Object Storage: Storage for Developers

Michael Factor, Ph.D.
IBM Fellow, Storage and Systems
IBM Research – Haifa

source: https://pixabay.com/en/tunisia-desert-sand-dune-pink-2740217/

source: https://pixabay.com/en/gold-bar-gold-bar-gold-bullion-296115/

Need to

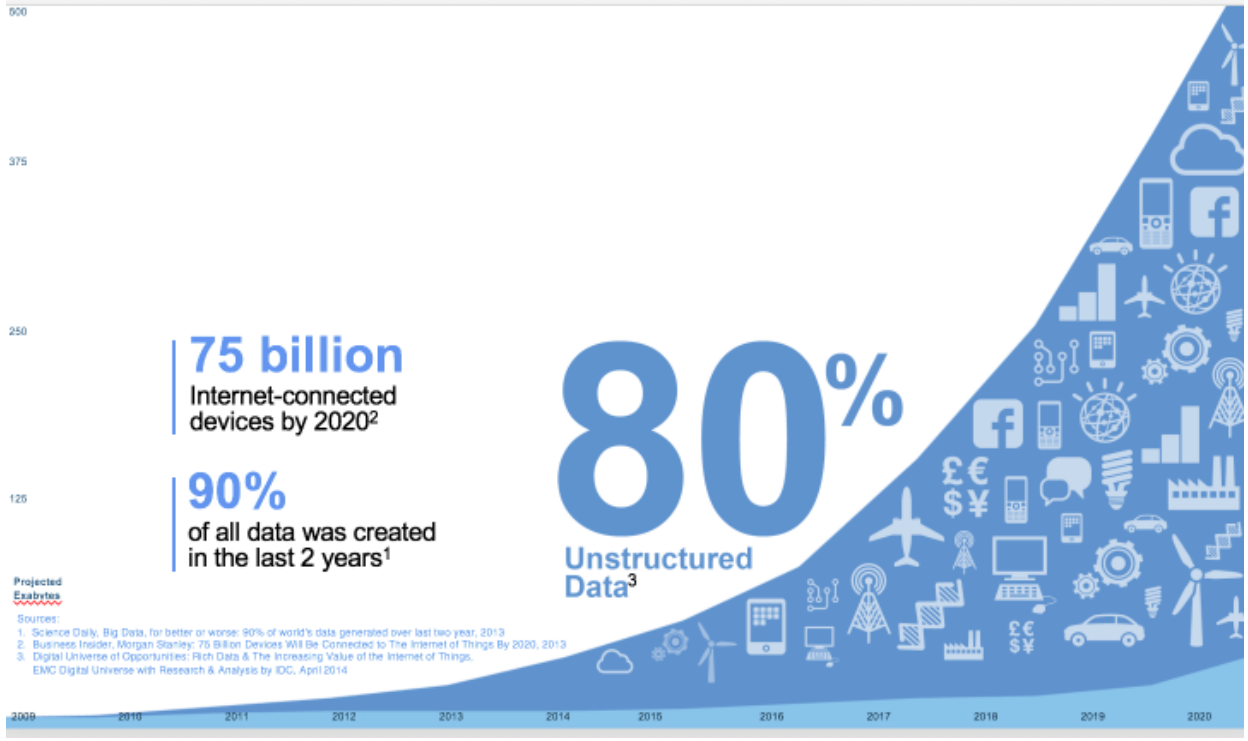
- Store
- Manage
- Protect
- Secure

while addressing

- Scale
- Cost

the data enabling developers to

- Collect
- Clean/transform
- Analyze



How should we do this?

And the answer is . . .

. . . Object Storage

What is object storage?

Block, File and Object

- **Block**: An array of bytes
- **File**: Explicitly managed hierarchy of randomly accessed blobs
- **Object**: Key-value (object)

Typical object storage features

- Buckets containing keys for objects
  - Hierarchy is in eyes of beholder
- RESTful (HTTP) access
- All or nothing atomic writes – no update in place
- Data with metadata
- Secure in flight and at rest
- Designed for scale out and durability

- Ideal for unstructured data and batch rectangular data

Designed for developers

- Simple, RESTful API
- Atomic operations
- Globally accessible
- “Limitless”

https://pixabay.com/en/man-adult-businessman-laptop-1459246/

How the APIs vary

Block

- READ
- WRITE
- FORMAT
- . . .

Table 62 — WRITE (10) command

Byte\Bit	7	6	5	4	3	2	1	0
0	OPERATION CODE (2Ah)							
1	WRPROTECT			DPO	FUA	Reserved	FUA_NV	Obsolete
2	(MSB)							
5	LOGICAL BLOCK ADDRESS							(LSB)
6	Reserved			GROUP NUMBER				
7	(MSB)							
8	TRANSFER LENGTH							(LSB)
9	CONTROL							

http://t10.org/ftp/t10/document.05/05-344r0.pdf

File

- OPEN
- CLOSE
- RENAME
- WRITE
- . . .

```
fd=open(”tmp.tmp”, O_WRONLY);
for(i=0;i<LIMIT;i++)
   write(fd,
         buffer[i*WRITE_SZ],
         WRITE_SZ);
close(fd)
rename(”tmp.tmp”, “real.name”)
```

