BIG DATA

STORAGE

LEARNING OBJECTIVES

- At the end of this presentation, you should be able to
  - explain what big data storage options are available;
  - understand the pros and cons of relational and non-relational databases;
  - given a dataset of a certain type and usage, make an informed decision how to store it;
  - explain the role of Cloud technology in data storage.

HISTORY

Big Data has existed for a long time

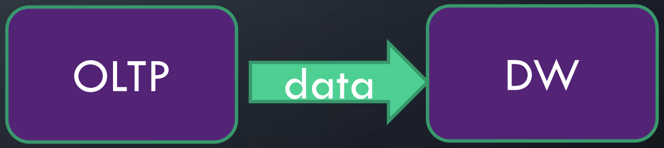
HISTORY OF LARGE DATA STORAGE

- Large companies with lots of data used large RDBMSs

IBM Db2

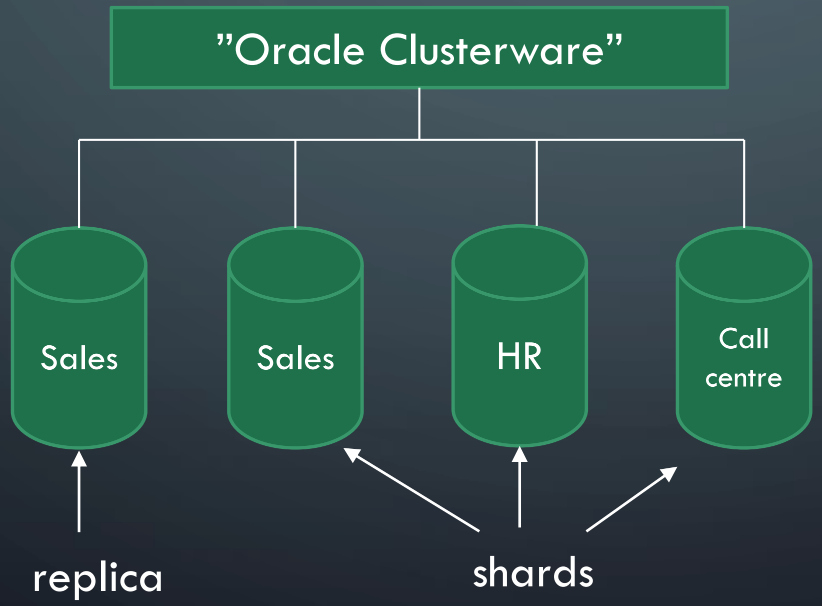
ORACLE

Version	Released	Features
Oracle v2	1979	First commercial RDBMS
Oracle 8i	1997	Recovery Manager, Partitioning, Java
Oracle 9i	2001	Clustering, data warehousing
Oracle 11g	2009	White papers on exports/imports with Hadoop
Oracle 12c	2013	Cloud service, JSON
Oracle 18c	2018	MDX queries



DEALING WITH BIG DATA BEFORE BIG DATA

- Scalability – an integral part of RDBMS development

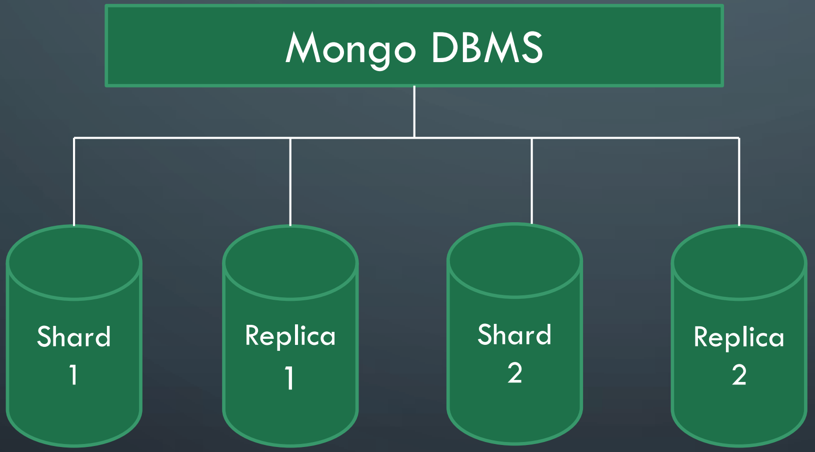


from Oracle 9i, 2001

up to 1000 servers

NOSQL DATABASES

- ”Not Only SQL”, but means ”non-relational” in practice

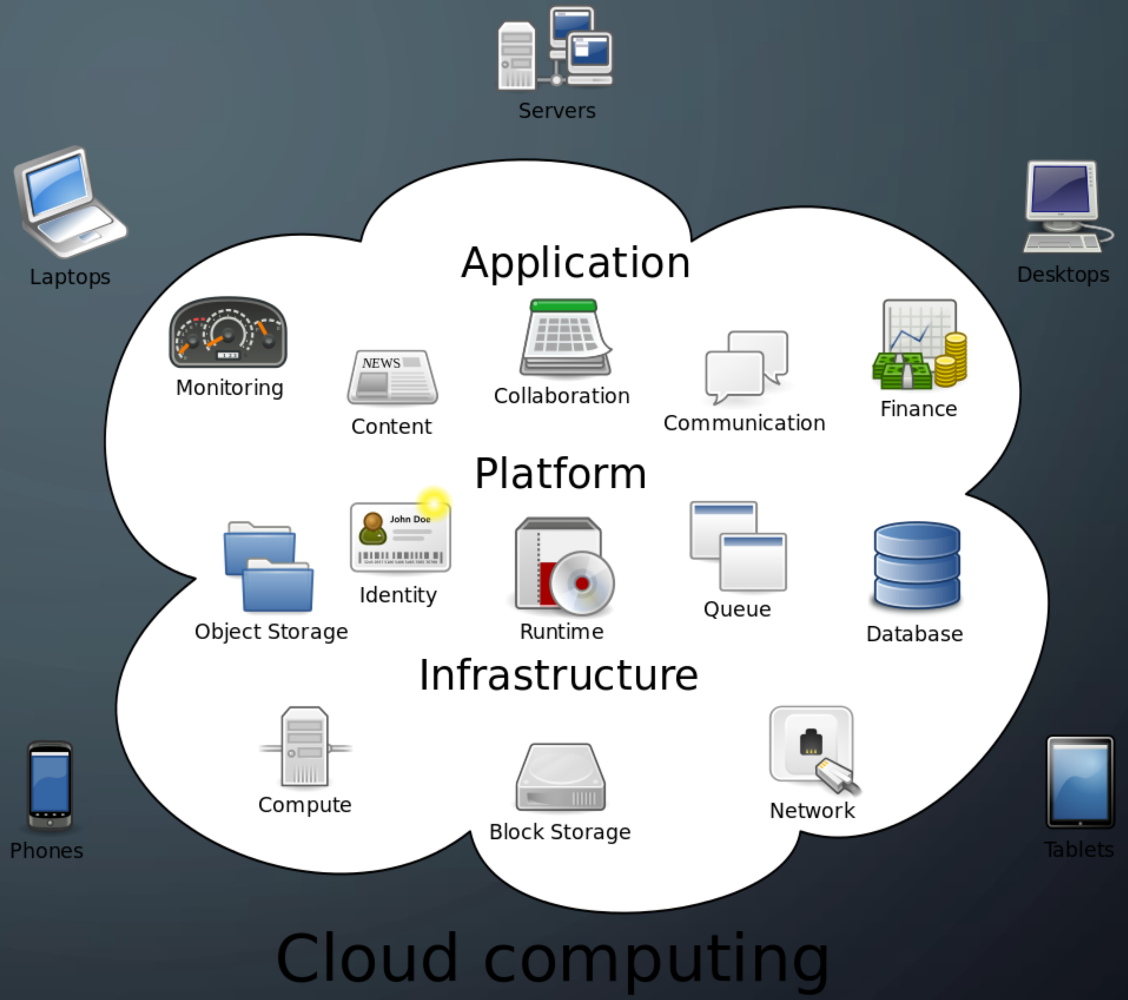


”Document database”

