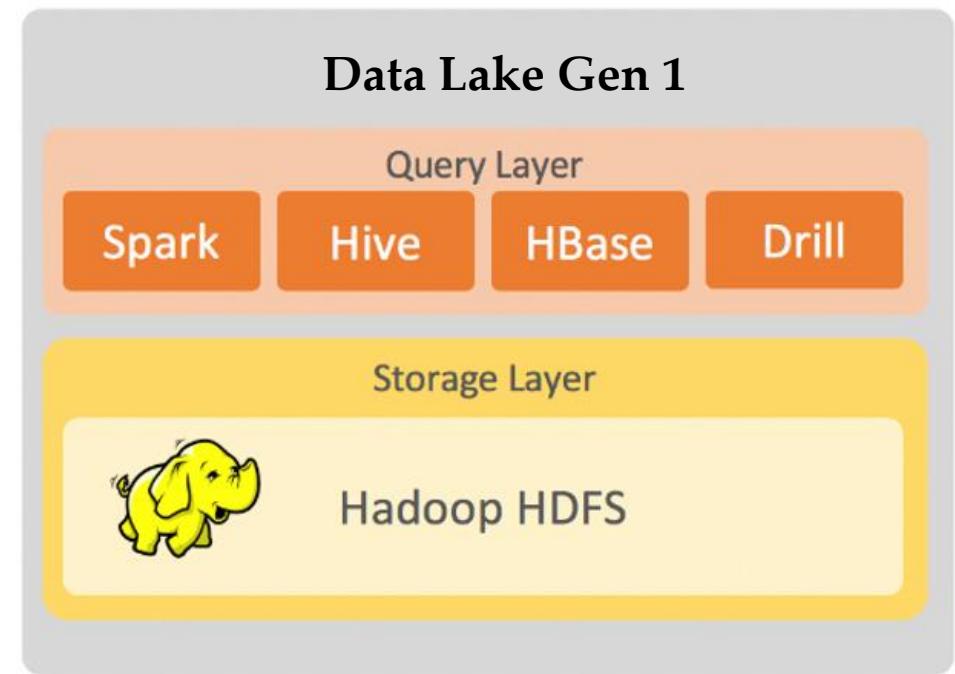
# Data Lake can use:



# Azure Data Lake Gen1 evolution



- Fault tolerant file system
- Runs on commodity hardware
- MapReduce, Pig, Hive, Spark etc.
- HDFS in Cloud -> Data Lake Storage Gen1

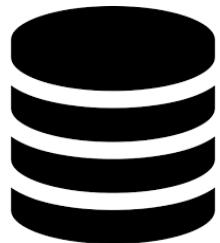


# Cloud storage challenge



## Processing

- Easy to optimize processing by increasing vCPU and Ram



## Storage

- Different requirements
- No direct solution

# Azure Blob Storage

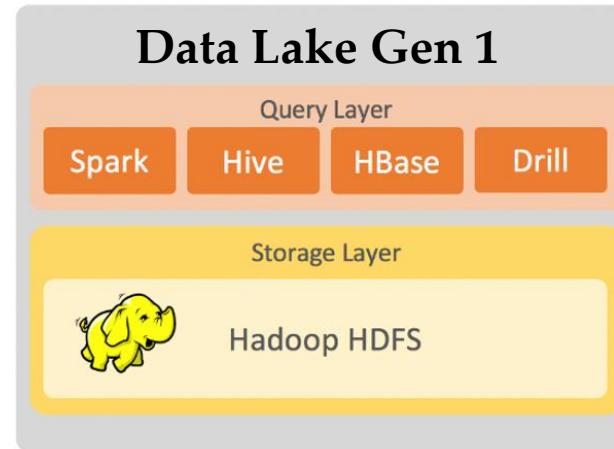
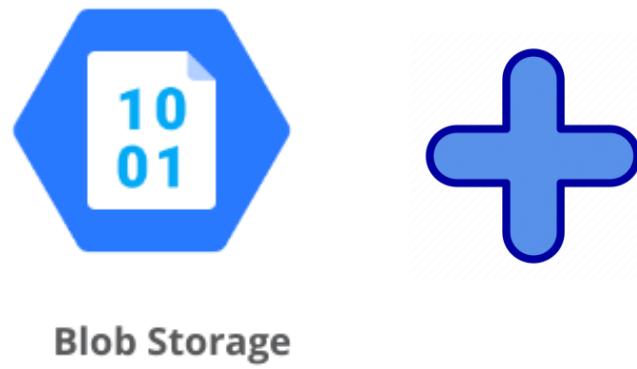
- Large object storage in cloud
- Optimized for storing massive amounts of unstructured data
  - Text or Binary Data
- General purpose object storage
- Cost efficient
- Provide multiple Tiers

**Microsoft Azure**  
Blob Storage



LearnCloud.Info

# Azure Data lake Gen 2



Azure Data Lake Storage Gen2

## MICRSOFT RECOMMENDS

Data Lake Storage Gen2  
for your big data storage needs.

Note: USQL currently not supported in Gen 2



## Azure Data Lake Storage Gen2



# Blob Storage vs Data Lake Storage

## Azure Blob Storage

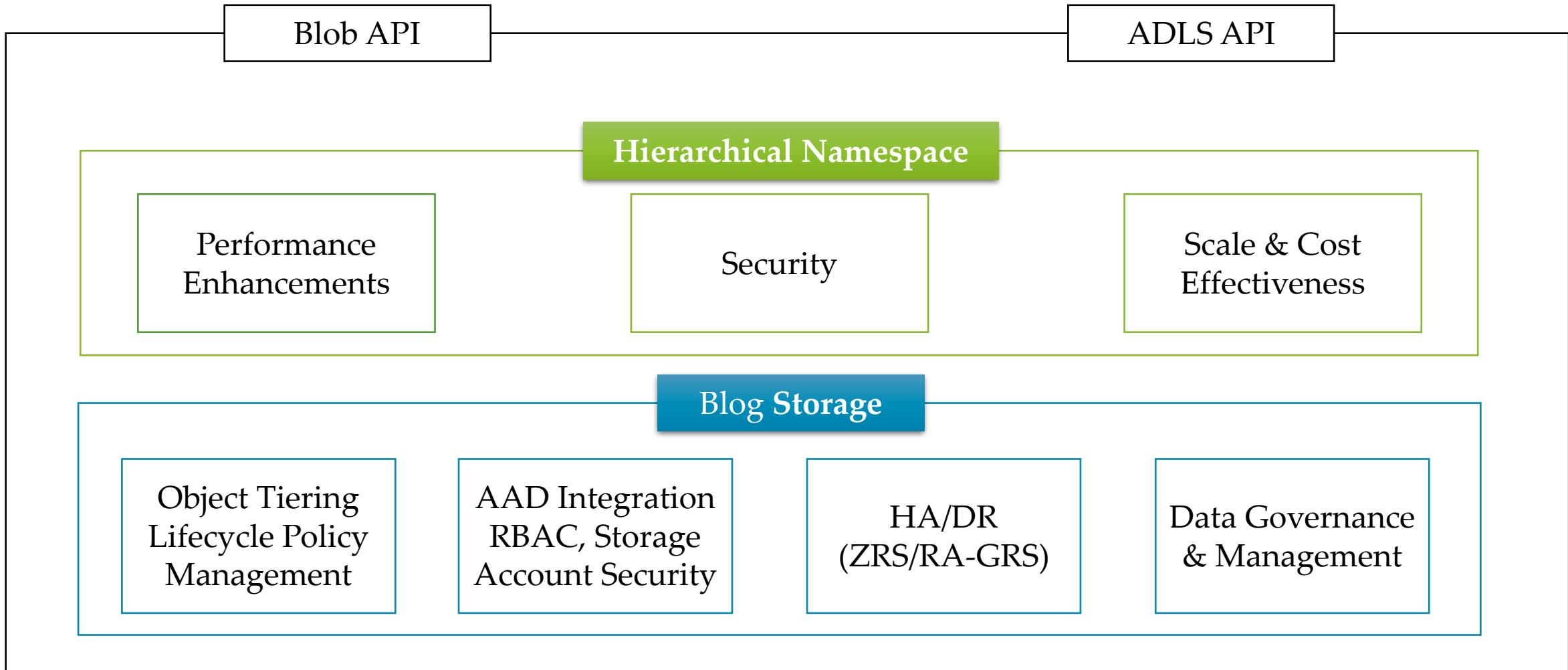
- General purpose data storage
- Container based object storage
- Available in every Azure region
- Local and global redundancy
- Processing performance limit

## Azure Data Lake Storage (Gen 2)

- Optimized for big data analytics
- Hierarchical namespace on Blob Storage
- Available in every Azure region
- Local and global redundancy
- Supports a subset of Blob storage features
- Supports multiple Azure integrations
- Compatible with Hadoop



# Data Lake Architecture



<https://expertslab.wordpress.com/2017/11/04/azure-storage-replication-options/>

<https://www.red-gate.com/simple-talk/cloud/cloud-data/understanding-azure-storage-options/>

<https://www.skylinesacademy.com/blog/2019/7/31/azure-storage-replication>

# Data redundancy for storage

**LRS – Locally-redundant storage:** Three copies of your data which is maintained within the same primary data center.

**ZRS- Zone-redundant storage:** Three copies of your data replicated synchronously to 3 Azure availability zones in a primary region. Zones are in different physical locations or different data centers.

