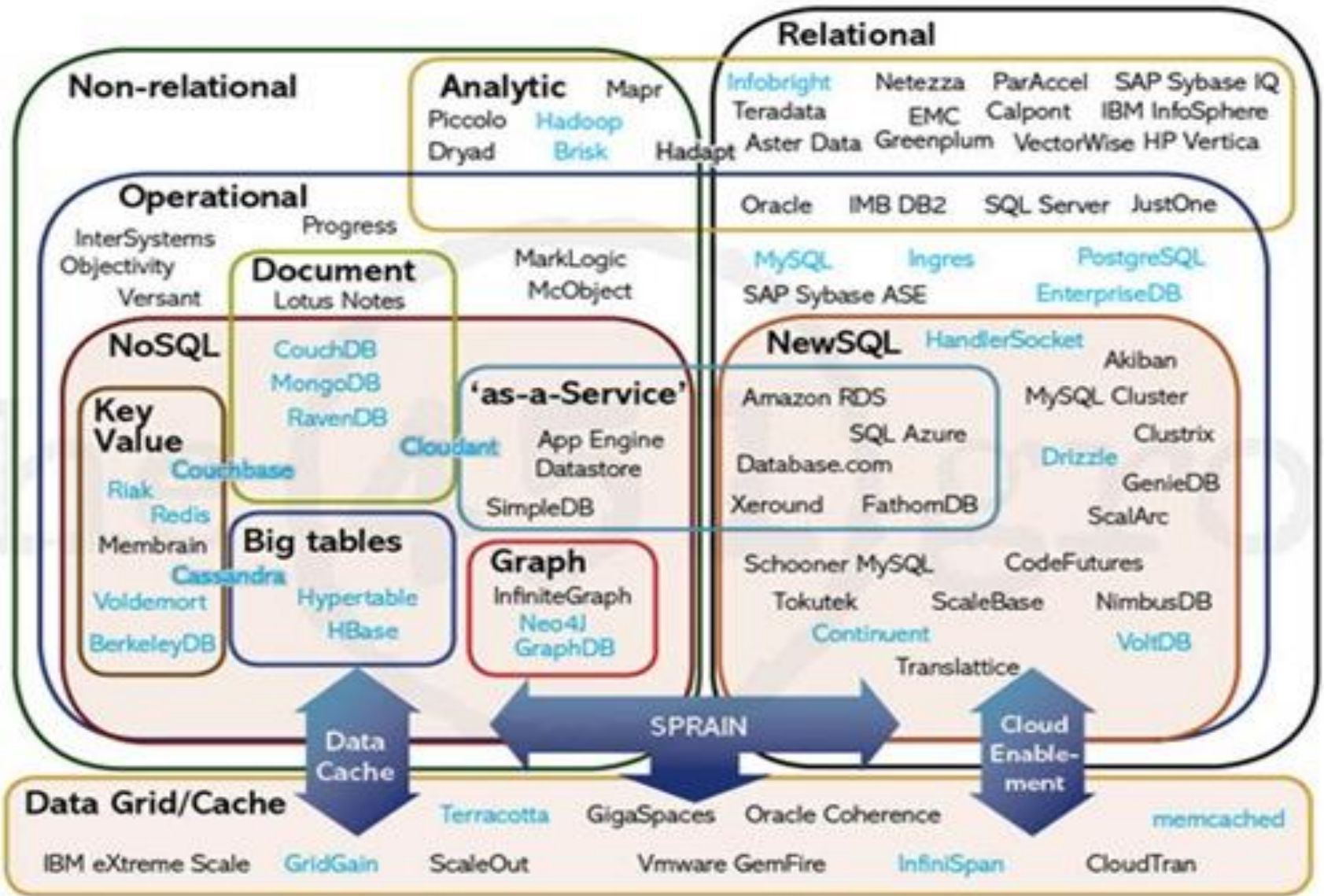


NoSQL (examples)

Sources:

Pramod J. Sadalage and Martin Fowler
*NoSQL Distilled: A Brief Guide to the
Emerging World of Polyglot Persistence*,
Pearson Education, 2013

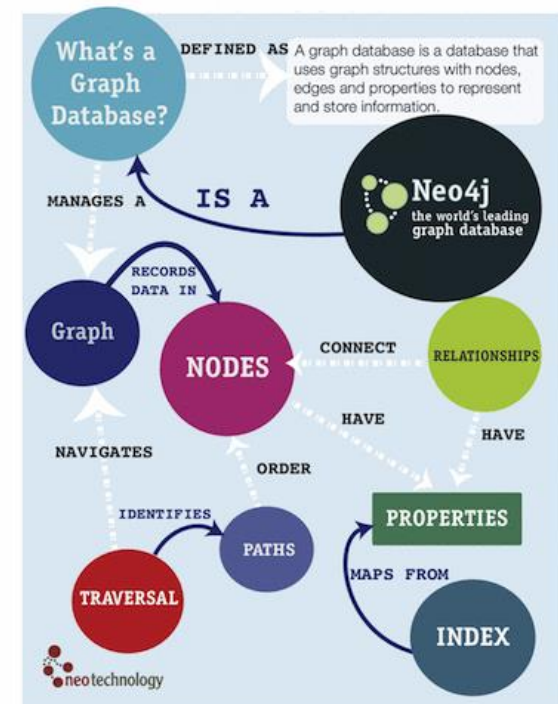


Types of NoSQL

- Method for storing/retrieving data in a non-tabular relation format.
 - Graphs
 - Key/Value
 - Documents

Graph NoSQL DBs

- Good for representing networks (social/industrial)
- Query graphs DBs with traversals.
- Repeated traversals indexed for optimization.



Graph NoSQL DBs

Examples:

- Neo4j
- DEX
- InfiniteGraph
- FlockDB

Key/Value NoSQL Store

- Way for storing data in a schema-less way.
- Key created for each record.
- Bin created for each piece of data.
- Each bin has key/value.
- No data - no bin

Example of unstructured data for user records

Key: 1	ID: sj	First Name: Sam
-----------	--------	-----------------

Key: 2	Email: jlb@gmail.com	Location: London	Age: 37
-----------	-------------------------	---------------------	------------

Key: 3	Facebook ID: jkirk	Password: xxx	Name: James
-----------	-----------------------	------------------	----------------

Key/Value NoSQL Store

Examples:

- Aerospike
- LevelDB
- Tarantool

Document style NoSQL DBs

- Notion of storing information in a document.
- Data is encapsulated.
- Can be stored in XML/JSON/BSON
- Easy to map objects in code to documents in a DB.

Document style NoSQL DBs

MongoDB (Cross platform)

- Different data types stored in different collections(like tables)
- Stores data in JSON-like format.
- Extensible queries(and, or, in, not, regex, etc...)
- Data very object-like
- Can have nested data
- No joins
- No transactions

Document style NoSQL DBs

RavenDB (.NET)

- Much like Mongo, documents stored in collections
- Collections are indexed to increase performance
- Query language is lucene
- Client API or RESTful requests
- Comes with built in RavenStudio
- Is said to fully supports ACID