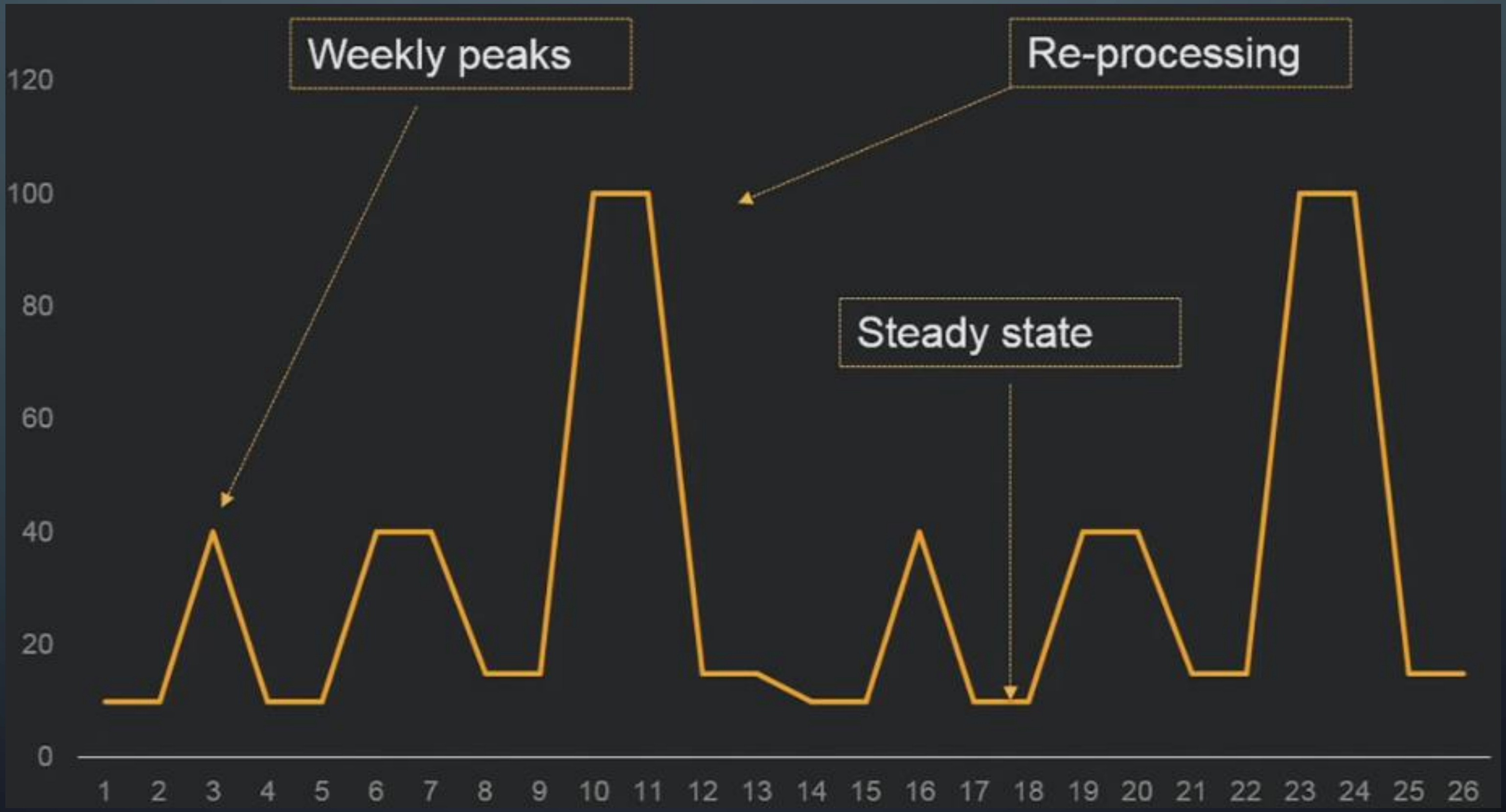
”Collections”

CLOUD COMPUTING

exists because of mobile devices and big data.



BIG DATA PROCESSING PATTERNS



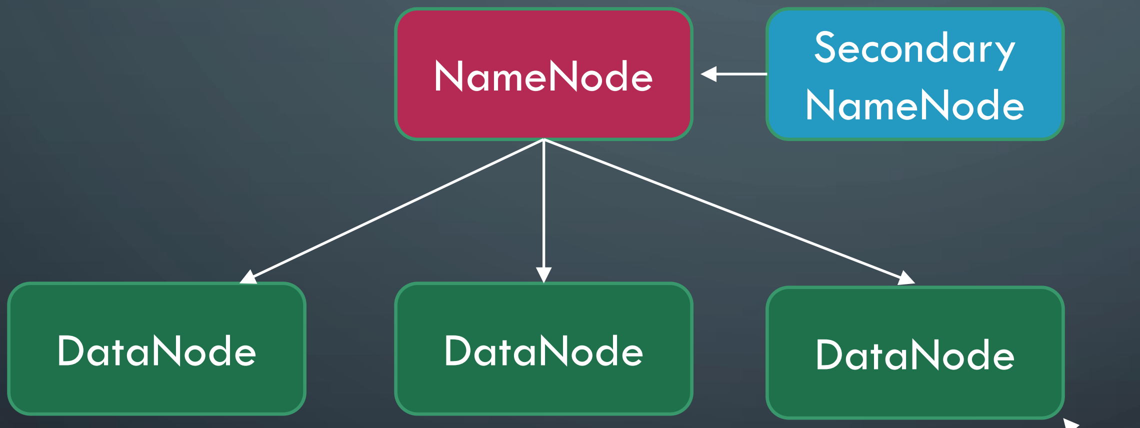
Source: Amazon

CLOUD SERVICES

- What?
  - Computing resources as a metered service (“pay as you go”)
  - Ability to dynamically provision virtual machines
- Why?
  - Cost: capital vs. operating expenses
  - Scalability: “infinite” capacity
  - Elasticity: scale up or down on demand
- Data Storage in the Cloud
  - Also scales with size and demand