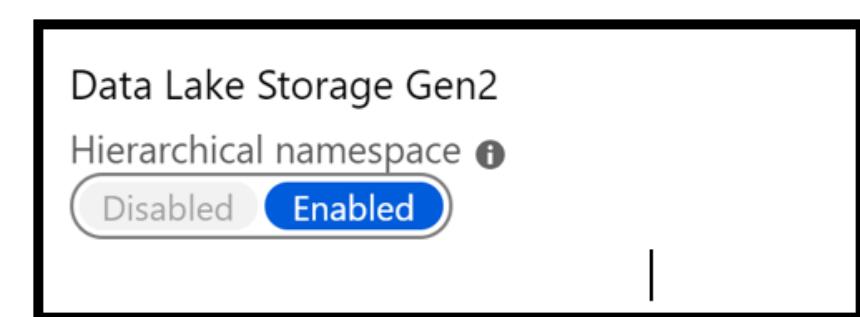
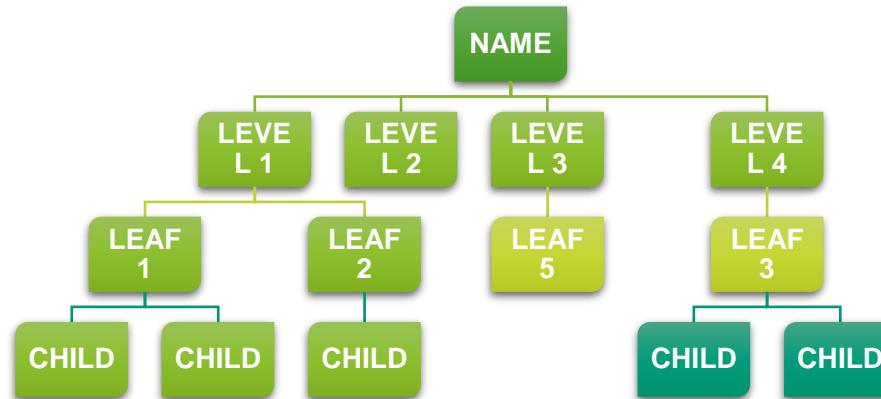
**GRS- Geo-redundant storage:** This allows your data to be stored in different geographic areas of the country or world. Again, you get three copies of the data within a primary region, but it goes one step further and places three additional asynchronous copies in another region. For example, you can now have a copy in Virginia and in California to protect your data from fires or hurricanes depending on the coast.

**RA-GRS- Read Access Geo-redundant storage.** This is GRS but adds a read-only element that allows you to have read access for things like reporting.

**Geo-zone-redundant storage (GZRS):** Copy your data synchronously over three primary region Azure availability zones using ZRS. It then asynchronously copies your data to a single physical location within the secondary region.

**Read-access geo-zone-redundant storage (RA-GZRS):** it adds a layer of readability to your secondaries.

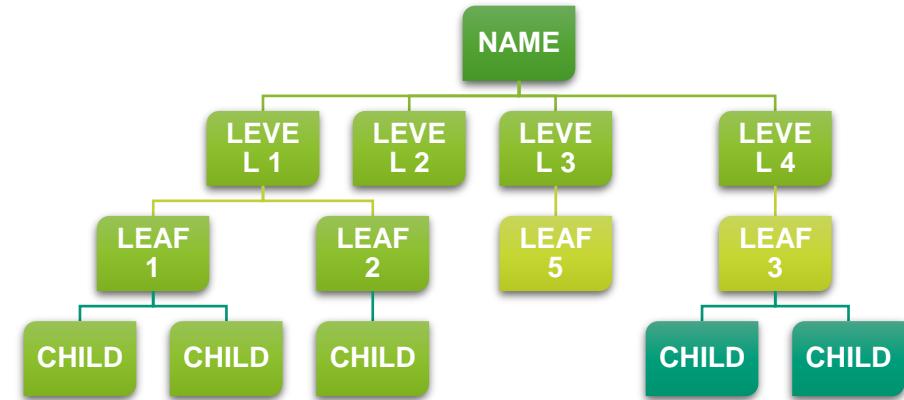
# Hierarchical namespace



- Hierarchical namespace organizes objects/files into a hierarchy of directories for efficient data access.
- Blob storage is not hierarchical namespace
  - Use slashes in Blob storage file names to stimulate a tree like directory structure
- Blob can't integrate with Hadoop
  - Because Blob doesn't have hierarchical namespace

# Hierarchical namespace

- Atomic directory manipulation
  - Manageable
  - Performance
  - Cost
  - Familiar interface style
- Why it was not done before in Blob?
  - Blob – limit scale
  - Lake Gen2 scale linearly so it does not degrade capacity or performance
- When you do not want to use Hierarchical namespace?
  - Where object organization stored separately
  - E.g.: Backups, image storage, etc.
- Note: Hierarchical namespace setting can't revert back