Object

- PUT
- GET
- HEAD
- POST

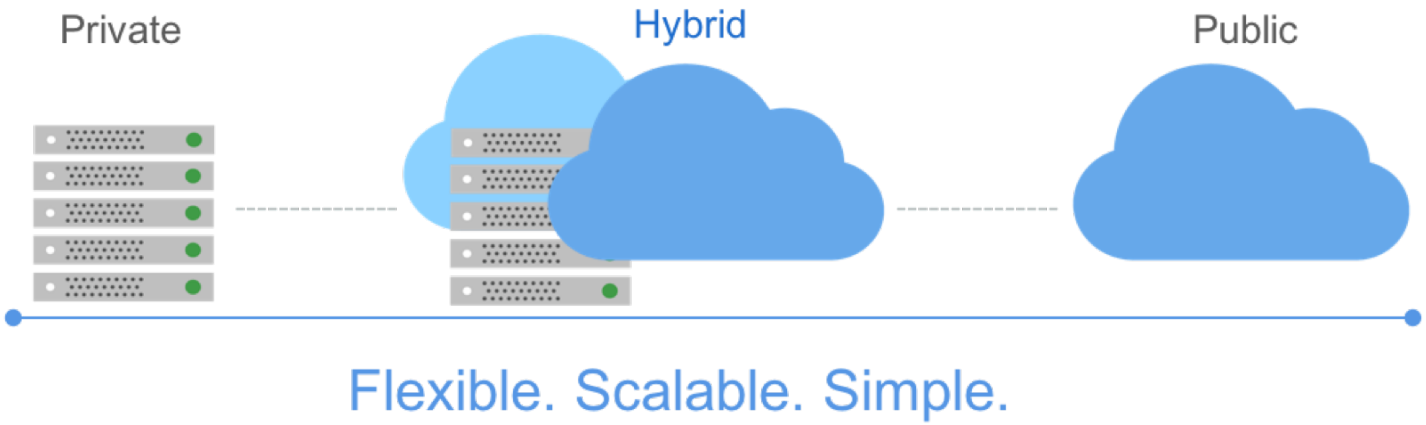
```
PUT /bucket/object HTTP/1.1
Authorization: {auth}
Content-MD5: 3097216...
Host: …storage.softlayer…
Content-Length: 533
```

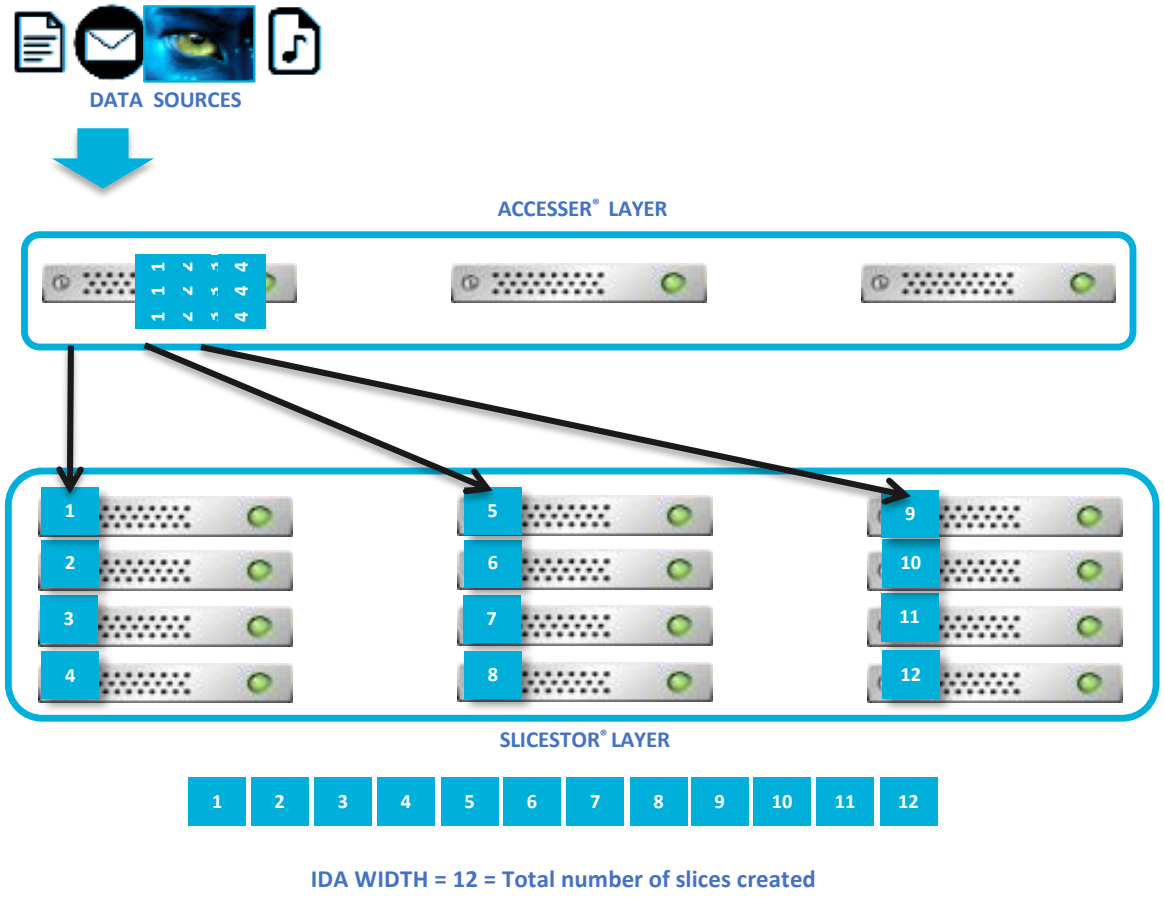
The 'queen' bee . . .