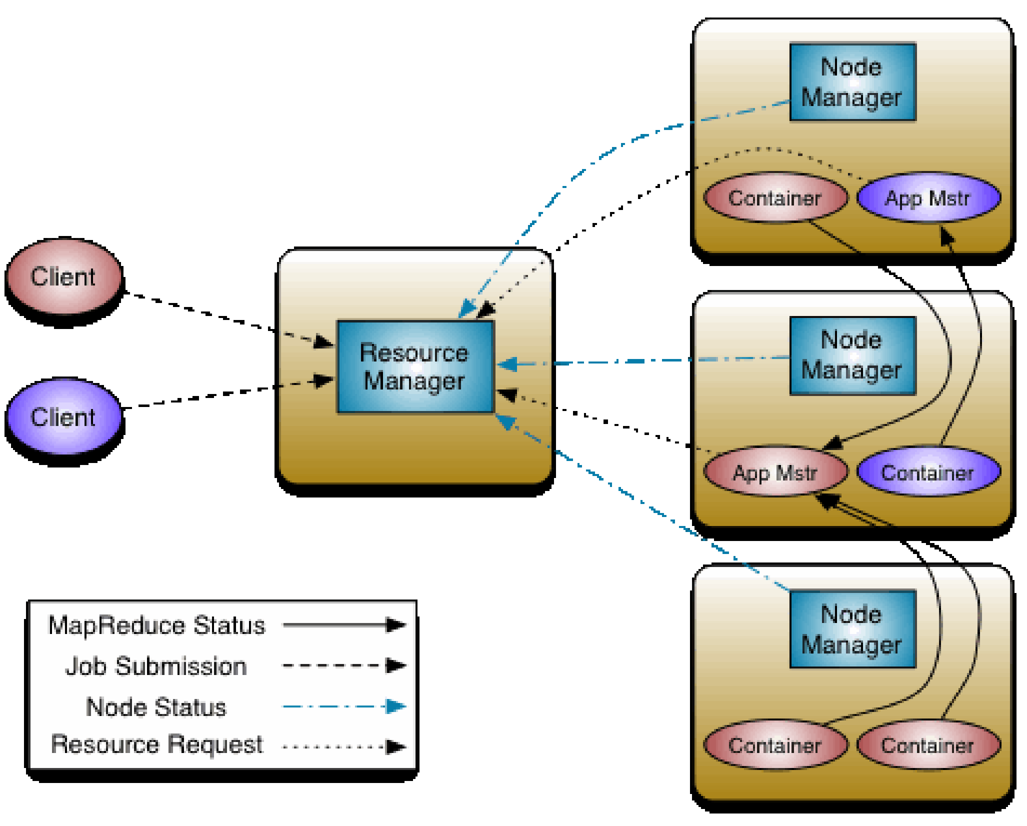
HADOOP: TECHNOLOGY FOR BIG DATA

- The Hadoop file system is always distributed.



default size 128MB

HADOOP CLUSTER - YARN



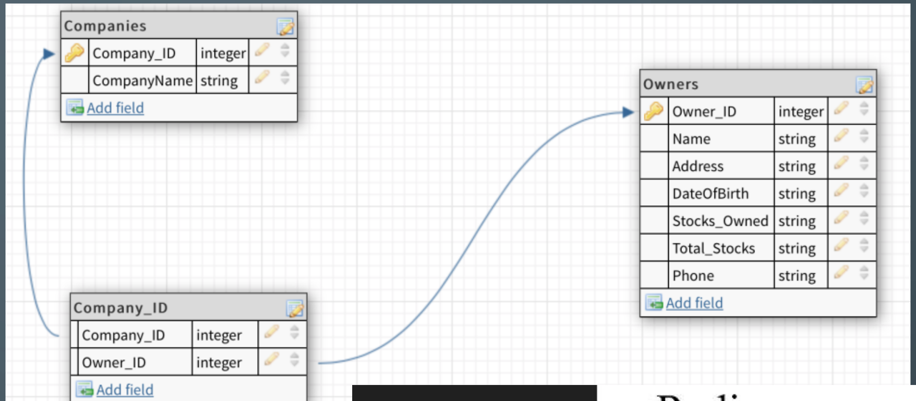
cloudera

aws

HORTONWORKS

STRUCTURE OR NO STRUCTURE?

Relational vs non-relational storage

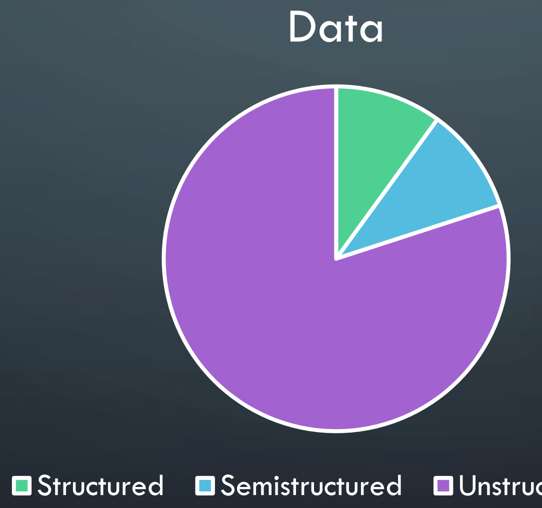


Companies	
Company_ID	integer
CompanyName	string
Add field	

Company_ID	
Company_ID	integer
Owner_ID	integer
Add field	

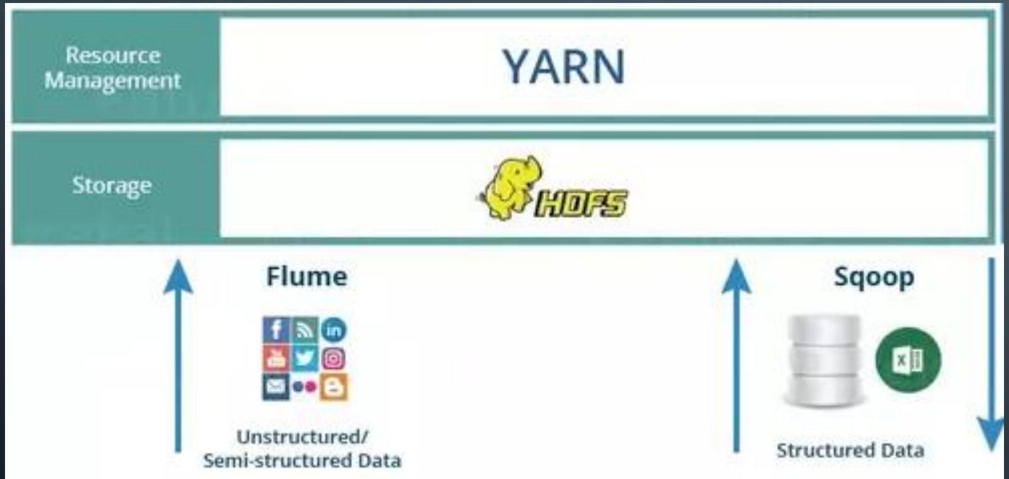
Owners	
Owner_ID	integer
Name	string
Address	string
DateOfBirth	string
Stocks_Owned	string
Total_Stocks	string
Phone	string
Add field	

NOT ONLY
SQL
Redis
HBASE
Cassandra
amazon
dynamo
mongoDB
riak
CouchDB
relax
Tokyo Cabinet
Scalaris
Project Voldemort
Google
bigtable
Neo4j
the graph database
membase



”THE RELATIONAL MODEL IS DEAD”

HADOOP



AMAZON

RELATIONAL VS NON-RELATIONAL

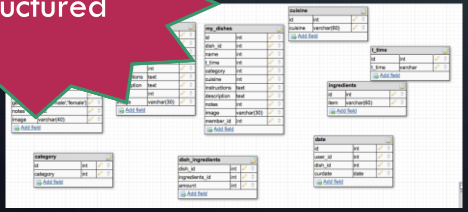
RELATIONAL

- Must be on line (available)
- Must be consistent
  - No duplication!