Data Lake Storage Gen2  
Hierarchical namespace ⓘ

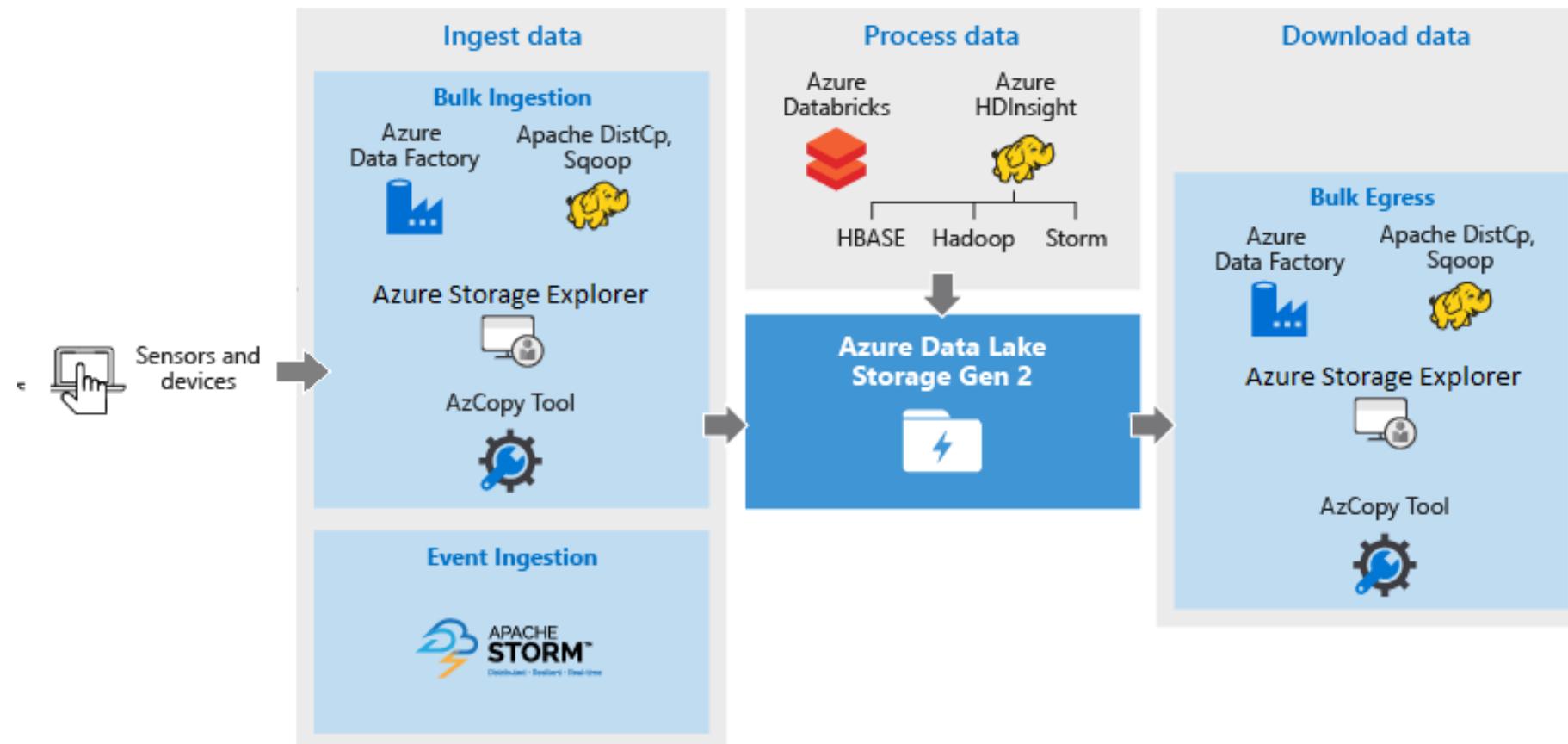
# Important features of Data Lake Gen2

- Integration
  - POSIX compliant
  - Hadoop Integration (use ABFS driver)
  - Other Azure services integration
- Scalability
- Cost effective
  - Build on top of Azure low cost blob storage
  - No need to move data
  - Hierarchical namespace -> less compute -> save cost
- Performance – optimized for high speed throughput
- Security
- Fault tolerant/ High availability/ Disaster recovery
- Global footprint
- Challenges
  - Hard to query unstructured data
  - Hard to manage data quality

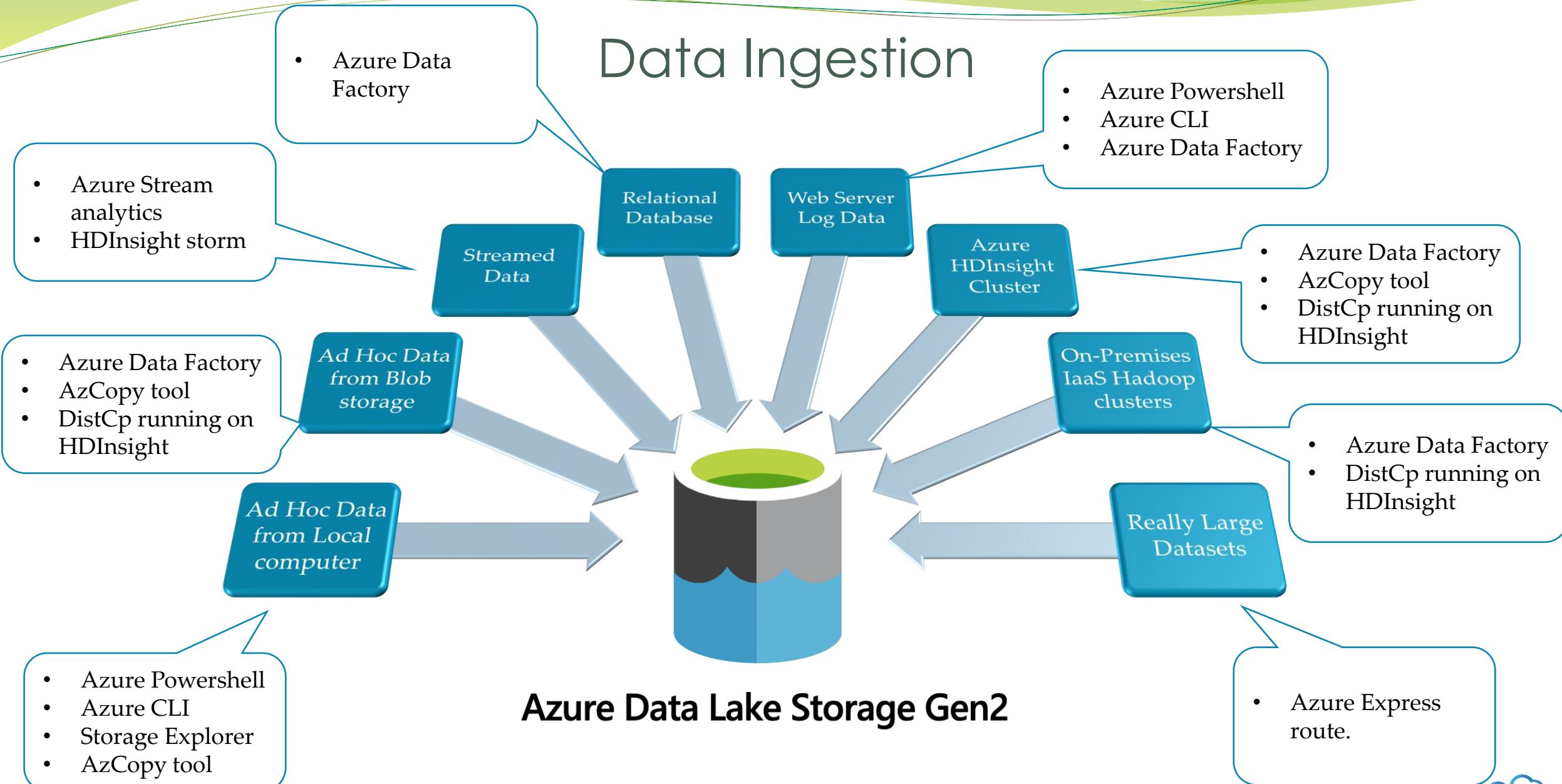


**Azure Data Lake Storage Gen2**

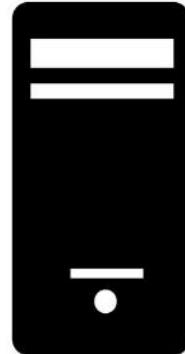
# Data Lake Architecture



# Data Ingestion



**Azure Data Lake Storage Gen2**

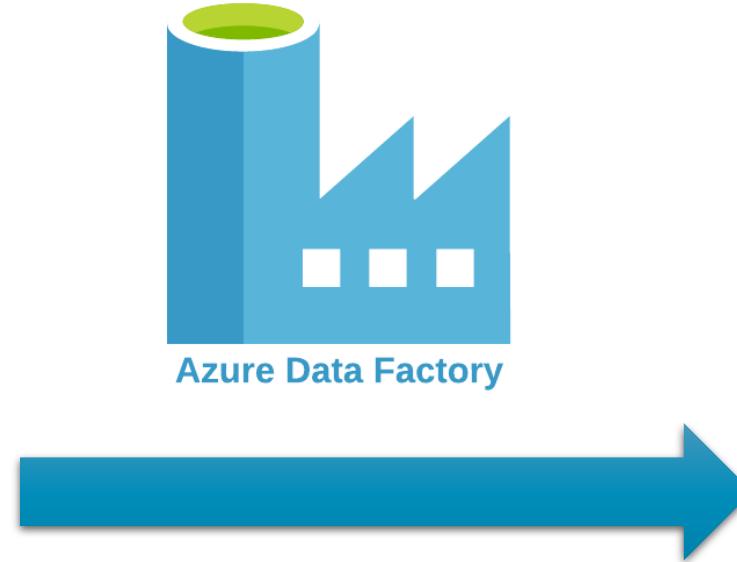


AzCopy



Azure Data Lake Storage Gen2

A screenshot of a Windows Command Prompt window titled "Command Prompt". The window displays the AzCopy 5.0.0 help text, which includes usage instructions, options, and detailed command-line help. The text is too small to read in detail but follows the standard AzCopy documentation structure.