Under the covers of one object store: IBM Cloud Object Storage

IBM Cloud Object Storage

- Two tiered, fully distributed, architecture
  - Can be deployed in multiple data centers – survive a data center outage
- Distributed erasure coding to protect the data
- RESTful protocol for data access (S3-compatible)
- Security via AONT-RS (All Or Nothing Transform-Reed Solomon)

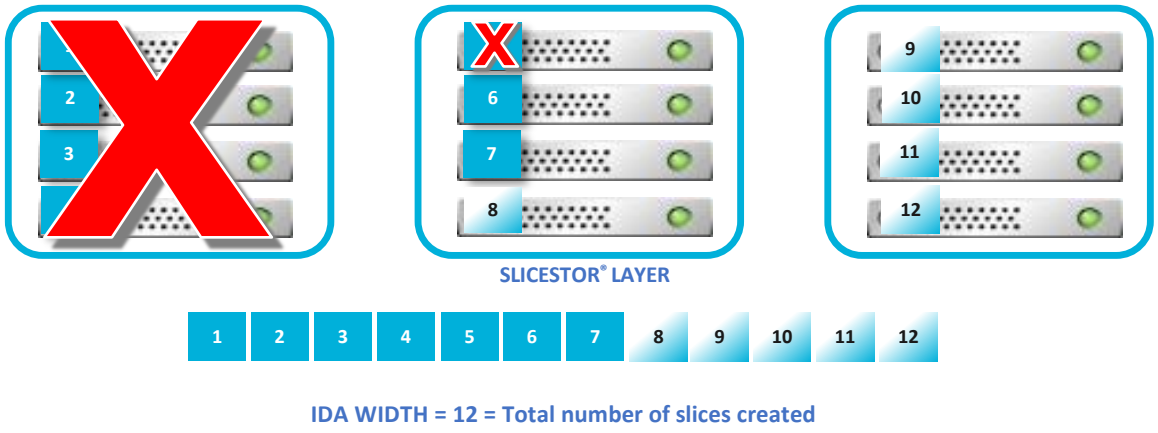




1 Data is encrypted, and sliced using Information Dispersal Algorithms (IDA).

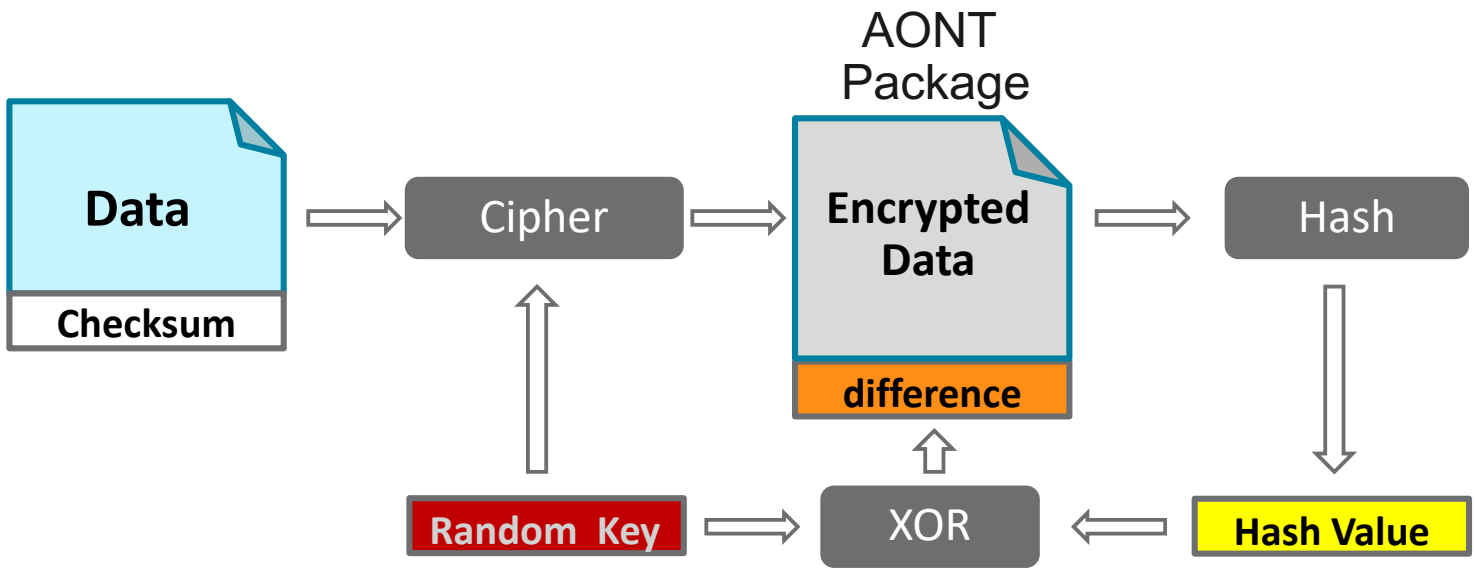
2 Slices are dispersed to separate disks, storage nodes and/or geographic locations.