CAP

”2 of 3”

structured



NON-RELATIONAL

- Must be on line (available)
- Must be partitionable (scalable) (for speed)
  - So we might have to tolerate duplication!

unstructured

TYPES OF DATA

Sales Order table

Name	Product	Quantity	Delivered
John Lee	tablet	5	05/02/2019

structured

semistructured

Invoice	
Delivery address: John Lee 22 Boundary Lane Camberwell	
5 tablets	$595.0
GST	$59.5
Total	$654.5
Due date 16 March 2016	
According to our returns policy, claims have to be made within 2 weeks.	

EMAIL

John Lee’s 5 tablets were sent by truck on 5 February to 22 Boundary Lane Camberwell.

unstructured

STRUCTURED DATA

Sales Order table

Name	Product	Quantity	Delivered
John Lee	tablet	5	05/02/2019

Customer name (VARCHAR, max 30)

Name of product (VARCHAR, max 20)

Number of items (Integer, max 4 digits)

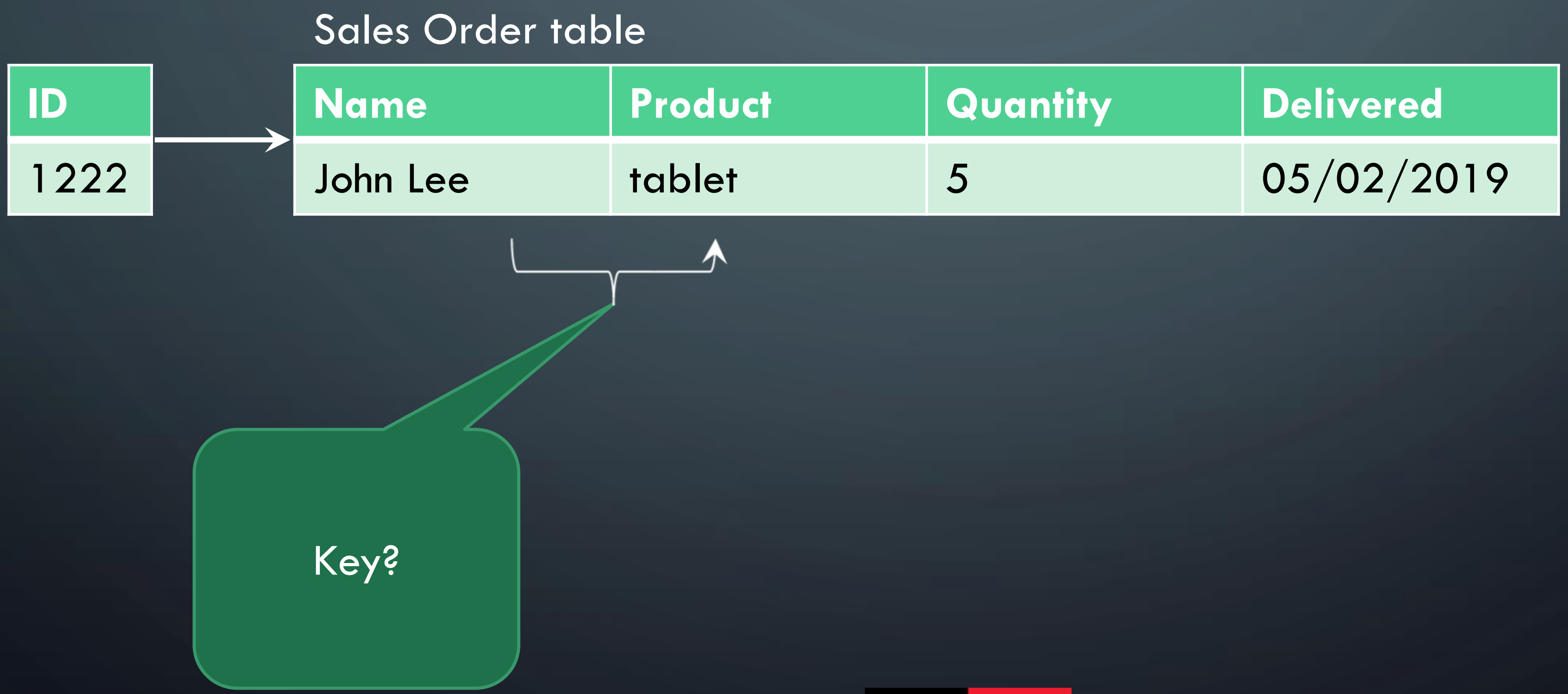
Time of delivery (Date)

EMAIL

John Lee's 5 tablets were sent by truck on 5 February to
22 Boundary Lane Camberwell.

STRUCTURED DATA

- We need a key to identify each tuple



Sales Order table

ID
1222

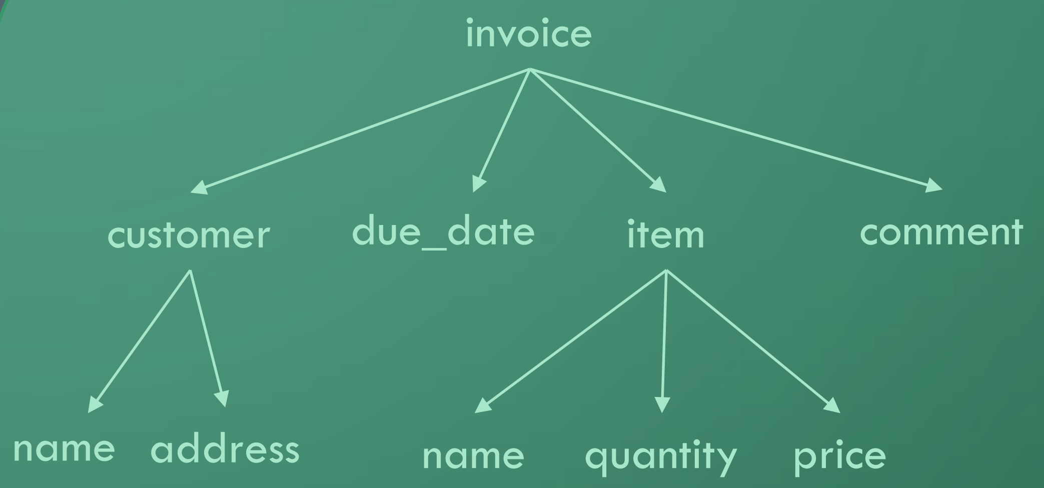
Name	Product	Quantity	Delivered
John Lee	tablet	5	05/02/2019

Key?