Azure  
SQL Database



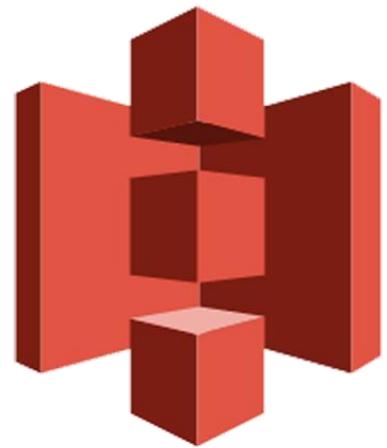
Azure Data Factory



Azure Data Lake Storage Gen2



LearnCloud.Info

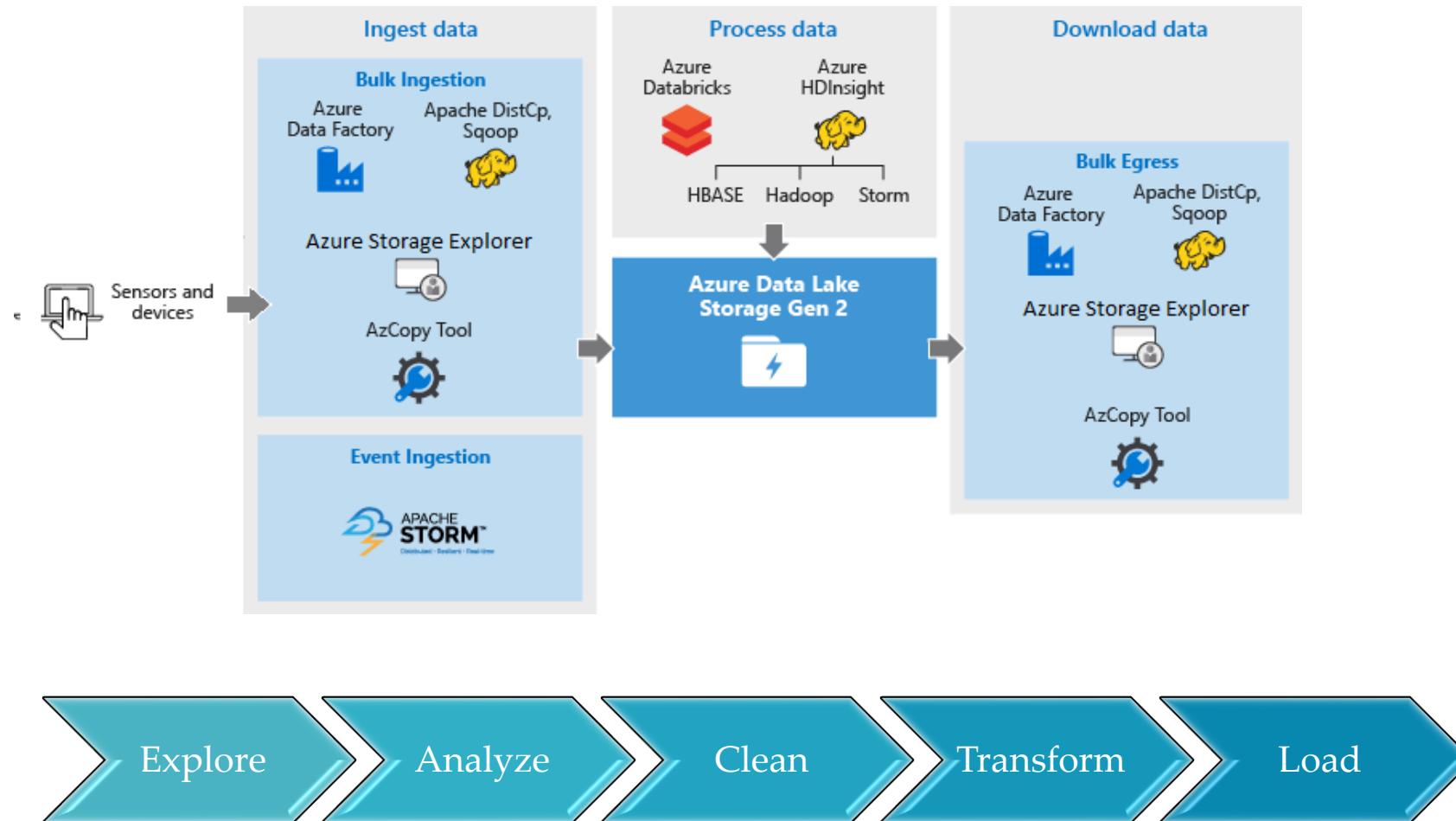


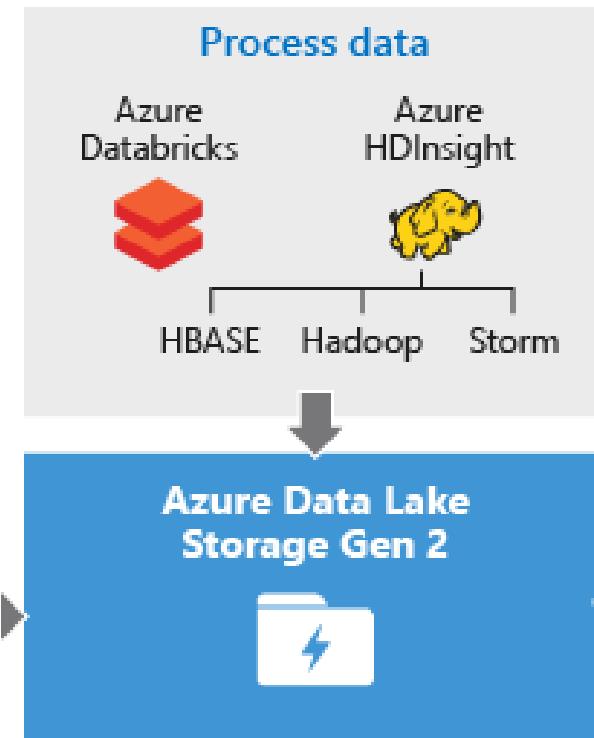
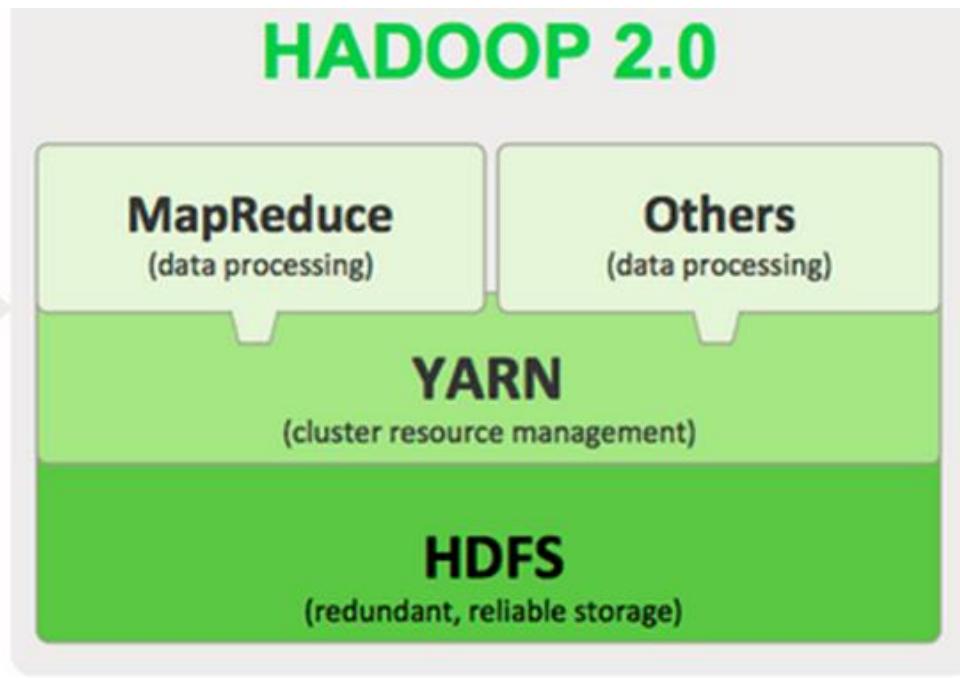
Amazon S3