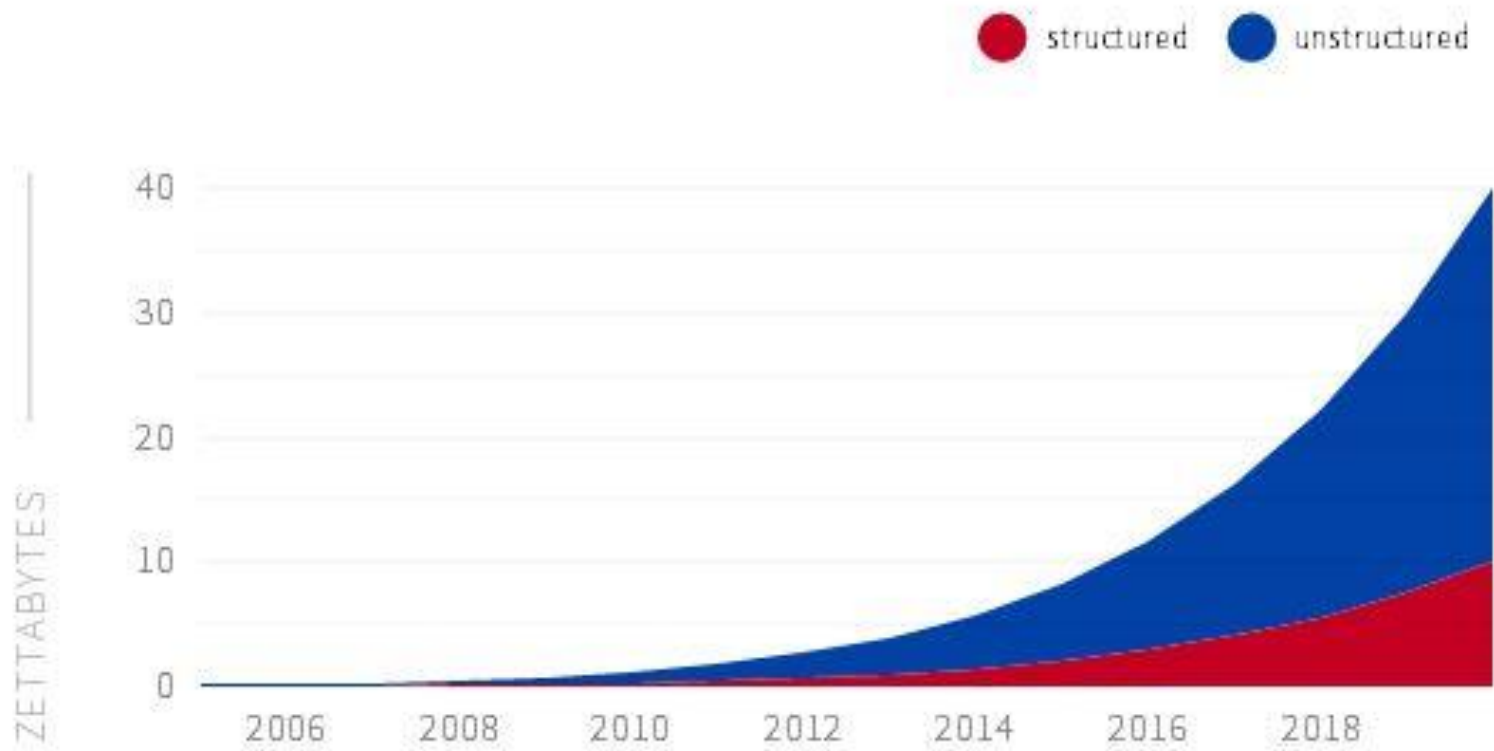
Advantages to using NoSQL

- Queries are very fast
- Less overhead (easier to deploy)
- No need to develop schemas (flexible)
- Cost effective and mainly open source
- Data can be easier to visualize
- More object-oriented
- Scale out instead of up

Disadvantages of NoSQL

- ACID not ensured
- Query languages vary
- Usually doesn't support joins
- Very narrow focus(mainly data storage)
- Performance > Consistency
- Lack of maturity compared to relational databases