SEMISTRUCTURED

```
<invoice>
  <customer>
     <name>John Lee</name>
     <address> 22 Boundary Lane
    Camberwell </address>
   </customer>
   <due_date>28 May 2019</due_date>
     <item>
       <name>Tablet</name>
       <quantity> 5 </quantity>
        <price> 119.99 </price>
     </item>
  <comment>
        Returns within 2 weeks.
     </comment>

</invoice>
```



Tree structure

SEMISTRUCTURED: JSON AND NOSQL

```
{
 "invoice": {
  "customer": {
   "name": "John Lee",
   "address": " 22 Boundary Lane Camberwell "
  },
  "due_date": "28 May 2019",
  "item": {
   "name": "Tablet",
   "quantity": " 5 ",
   "price": " 119.99 "
  },
  "comment": "Returns within 2 weeks."
 }
}
```

mongoDB®

CouchDB
relax

ORACLE®

STRUCTURED VS SEMI-STRUCTURED DATA

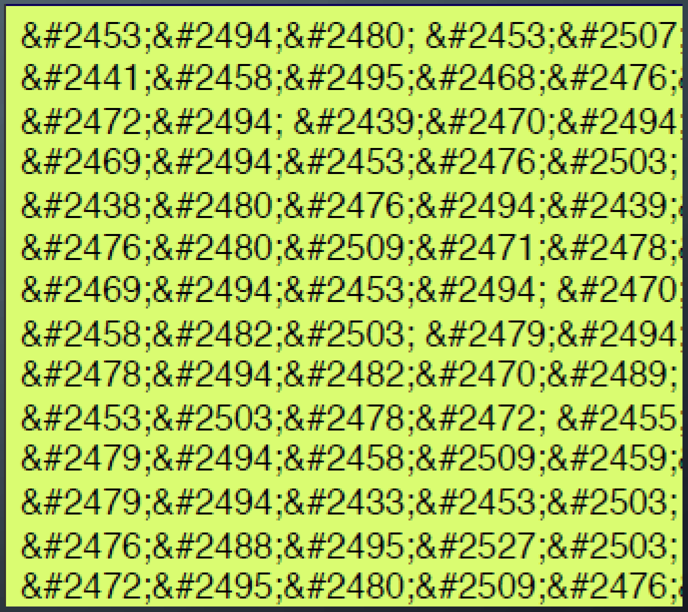
- Structured
  - Hard to create
  - Hard to change
  - Easy to analyse

- Semi-structured (& unstructured)
  - Flexible; easy to create
  - Easy to change
  - Harder to analyse

UNSTRUCTURED

```
180.76.15.31 - - [09/Jun/2015:17:12:08 -0700] "GET /Archive/ HTTP/1.1" 200 1796 "-"
"Mozilla/5.0 (compatible; Baiduspider/2.0; +http://www.baidu.com/search/spider.html)"
"www.redlug.com" 50.118.159.140 - - [09/Jun/2015:17:17:45 -0700] "GET /logs/access.log
HTTP/1.1" 200 178 "http://redlug.com/" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_8_3)
AppleWebKit/536.29.13 (KHTML, like Gecko) Version/6.0.4 Safari/536.29.13" "redlug.com"
61.152.102.40 - - [09/Jun/2015:17:17:51 -0700] "GET /logs/access.log HTTP/1.1" 200 304
"http://redlug.com/" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_8_3) AppleWebKit/536.29.13
(KHTML, like Gecko) Version/6.0.4 Safari/536.29.13" "redlug.com" 220.181.108.115 - -
[09/Jun/2015:17:17:51 -0700] "GET /old_socialistview.htm HTTP/1.1" 200 4516 "-" "Mozilla/5.0
(compatible; Baiduspider/2.0; +http://www.baidu.com/search/spider.html)" "www.redlug.com"
104.209.130.212 - - [09/Jun/2015:17:18:15 -0700] "GET /paper2004JD/0409IraqWar.htm
HTTP/1.1" 200 2965 "http://redlug.com/" "Mozilla/5.0 (Windows NT 6.1; WOW64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/34.0.1847.116 Safari/537.36" "redlug.com"
77.247.181.162 - - [09/Jun/2015:17:21:05 -0700] "GET /logs/ HTTP/1.1" 200 50141
"http://tophamsterporn.com/" "Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/37.0.2062.124 YaBrowser/14.10.2062.12061 Safari/537.36"
"redlug.com" 100.43.81.131 - - [09/Jun/2015:17:22:48 -0700] "GET /robots.txt HTTP/1.1" 200 37
"-" "Mozilla/5.0 (compatible; YandexBot/3.0; +http://yandex.com/bots)" "redlug.com"
23.229.30.164 - - [09/Jun/2015:17:26:44 -0700] "GET /logs/access.log HTTP/1.1" 200 560
"http://redlug.com/" "Mozilla/5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/40.0.2214.85 Safari/537.36" "redlug.com"
```

Text



```
&#2453;&#2494;&#2480; &#2453;&#2507
&#2441;&#2458;&#2495;&#2468;&#2476;
&#2472;&#2494; &#2439;&#2470;&#2494
&#2469;&#2494;&#2453;&#2476;&#2503;
&#2438;&#2480;&#2476;&#2494;&#2439;
&#2476;&#2480;&#2509;&#2471;&#2478;
&#2469;&#2494;&#2453;&#2494; &#2470
&#2458;&#2482;&#2503; &#2479;&#2494
&#2478;&#2494;&#2482;&#2470;&#2489;
&#2453;&#2503;&#2478;&#2472; &#2455
&#2479;&#2494;&#2458;&#2509;&#2459;
&#2479;&#2494;&#2433;&#2453;&#2503;
&#2476;&#2488;&#2495;&#2527;&#2503;
&#2472;&#2495;&#2480;&#2509;&#2476;
```

Picture

NOSQL DATA FORMATS

```
{
 "invoice": {
  "customer": {
   "name": "John Lee",
   "address": " 22 Boundary
   Lane Camberwell "
  },
  "due_date": "28 May 2019",
  "item": {
   "name": "Tablet",
   "quantity": " 5 ",
   "price": " 119.99 "
  },
  "comment": "Returns within 2 weeks."
 }
}
```

```
{
 "invoice": {
  "customer": {
   "name": "John Lee",
   "address": " 22 Boundary
   Lane Camberwell "
  },
  "due_date": “10 August 2019",
  "item": {
   "name": “Display",
   "quantity": “ 10 ",
   "price": " 550.00 "
  },
  "comment": "Returns within 1 week."
 }
}
```

Invoice Collection

The mongo way of doing things

NOSQL DATA FORMATS

mongoDB®

```
{
 "invoice": {
  "customer": {
   “id": “122"
  },
  "due_date": "28 May 2019",
  "item": {
   "name": "Tablet",
   "quantity": " 5 ",
   "price": " 119.99 "
  },
  "comment": "Returns within 2 weeks."
 }
}
```

```
{
 "invoice": {
  "customer": {
   “id": “122"
  },
  "due_date": “10 August 2019",
  "item": {
   "name": “Display",
   "quantity": “ 10 ",
   "price": " 550.00 "
  },
  "comment": "Returns within 1 week."
 }
}
```

Invoice Collection

```
{
   "customer": {
    “id": “122",
    "name": "John Lee",
    "address": " 22 Boundary Lane
    Camberwell "
   }
}
{
   "customer": {
    “id": “123",
    "name": “Sarah Martin",
    "address": " 11 Daniell Pl Kew "
   }
}
```

Customer Collection

NOSQL DATA FORMATS

mongoDB®

unstructured

```
{
  “email":
       “Dear Sarah, I have sent the tablets you
       requested. If you are still experiencing
       problems, return them to our depot within
       15 days for a full refund. Having said that,
       everyone knows these tablets aren’t very
       reliable and won’t work for very long, so
       you better close your company when you’ve
       sold them. Kind regards, Jeff"

}
```

```
{
  “email":
       “Hi Greg, we haven’t heard from you in a
       while, are you still in the IT retail business?
       We have some tablets on special.
       Also, we have started direct imports from
       China, which might bring some bargains.
       Cheers, Dan"
}
```

+ text indexing

+ MapReduce

Email Collection

CASE STUDY

Where to store the data?