Azure Data Lake Storage Gen2

# Data flow around Data Lake





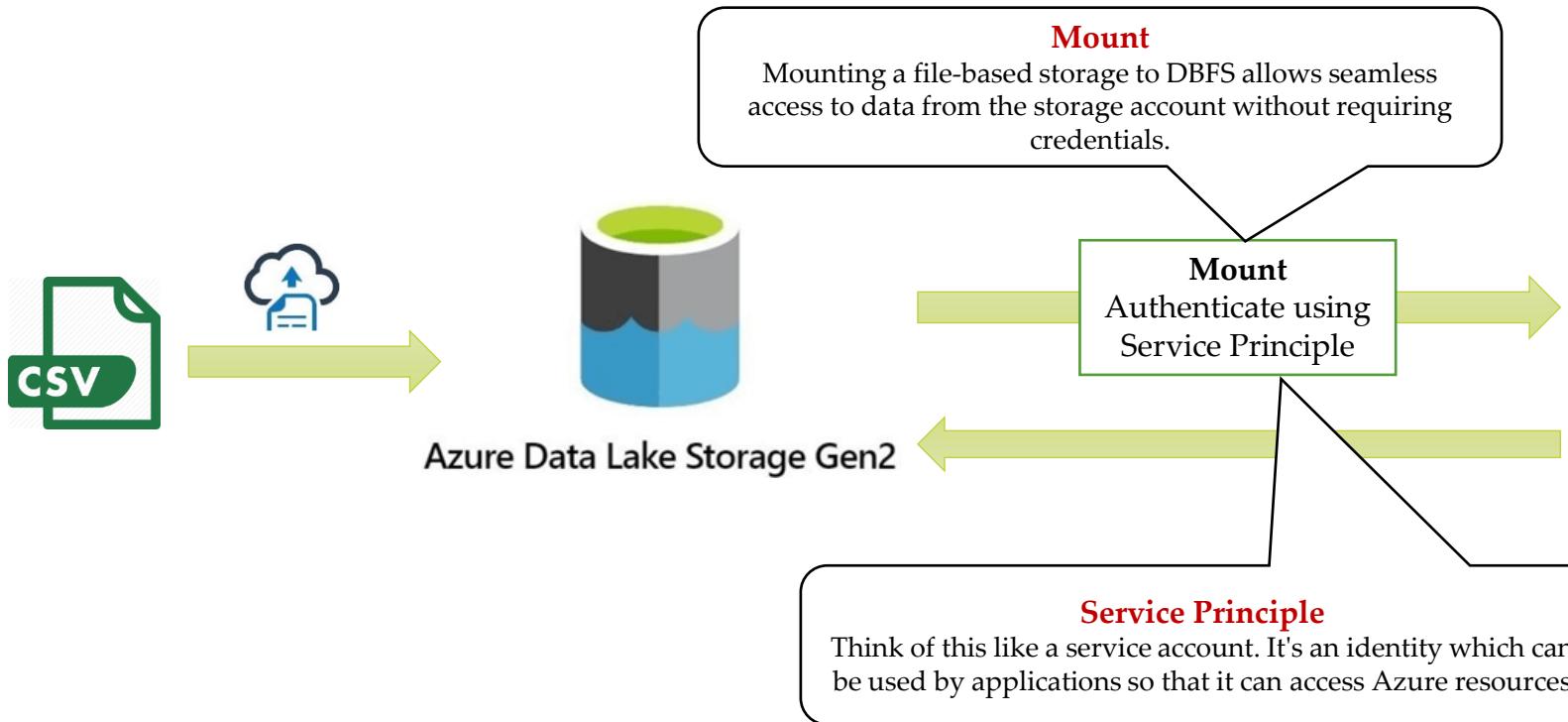
- Clusters are tightly coupled with HDFS
- If you stop cluster
  - storage will also gone
  - and you will lose all the data
- Costly
  - Clusters have to keep running even if there is no processing, and pay for both cluster and storage

- Storage is separate from clusters
- If you stop cluster
  - Data is stored in Data Lake separately
  - and you will NOT lose any data
- Cost efficient
  - You can terminate cluster when not required, and only pay for storage

# Demo overview



databricks®



# Learning objective



- **Authentication**
  - Storage Account keys
  - Shared access signature (SAS)
  - Azure Active Directory (Azure AD)
- **Access Control**
  - Role based access control (RBAC)
  - Access control list (ACL)
- **Network access**
  - Firewall and virtual network
- **Data Protection**
  - Data encryption in transit
  - Data encryption at rest
- **Advanced threat Protection**

# **Storage Account Access Keys**

Authentication

# Shared Access Signature (SAS)

Authentication



# Shared Access Signature (SAS)



Shared Access Signature

Security token string  
“SAS Token”  
Contains permission like start and end time  
Azure doesn't track SAS after creation  
To invalidate, regenerate storage account  
key used to sign SAS

# Stored access policy



Stored access policy

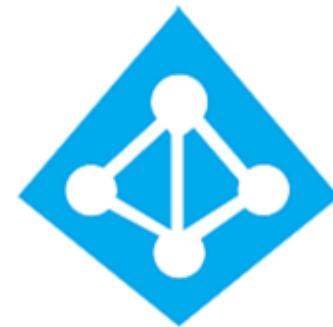
- Reused by multiple SAS
- Defined on a resource container
- Permissions + validity period
- Service level SAS only
- Stored access policy can be revoked

# Azure Active Directory

Authentication



Azure Active Directory



# Azure Active Directory

# Azure Active Directory (AD)

- Grant access to Azure Active directory (AD) **Identities**
- AD is an enterprise identity provider, Identity as a Service (IDaaS)
- Globally available from virtually any device
- Identities – user, group or application principle
- Assign role at Subscription, RG, Storage account, container level.
- No longer need to store credentials with application config files
- Similar to IIS Application pool identity approach