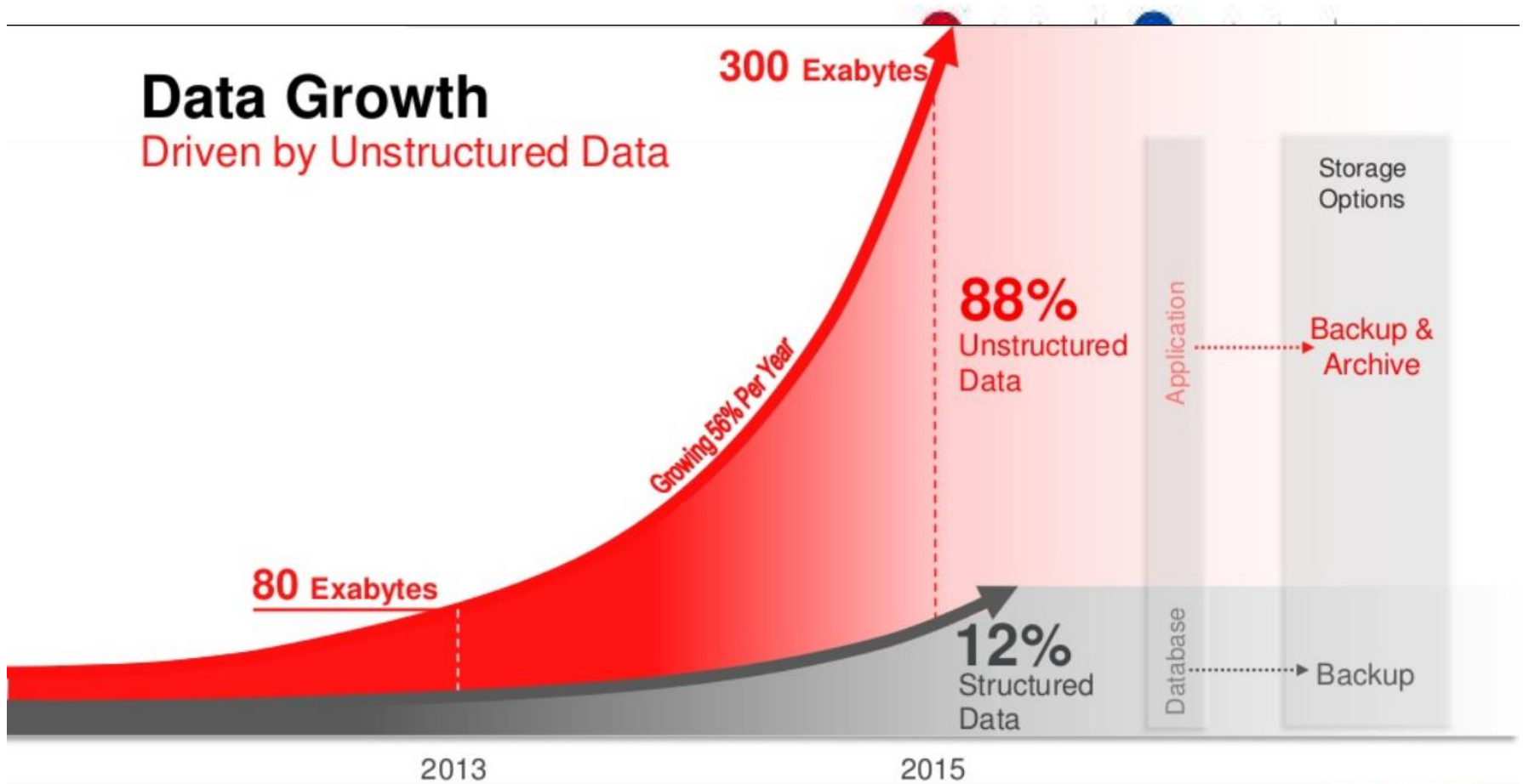
Data predictions



Data predictions

Data Growth

Driven by Unstructured Data



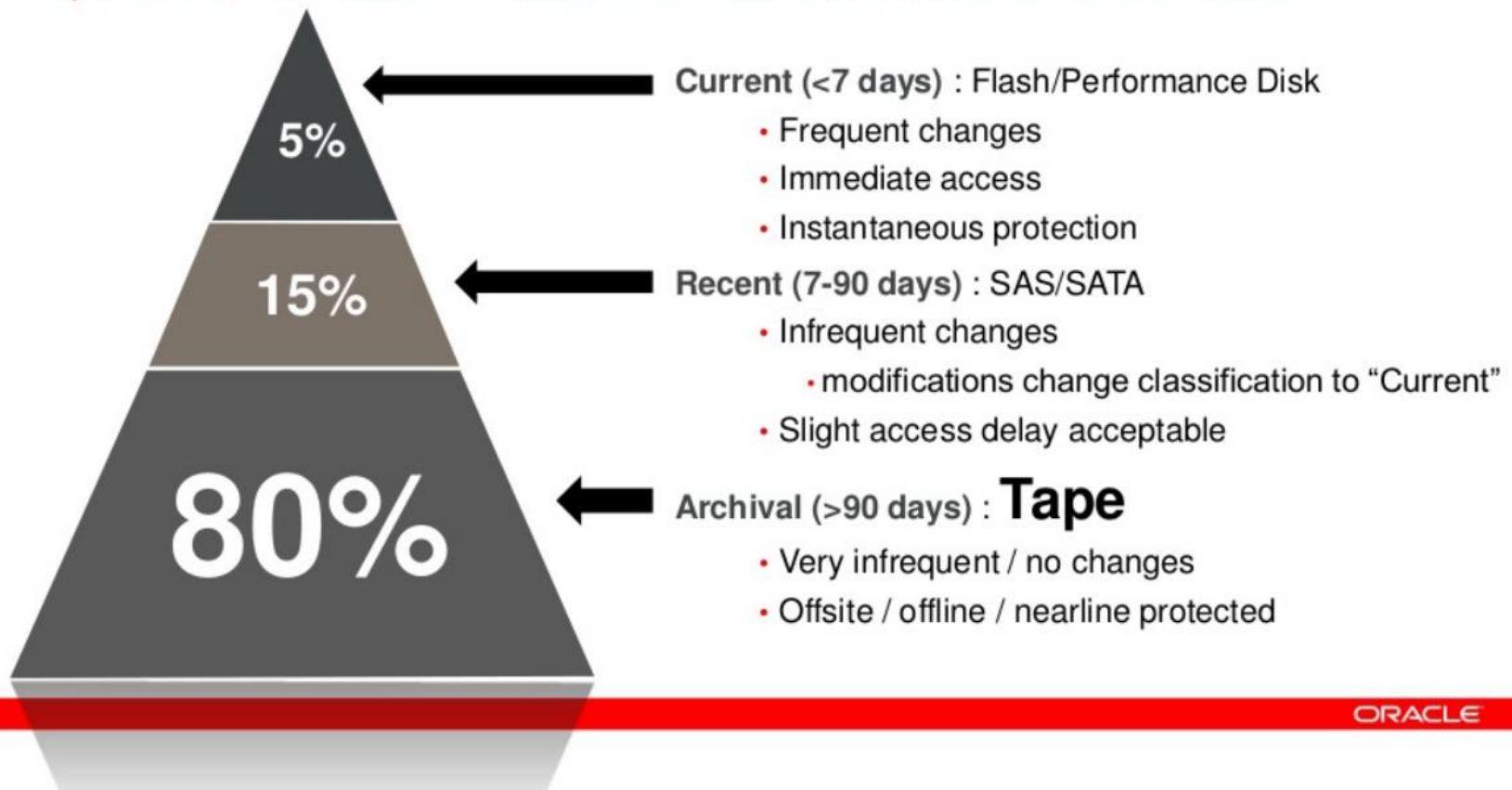
ORACLE

* Source: ESG Digital Archive Market Forecast

Data (side issue)

Economics of Tiered Storage

Tape is the Foundation: Most of the Data Stored at the Lowest Cost



Enterprise dilemma

- As enterprises grow, their data needs also grow.
- Continued dealings with extremely sensitive data (financial, commerce, etc.).
- Need the ability to scale and maintain ACIDity.
- SQL can be expensive to scale.

“NewSQL”(?)

- An answer to the enterprise dilemma.
- Emerging middle ground between SQL and NoSQL.
- Main types
 - New architecture
 - General-purpose
 - In-memory
 - SQL Engines
 - Transparent Sharding