CASE STUDY

- A car manufacturer naturally keeps track of all sales, models, customers.
  - This is vital data that must not be lost.
  - If we lose it, the company no longer knows where the cars went and where the money came from.

Tax office fines us if we lose this

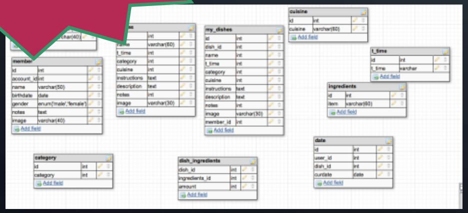
- A car has lots of sensors. Sensor data is valuable to the manufacturer to find out why/when cars fail.
  - There is heaps of this data.
  - If some of it goes missing, no one cares.

This model can't stand stop and go

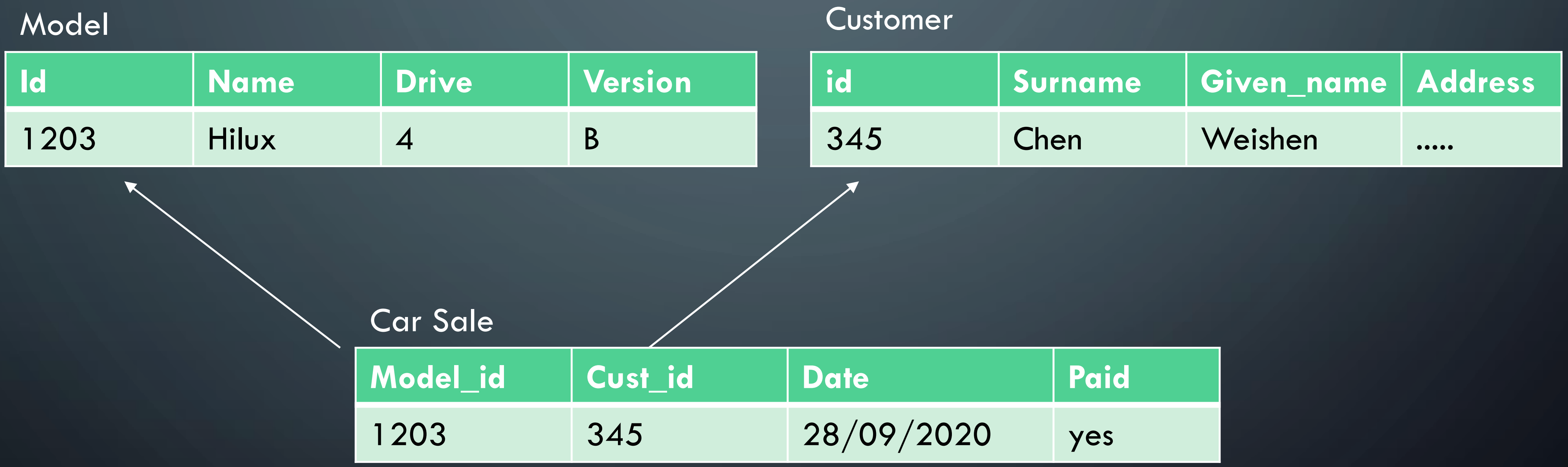
structured

integrate

unstructured



CASE STUDY – BUSINESS DATABASE



Model

Id	Name	Drive	Version
1203	Hilux	4	B

Customer

id	Surname	Given_name	Address
345	Chen	Weishen	…..

Car Sale

Model_id	Cust_id	Date	Paid
1203	345	28/09/2020	yes

Normalising / denormalising

CASE STUDY – SENSOR DATABASE

Sensor signal

Id	Gps pos	Speed	Dist	Brake fluid	Direction	Dashcam	Petrol	Brake pads	Alerts
10001		35			NNE				
10002	-85.565			73					
10303	-85.634		205		SSE			22mm	
19332						#233,#235,#133			
20063	-85775						285		

CASE STUDY AND STORAGE TECHNOLOGY

Technology	Business data	Sensor data
Oracle	Great solution, good for consistency. No redundancy.	Scalability might be slow, format might be a problem if not very uniform.
Oracle + MongoDB	Using Oracle: good for consistency, and a good option if the data is already relational.	Using MongoDB: Works well for large volumes and varying formats as well as missing data.
MongoDB	Business data has to be migrated. The question of using embedded document versus document links has to be addressed. Consistency might be affected.	Works well for large volumes and varying formats as well as missing data.
Oracle + Hadoop	Using Oracle: good for consistency, and a good option if the data is already relational.	Using Hadoop: Works well for large volumes and varying formats as well as missing data. Can also work on streaming data.
Hadoop	Hadoop can maintain consistency and speed with structured data. Integrates well using Yarn.	Hadoop well for large volumes and varying formats as well as missing data. Can also work on streaming data.

CASE STUDY – SENSOR DATABASE

```
{
 “output": {
    “id”: “10001”,
   “speed”:35”,
   ”direction”: ”NNE”
  },
 “output": {
    “id": “10002",
    “gps": “-85.565",
    “brake_fluid": “73"
  },
“output": {
    “id": “10303",
    “gps": “-85.634",
    ”dist”:”205”
    “brake_pads": “22mm"
  },
“output": {
    “id": “10303",
    “gps": “-85.634",
    ”dist”:”205”
    ”direction”: ”SSE”,
    “brake_pads": “22mm"
  },
  ……..
}
```