**Azure Active Directory**

# **Role based access control (RBAC)**

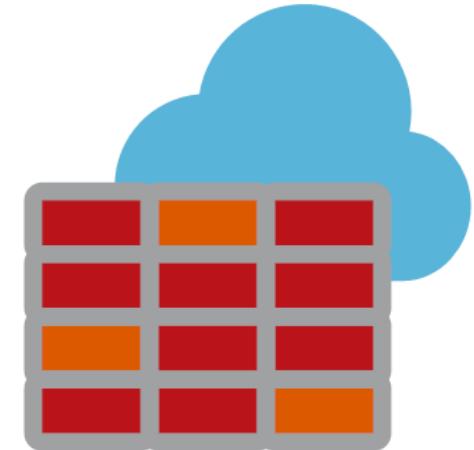
Access Control



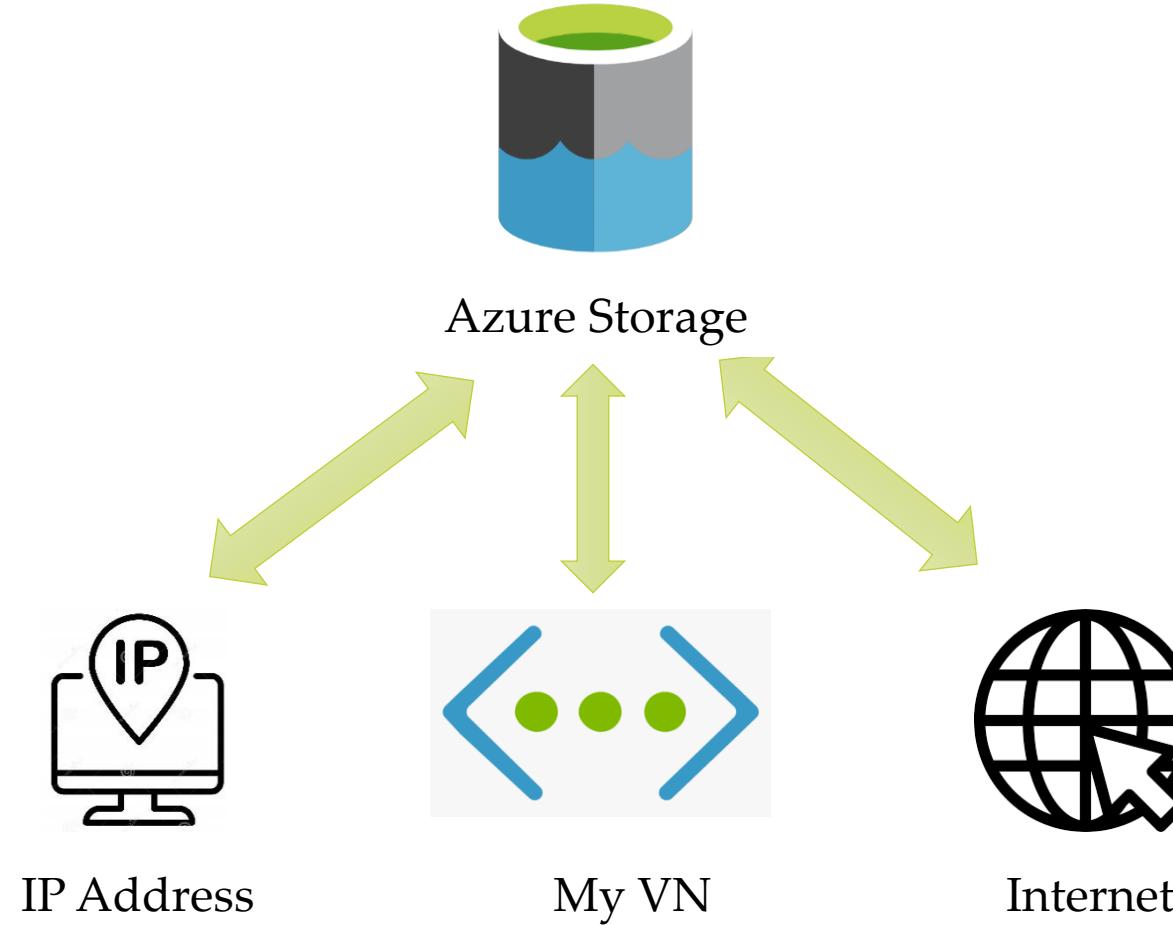
# **Role based access control (RBAC)**

Access control

# Firewalls and Virtual Networks



# Firewalls and Virtual Networks



# **Storage Account Access Keys**

Authentication

# **Shared Access Signature (SAS)**

Authentication

# Shared Access Signature (SAS)



Azure doesn't track SAS after creation

To invalidate, regenerate storage account  
key used to sign SAS

# Stored access policy



Stored access policy

- Reused by multiple SAS
- Defined on a resource container
- Permissions + validity period
- Service level SAS only
- Stored access policy can be revoked

# Azure Active Directory

Authentication

# **Access Control List (ACL)**

Access control

# Azure Active Directory (AD)

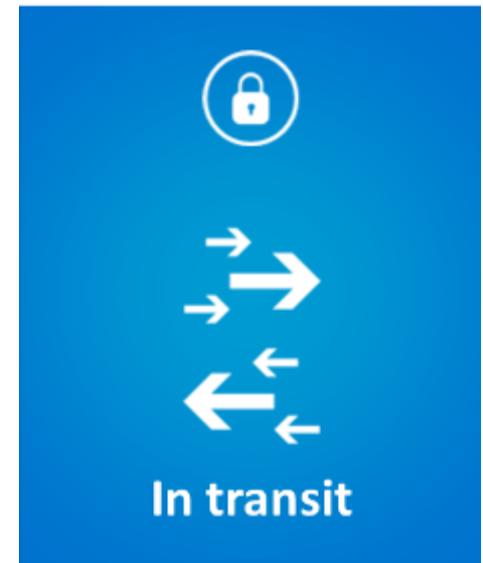
- Grant access to Azure Active directory (AD) **Identities**
- AD is an enterprise identity provider, Identity as a Service (IDaaS)
- Globally available from virtually any device
- Identities – user, group or application principle
- Assign role at Subscription, RG, Storage account, container level.
- No longer need to store credentials with application config files
- Similar to IIS Application pool identity approach



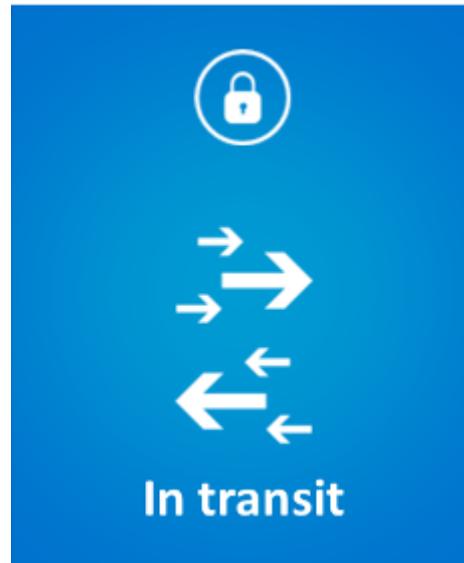
**Azure Active Directory**

# Encrypting Data in Transit

Data Protection

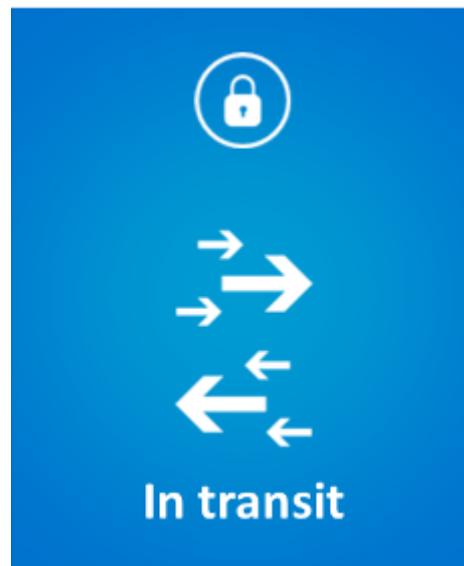


# Encrypting Data in Transit – Advance



- Site-to-site VPN
- Point-to-site VPN
- Azure ExpressRoute

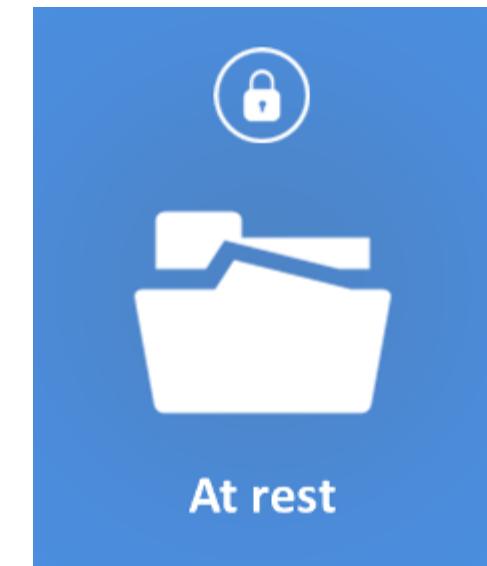
# Client-side Encryption



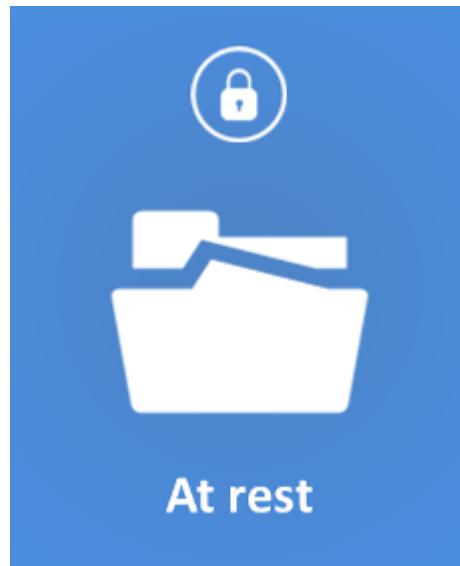
- Encrypt data within application
- Data is encrypted in transit and at rest
- Application decrypt data when retrieved
- HTTPS has integrity checks built-in
- .Net and Java storage client libraries
- Can leverage Azure Key Vault to generate and/or store encryption keys

# Encrypting Data at Rest

Data Protection



# Encrypting Data at Rest



- Encryption enabled by default
- Can't be disabled
- Storage Service Encryption (SSE)
  - Automatically encrypt and decrypt while writing and reading
  - It's free, no charge
  - Applied to both standard and premium tiers
  - 256 bit AES Encryption
- Option: Use your own encryption keys
  - Blobs and files only

 Microsoft Azure

MEDIUM SEVERITY

Someone has accessed your Storage account 'mystorageaccount' from an unusual location.