- With a 7+5 RS encoding, can read data from any 7 slices
  - If distributed over three data centers, can lose an entire data center with no loss of data or access
  - Space overhead of 71% as compared to 200% with triplication



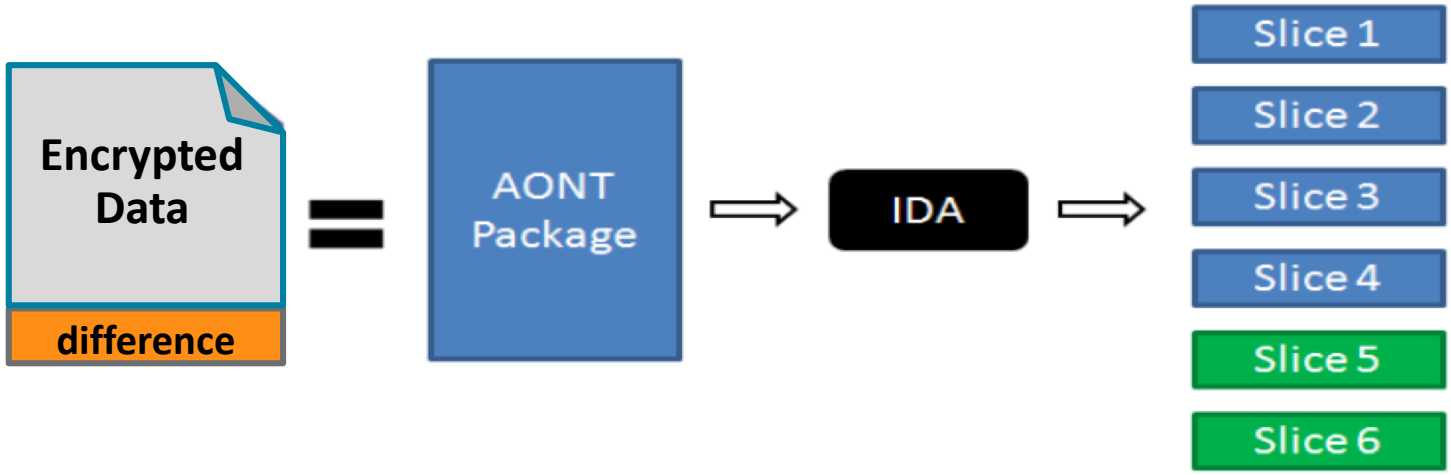
AONT-RS: Keyless Encryption

All Or Nothing Transform – Reed-Solomon



https://www.usenix.org/conference/fast11/aont-rs-blending-security-and-performance-dispersed-storage-systems

AONT-RS: Keyless Encryption
All Or Nothing Transform – Reed-Solomon



- Without a threshold number of slices, cannot calculate hash and thus cannot separate key out of **difference** which is XOR of key and hash

https://www.usenix.org/conference/fast11/aont-rs-blending-security-and-performance-dispersed-storage-systems