Every entry can have different attributes

STORAGE

Popular Technologies

APACHE CASSANDRA

cassandra

- Column store
  - Mix between table and key-value
- Very fast (linear speed increase)

cannot do joins

```
CREATE COLUMNFAMILY person (id text, name text, city text, PRIMARY KEY(id));

INSERT INTO person (id, name, city) VALUES ('1', 'Ravinder Singh', 'New Delhi');
```

supports MapReduce

CQL

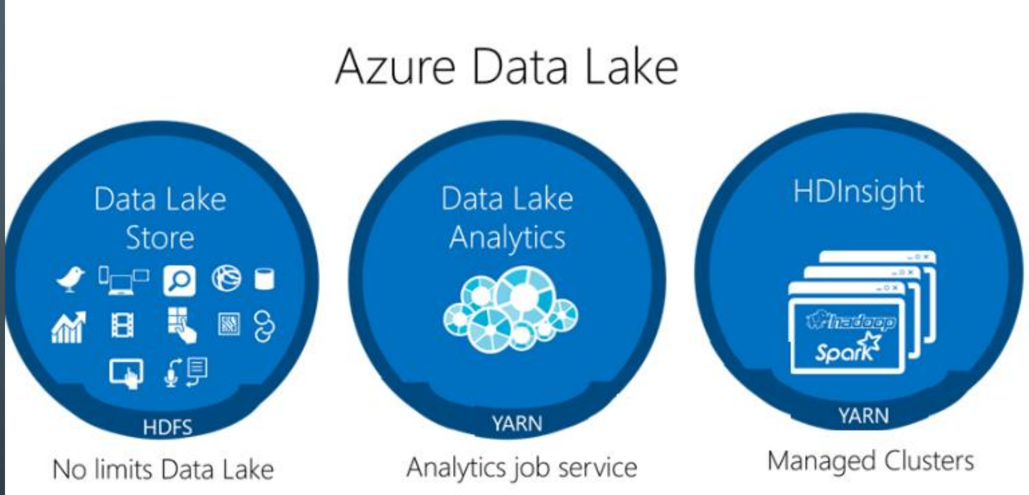
APACHE COUCHDB

- Focus on replication and durability
  - = data safety
- Document database
  - JSON
- Multi-version concurrency control
  - Consistency + throughput

supports MapReduce

DATA LAKE

- File system for diverse file types
- Based on distributed file system
- Multiple files can be accessed at the same time
- Files can be temporary



This Photo by Unknown Author is licensed under CC BY-SA

ON-READ STRUCTURING

- Hadoop Hive can turn a tabular file (like .csv) into a structured table in one command!
  - Analyse and abandon
  - Analyse and store

SUMMARY

- Big Data requires special infrastructure for fast computation and efficient storage.
- Cloud storage services offer scalability and flexible pricing.
- Different types of data sets require different storage formats.
- Relational DBs offer consistency and availability, and extraction in new combinations at the expense of scalability.
- Non-relational DBs are highly scalable but not necessarily consistent if optimised for speed.

BIG DATA

INTRODUCTION TO CLOUD – MICROSOFT AZURE

WHAT IS MICROSOFT AZURE

Microsoft Azure

- Complete Cloud platform
  - Databases
  - Data lakes
  - Analytical tools
  - Batch processing tools
  - Web server
  - Integration tools
  - …..

- Cloud Service Providers
  - AWS 35%
  - MS Azure 16%
  - Google Cloud 9%
  - Alibaba Cloud 4%
  - IBM Bluemix 4%
  - ...and a few others

WHAT ARE THE '..AAS'ES?

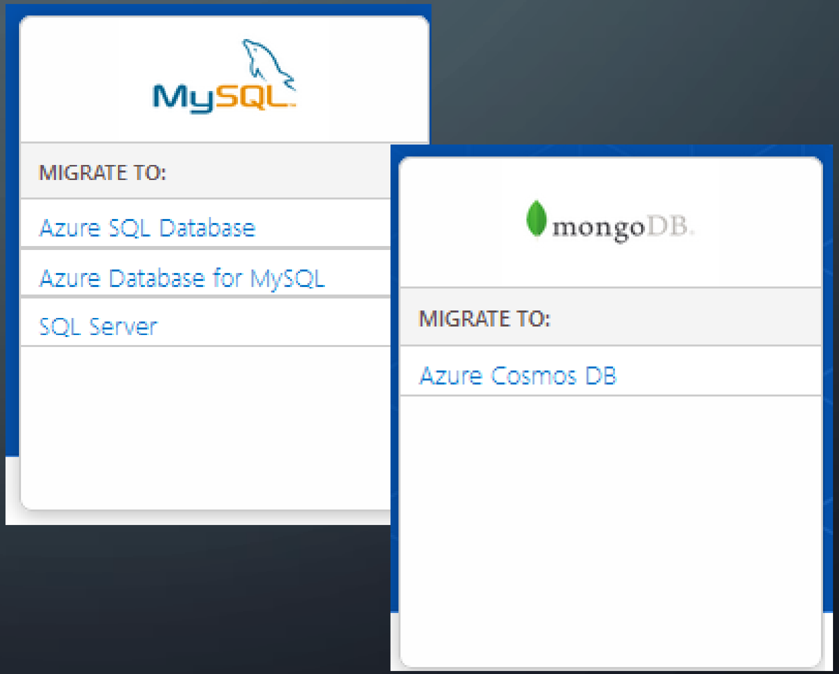
- IaaS – Infrastructure as a Service
  - As if you bought a laptop without an operating system (just a lot bigger).
- PaaS – Platform as a Service
  - As if you bought a laptop with an operating system, ready to install all the nice programs you want to use (just a lot bigger).
- SaaS – Software as a Service
  - As if you bought a laptop perfectly configured with all applications you need (just a lot bigger).

DATABASES IN THE CLOUD

AWS

- Aurora
- Relational Database Service
- RDS on VMware
- DynamoDB
- ElastiCache
- Neptune
- Can be deployed:
  - Mongo, Oracle, SQL Server, Couchbase

AZURE



MIGRATE TO:

- Azure SQL Database
- Azure Database for MySQL
- SQL Server

MIGRATE TO:

- Azure Cosmos DB

FILE STORAGE

AMAZON S3
SIMPLE STORAGE SERVICE

- Like a file system – can store any type of file
- High durability – replicated over several servers
- Safety – encryption offered at upload
- Computing power close to the data
- ’Data Lakes’
- Integration of structured and unstructured data ad-hoc for analysis