**Activity details**

Subscription ID	34B9A4E7-002a-411f-88d0-98557A6C79E6
Storage account	mystorageaccount
Storage type	Blob
Container	mycontainer
Application	myTestApplication
IP address	13.80.80.98
Location	Washington, United States
Data center	SCUS
Date	May 17, 2018 7:50 UTC
Potential causes	Unauthorized access that exploits an opening in the firewall. Legitimate access from a new location
Investigation steps	<a href="#">For a full investigation, configure diagnostics logs for read, write, and delete</a>
Remediation steps	Be sure to follow the principle of "least privilege" and <a href="#">limit access to your data</a>

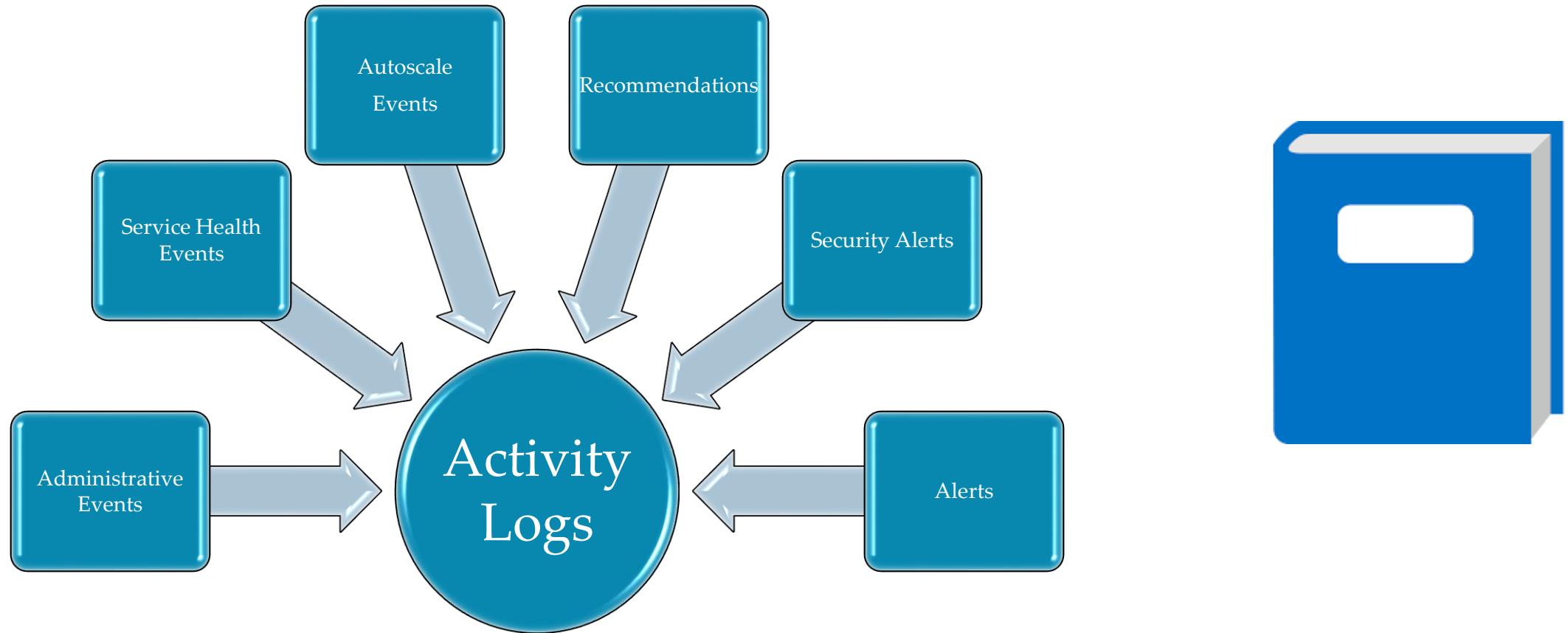
# Advanced threat protection

# Activity logs

- Logs operations performed on Storage Account itself (Management log)
  - Role assignments
  - Regenerating Storage Account keys
  - Changing Storage Account settings
- Does not include data related events
  - Diagnostic events are handled by Storage Analytics
- Subscription wide log



# Activity Logs



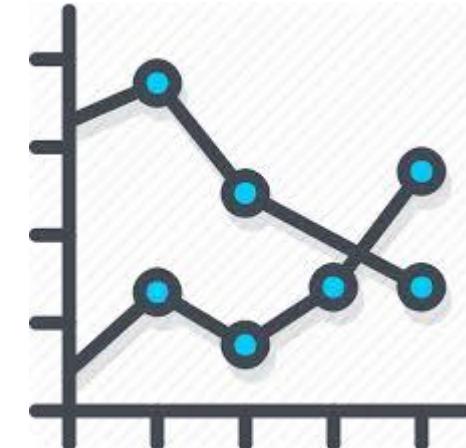
# Activity logs

- Events available in Azure Portal for 90 days
- Activity Log destinations
  - Storage Account
  - Event Hub
  - Log Analytics
- Query Activity Log data
  - Within Azure Portal
  - PowerShell, Azure CLI, REST API
  - Azure Log Analytics



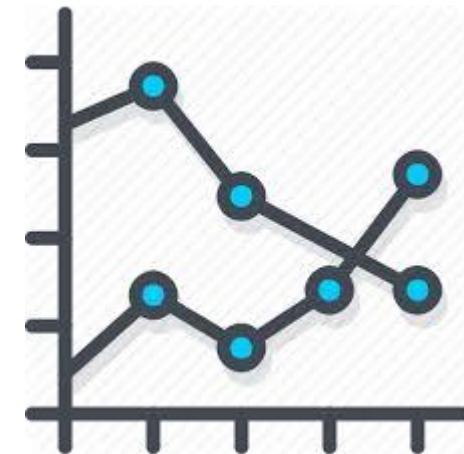
# Metrics

- Numerical values
- Collected at regular intervals
- Azure Storage
  - Capacity metrics
  - Transactions metrics
- Can set alerts on metrics



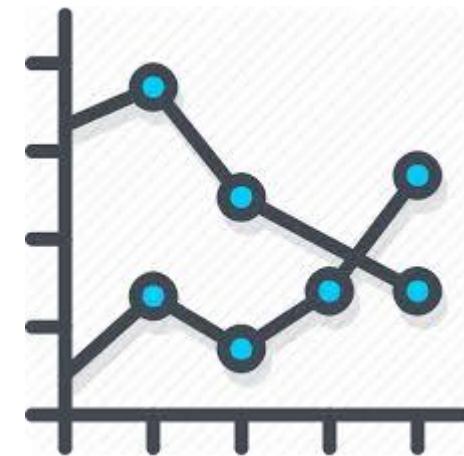
# Metrics

- Time-series database
- Set of metric values contains:
  - Time value collected
  - Resource collected from
  - Category
  - Metric name
  - Metric value
  - Name/value pairs (multi-dimensional metrics)



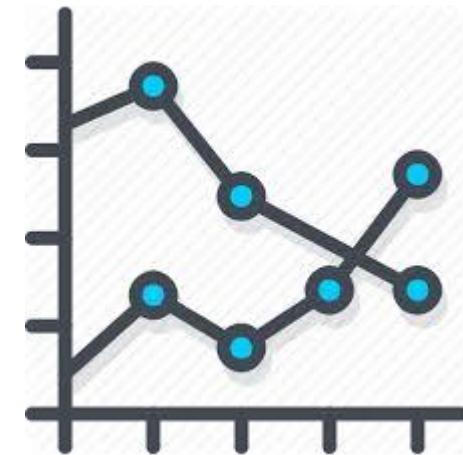
# Metrics

- Most metrics collected every minute
- Near real-time monitoring
- 93 day retention
- Alerts



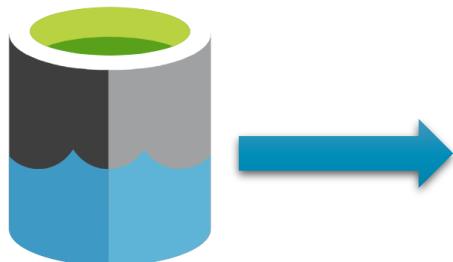
# Metrics

- Access Storage Account metrics
  - Storage Account
  - Azure Monitor
  - Azure Monitor REST Endpoints
  - .NET SDK
  - PowerShell
  - Azure CLI
  - Event Hubs