Putting data to work

Need to

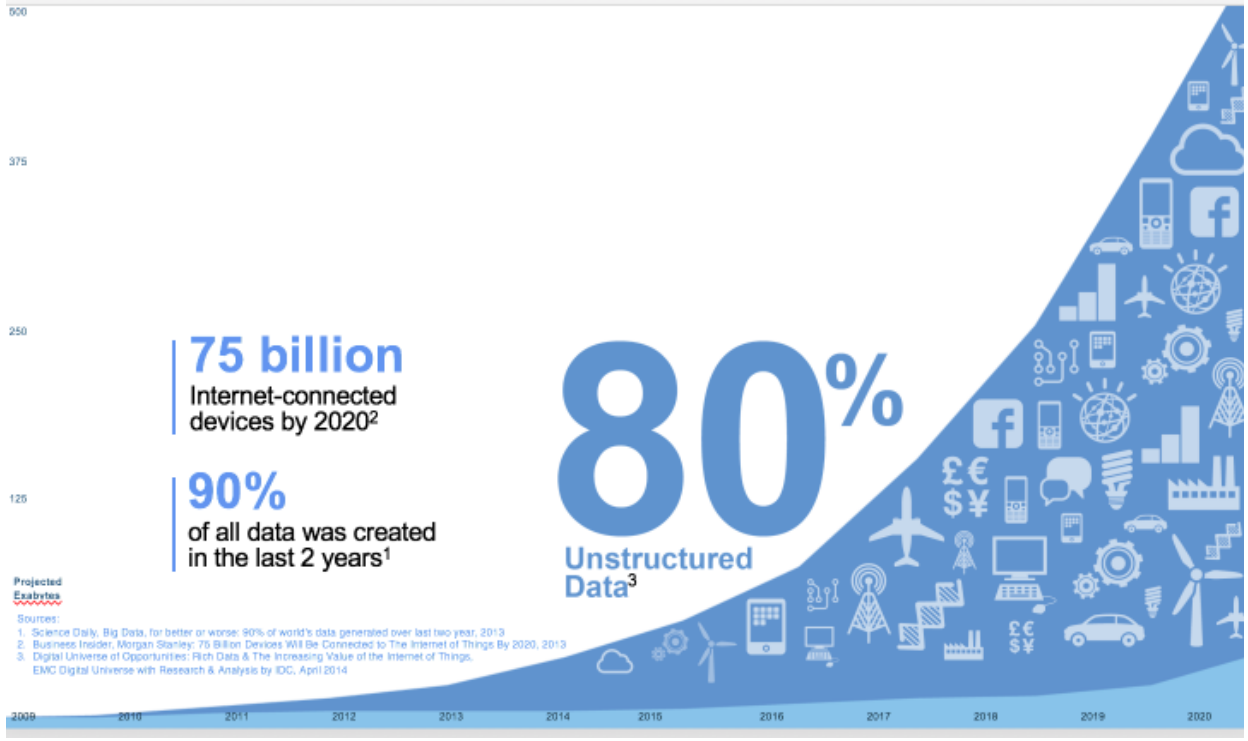
- Store
- Manage
- Protect
- Secure

while addressing

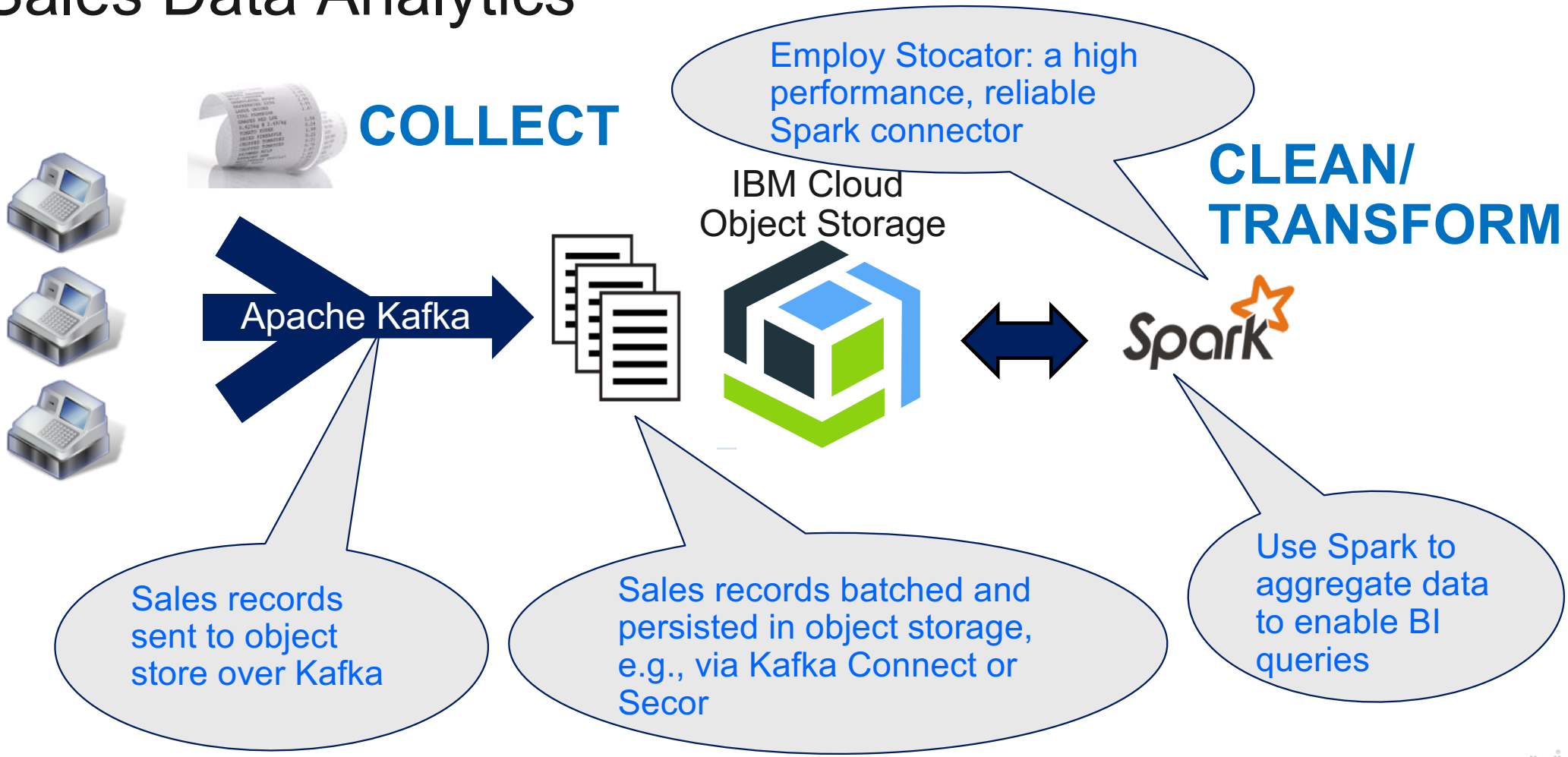
- Scale
- Cost

the data enabling developers to

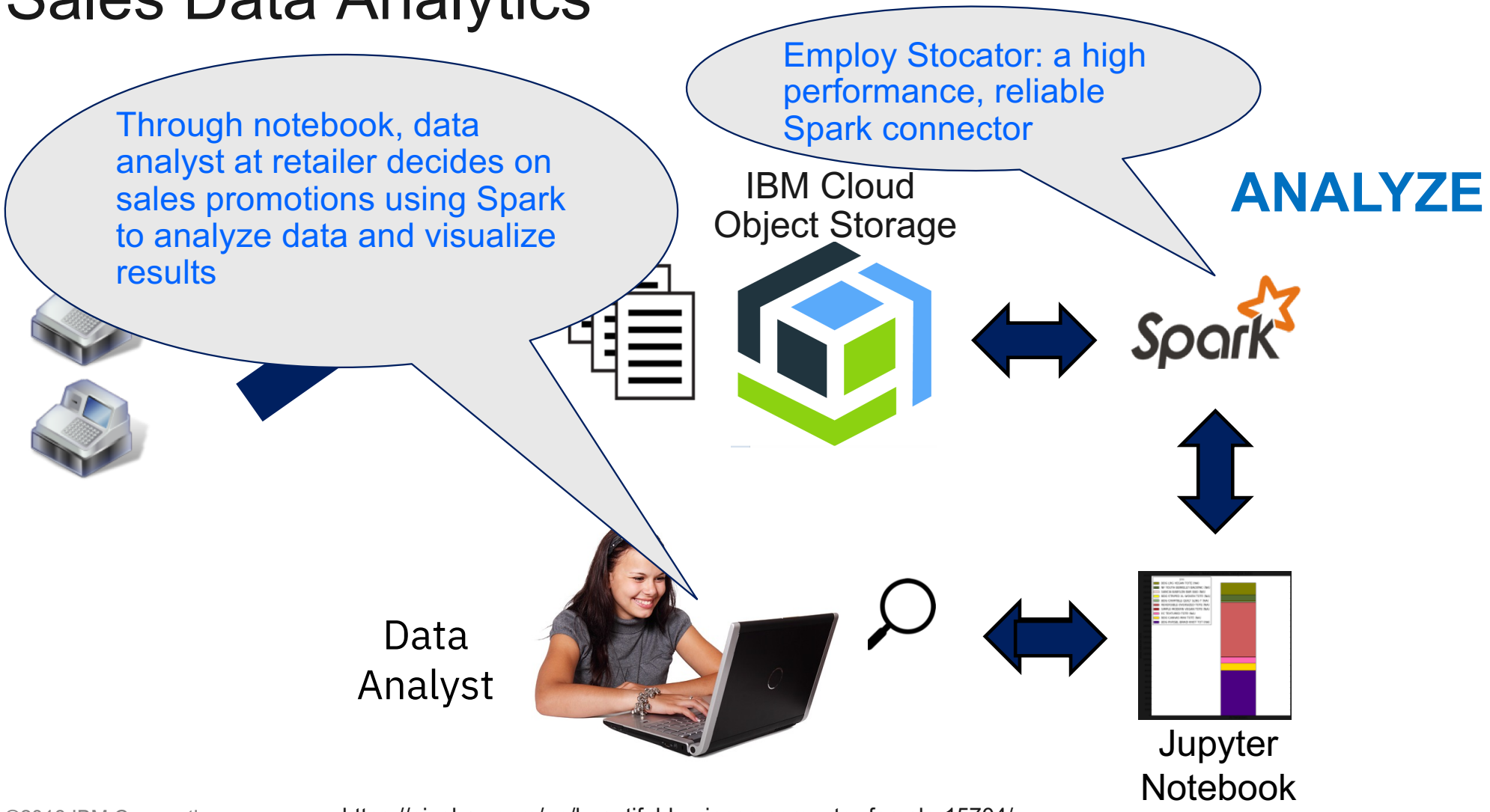
- Collect
- Clean/transform
- Analyze



Sales Data Analytics



Sales Data Analytics



https://pixabay.com/en/beautiful-business-computer-female-15704/