AZURE FILES / AZURE BLOB STORAGE / AZURE DATA LAKE STORAGE GEN1/GEN2

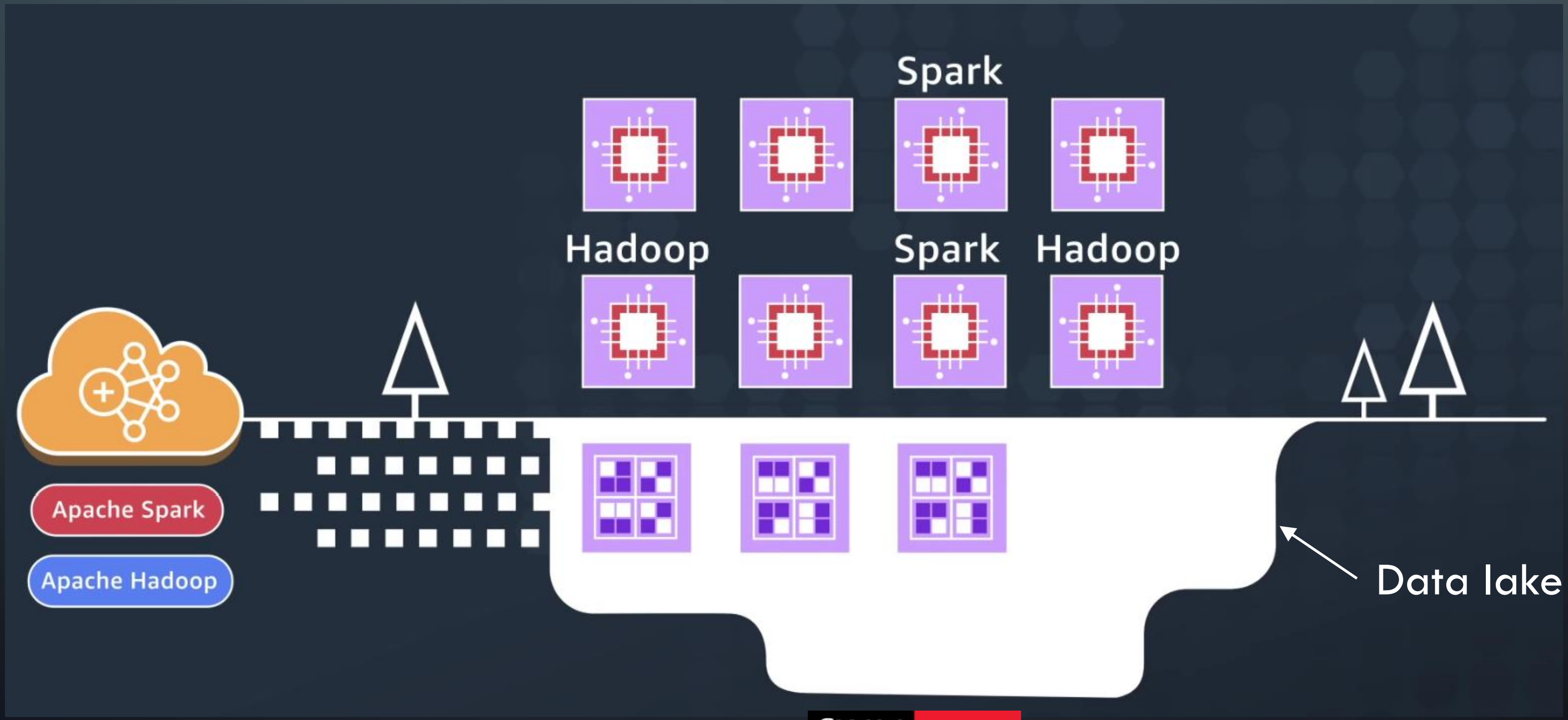
- Azure Files is a file sharing system
    - Designed for synchronisation with hard disk
- Azure Blob Storage (WASB)
    - Staging area for ETL
- ADLS
    - Same features as S3 (roughly)

WINDOWS AZURE STORAGE BLOBS (WASB)

- HDFS runs on a Hadoop cluster
  - Hadoop clusters need nodes (resources)
  - In the Cloud, you pay for resources.
- WASB is built on HDFS
  - You can store your data there
  - then start a cluster for an analysis task
- WASB data persists after the cluster is deleted

HADOOP

Amazon EMR (Elastic MapReduce)   Azure HDInsight



Source: Amazon

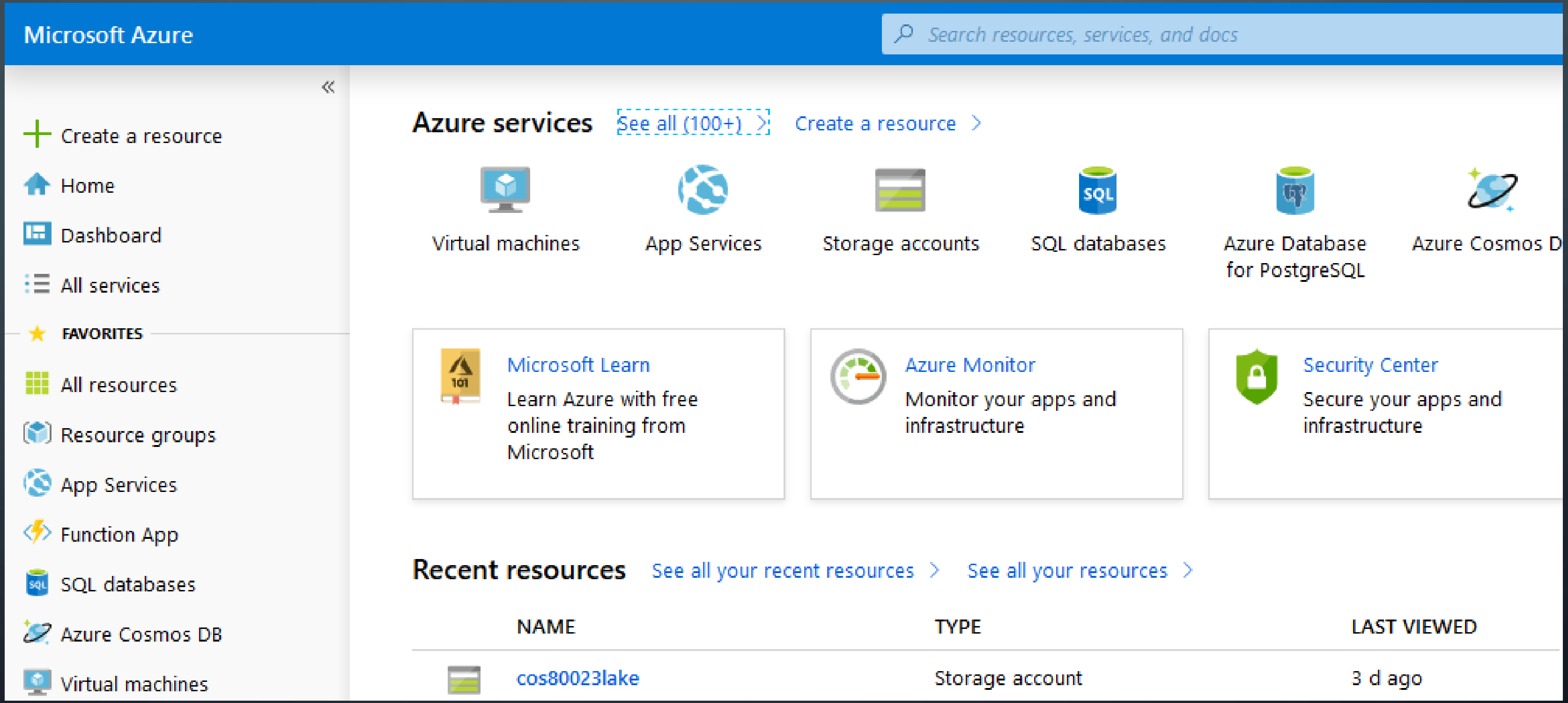
AZURE TECHNOLOGIES WE WORK WITH

- SQL DB
- WASB
- Data lake
- Hadoop Cluster

- MapReduce
- Hive
- Pig
- Sqoop

Hadoop ↔ HDInsight

LET'S HAVE A LOOK AT THE AZURE PORTAL..



Microsoft Azure

Search resources, services, and docs

- Create a resource
- Home
- Dashboard
- All services

FAVORITES

- All resources
- Resource groups
- App Services
- Function App
- SQL databases
- Azure Cosmos DB
- Virtual machines

Azure services   See all (100+)   Create a resource

Virtual machines   App Services   Storage accounts   SQL databases   Azure Database for PostgreSQL   Azure Cosmos D

Microsoft Learn
Learn Azure with free online training from Microsoft

Azure Monitor
Monitor your apps and infrastructure

Security Center
Secure your apps and infrastructure

Recent resources   See all your recent resources   See all your resources

NAME	TYPE	LAST VIEWED
cos80023lake	Storage account	3 d ago