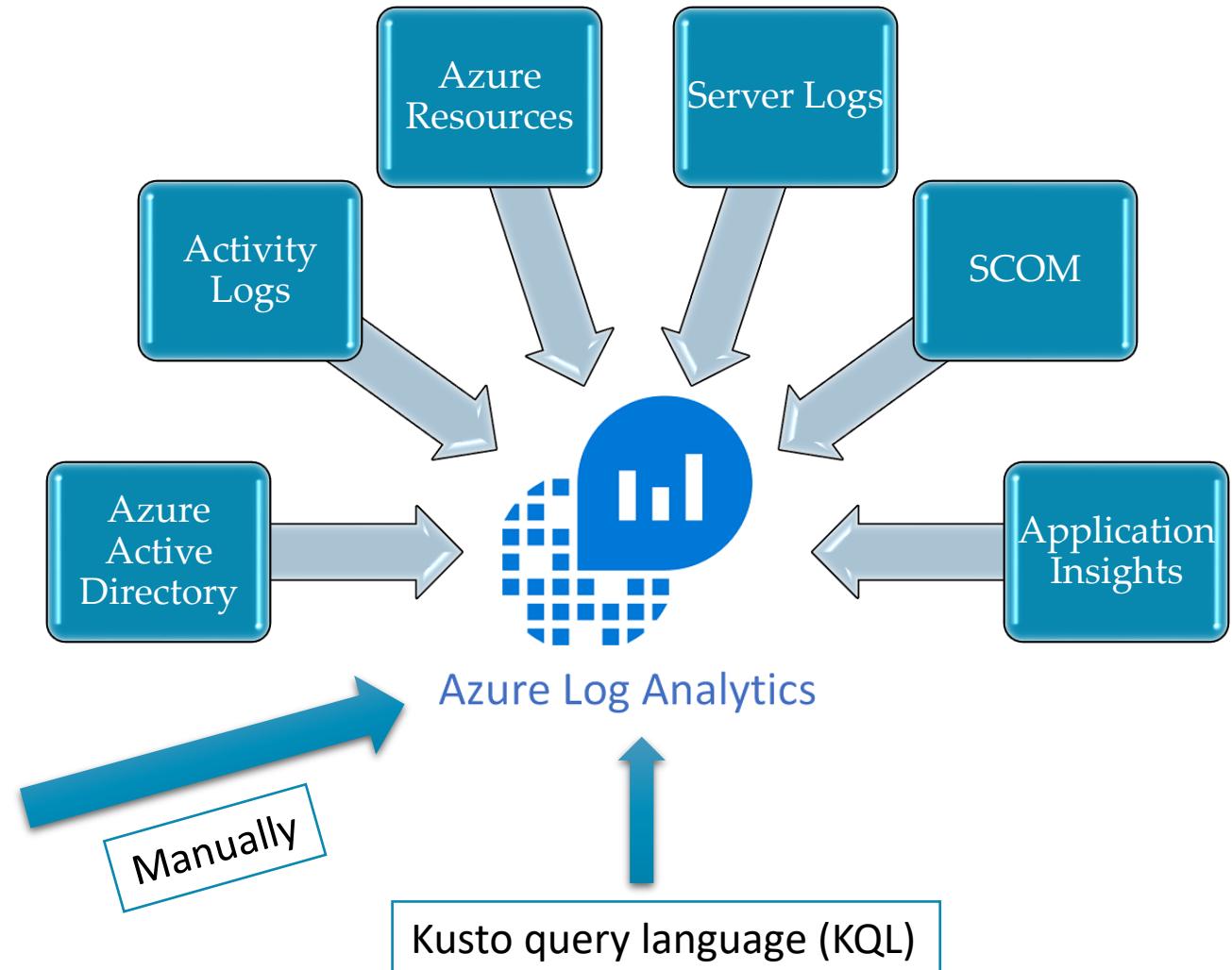


# Diagnostic settings

- Logs
  - Events that occur within a system
  - Numeric or text data
- \$Logs contains
  - Details of read, write and delete operations
  - Reasons for failed request

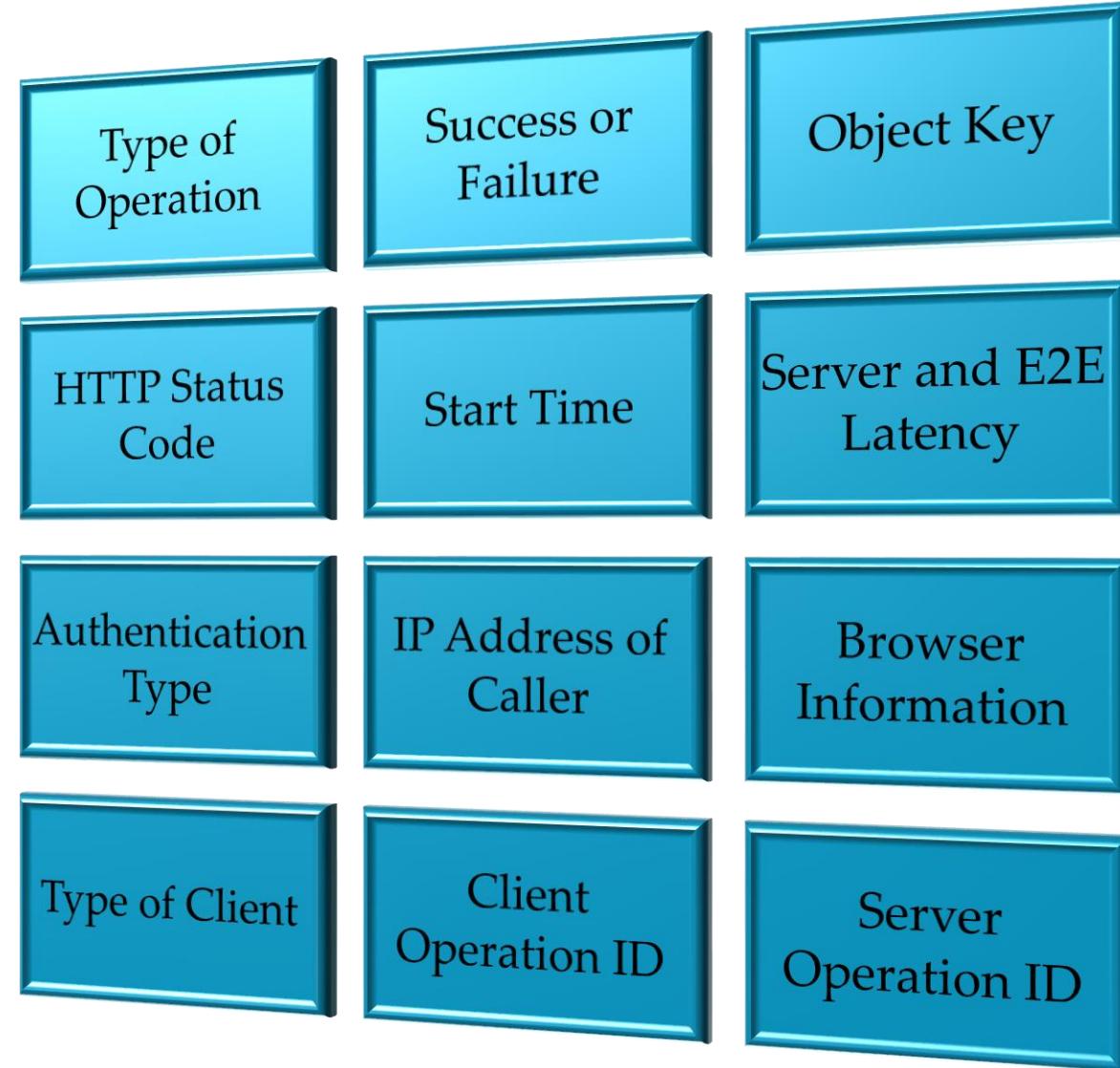


\$Logs  
(Folder in container)





Storage Analytics Logs





### Storage Analytics Logs

Blocks are written immediately

Blob is not available until log writer is flushed by Azure

“Last modified datetime” may not accurately reflect log contents

Search +/-15 minutes

Search based on Log metadata

There are costs for logging

20 TB limit, independent of Storage Account total limit

# Data Lake Optimization Techniques



- **Data Ingestion considerations**
  - Storage hardware
  - High speed internal network
  - Fast network connection b/w on-premises and cloud
- **Parallel read/write**
  - e.g. Data Factory parallel copies settings
- **Structure your data set**
  - File size vs number of files
    - File size b/w 256 MB to 100 GB
  - Folder and file structure
    - e.g. Dataset\YYYY\MM\DD\datafile\_YYYY\_MM\_DD\_HH\_MM.tsv
- **Same region**
- **Batch Data**