A right way and a wrong way to use object storage

Wrong Way

Pretend it is a file system

Emulate design patterns such as write to temp file and rename to prevent partial data

Create empty objects to represent directories

Right Way

Leverage object storage semantics and scale

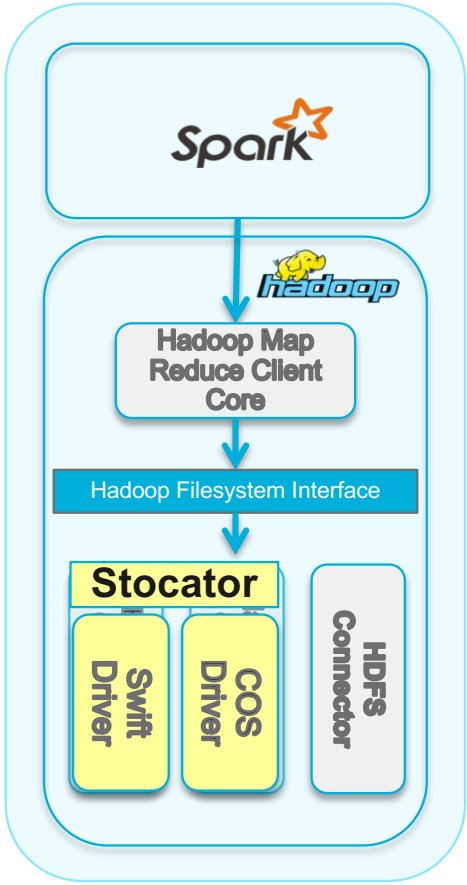
Use atomicity of PUTs to prevent partial data

Just create objects with hierarchical name

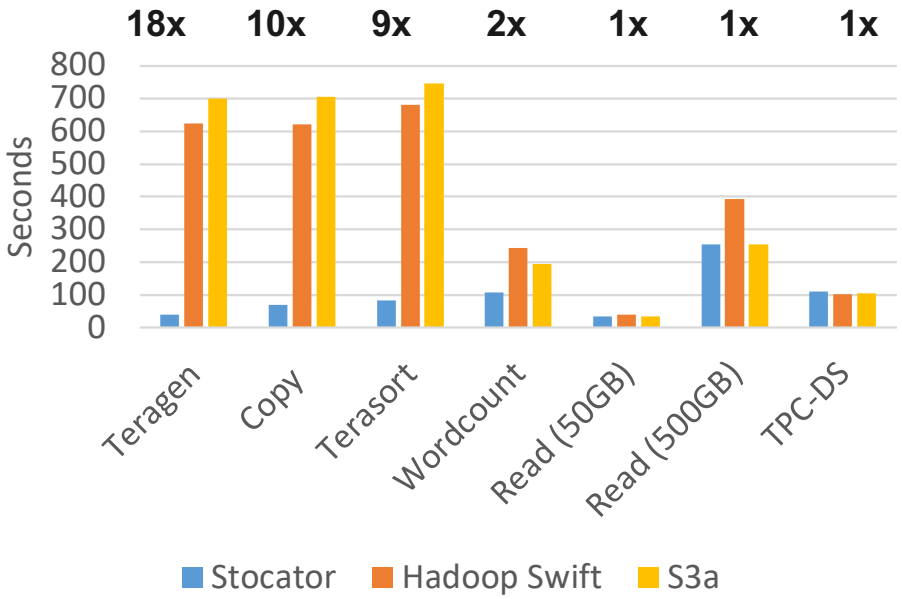
https://pixabay.com/en/shop-store-open-shopping-retail-2607121/
https://pixabay.com/en/wrong-way-sign-road-caution-167535/

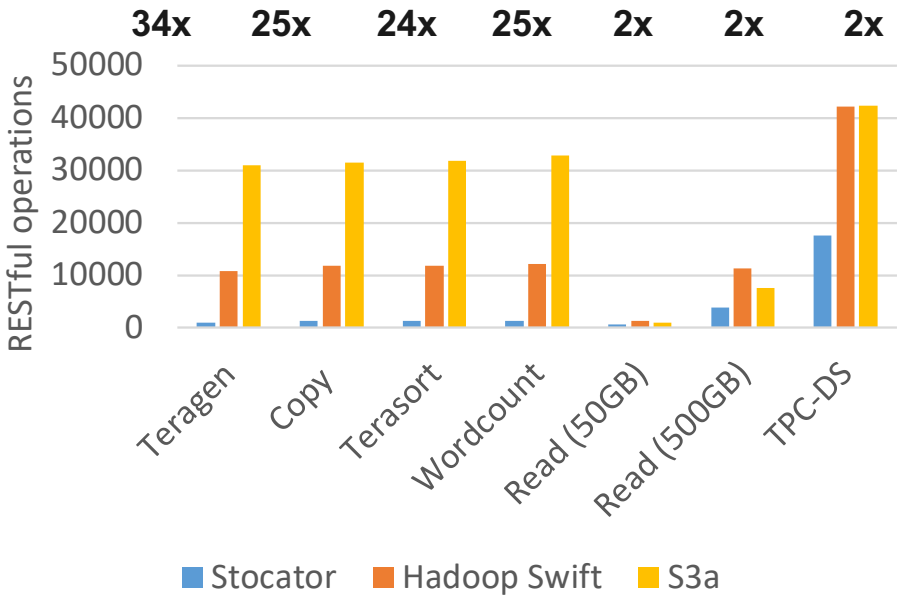
Stocator: Enabling Apache Spark for IBM Cloud Object Storage

- Historically community treated objects stores as file systems
  - Leads to inefficiencies and races
    - e.g., multiple non-atomic operations where a single operation would suffice
- Stocator is our opinionated alternative
  - Knows it is talking to an object store
    - Uses atomic PUTs and not renames
    - No dummy objects for directories
    - . . .
  - Both fast and correct
- Stocator is in open source
  - https://github.com/SparkTC/stocator



Stocator is much faster for write-intensive workloads; has equivalent performance for read workloads; and issues many fewer REST requests





As compared with the object storage connectors of Hadoop 2.7.3 run with their default parameters with Spark 2.0.1
See https://arxiv.org/abs/1709.01812

Need to

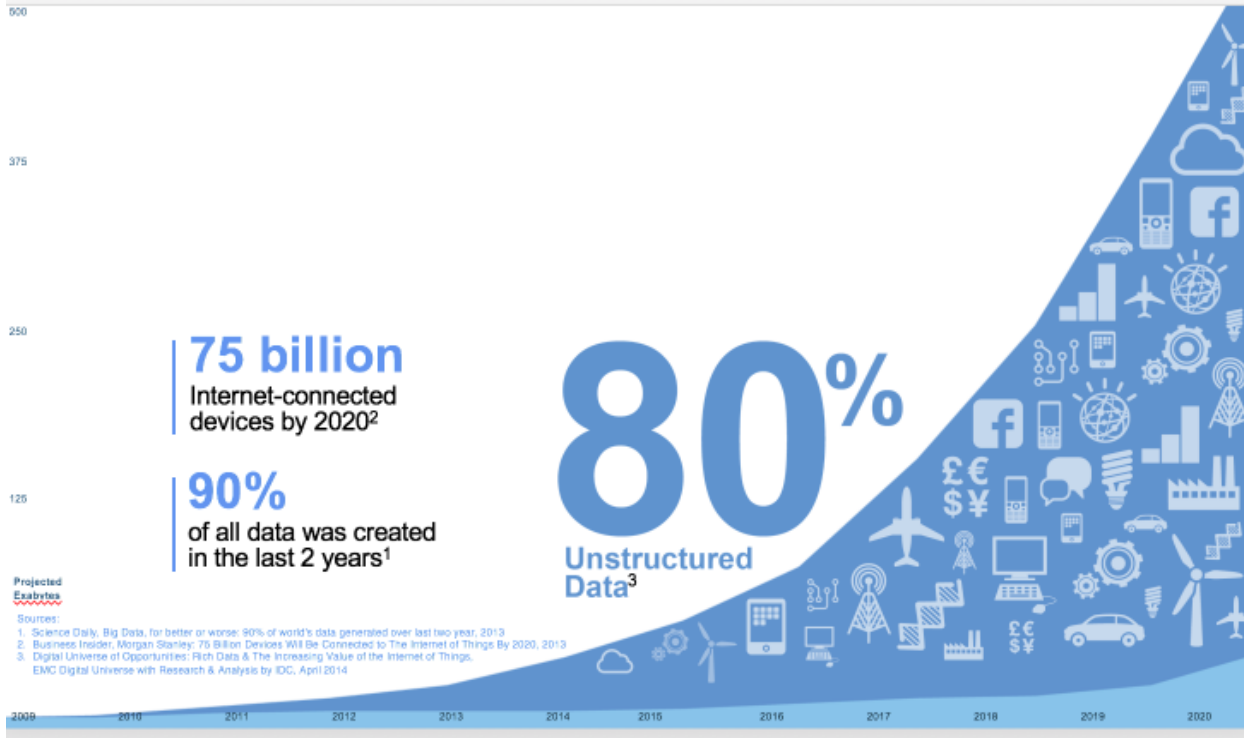
- Store
- Manage
- Protect
- Secure

while addressing

- Scale
- Cost

the data enabling developers to

- Collect
- Clean/transform
- Analyze



THANK YOU

Notices and Disclaimers

Information in these presentations (including information relating to products that have not yet been announced by IBM) has been reviewed for accuracy as of the date of initial publication and could include unintentional technical or typographical errors. IBM shall have no responsibility to update this information. THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IN NO EVENT SHALL IBM BE LIABLE FOR ANY DAMAGE ARISING FROM THE USE OF THIS INFORMATION, INCLUDING BUT NOT LIMITED TO, LOSS OF DATA, BUSINESS INTERRUPTION, LOSS OF PROFIT OR LOSS OF OPPORTUNITY. IBM products and services are warranted according to the terms and conditions of the agreements under which they are provided.

Any statements regarding IBM's future direction, intent or product plans are subject to change or withdrawal without notice.

Performance data contained herein was generally obtained in a controlled, isolated environments. Customer examples are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual performance, cost, savings or other results in other operating environments may vary.

References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business.

It is the customer's responsibility to insure its own compliance with legal requirements and to obtain advice of competent legal counsel as to the identification and interpretation of any relevant laws and regulatory requirements that may affect the customer's business and any actions the customer may need to take to comply with such laws. IBM does not provide legal advice or represent or warrant that its services or products will ensure that the customer is in compliance with any law

Notices and Disclaimers Con't.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products. IBM does not warrant the quality of any third-party products, or the ability of any such third-party products to interoperate with IBM's products. IBM EXPRESSLY DISCLAIMS ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

The provision of the information contained h erein is not intended to, and does not, grant any right or license under any IBM patents, copyrights, trademarks or other intellectual property right.

IBM, the IBM logo, ibm.com, Aspera®, Bluemix, Blueworks Live, CICS, Clearcase, Cognos®, DOORS®, Emptoris®, Enterprise Document Management System™, FASP®, FileNet®, Global Business Services ®, Global Technology Services ®, IBM ExperienceOne™, IBM SmartCloud®, IBM Social Business®, Information on Demand, ILOG, Maximo®, MQIntegrator®, MQSeries®, Netcool®, OMEGAMON, OpenPower, PureAnalytics™, PureApplication®, pureCluster™, PureCoverage®, PureData®, PureExperience®, PureFlex®, pureQuery®, pureScale®, PureSystems®, QRadar®, Rational®, Rhapsody®, Smarter Commerce®, SoDA, SPSS, Sterling Commerce®, StoredIQ, Tealeaf®, Tivoli®, Trusteer®, Unica®, urban{code}®, Watson, WebSphere®, Worklight®, X-Force® and System z® Z/OS, are trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at: www.ibm.com/legal/copytrade.shtml.

Elasticsearch is a trademark of Elasticsearch BV, registered in the U.S. and in other countries
Docker and the Docker logo are trademarks or registered trademarks of Docker, Inc. in the United States and/or other countries. Docker, Inc. and other parties may also have trademark rights in other terms used herein.
Apache, Apache Spark, Spark, Apache CouchDB, CouchDB, Apache Hadoop, Hadoop, Apache Parquet, Parquet, Apache Flume, Flume, Apache Mesos, Mesos, Apache Kafka and Kafka are trademarks of the Apache Software Foundation
OpenStack and the OpenStack Logo are trademarks of the OpenStack Foundation

Some of this work has received funding from the European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreement no 609043