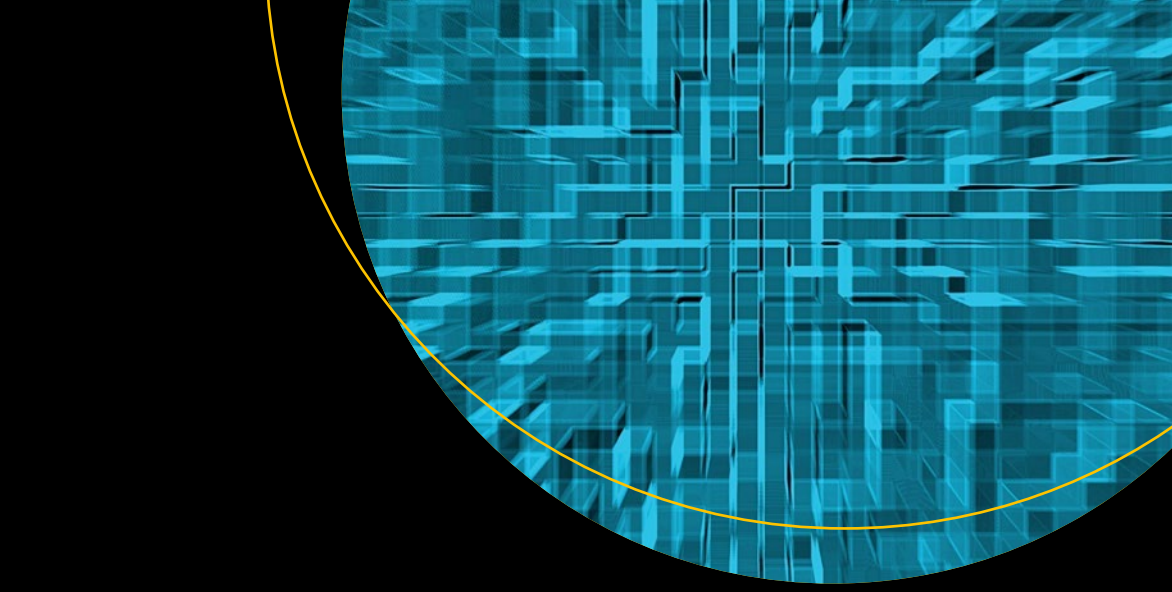


Practical Enterprise Data Lake Insights

Handle Data-Driven Challenges in an Enterprise Big Data Lake

Saurabh Gupta
Venkata Giri

Apress®

Practical Enterprise Data Lake Insights

Handle Data-Driven Challenges in an Enterprise Big Data Lake

Saurabh Gupta
Venkata Giri

Apress®

Practical Enterprise Data Lake Insights

Saurabh Gupta
Bangalore, Karnataka, India

Venkata Giri
Bangalore, Karnataka, India

ISBN-13 (pbk): 978-1-4842-3521-8
ISBN-13 (electronic): 978-1-4842-3522-5
https://doi.org/10.1007/978-1-4842-3522-5

Library of Congress Control Number: 2018948701

Managing Director, Apress Media LLC: Welmoed Spahr
Acquisitions Editor: Nikhil Karkal
Development Editor: Laura Berendson
Coordinating Editor: Divya Modi

Cover designed by eStudioCalamar

Cover image designed by Freepik (www.freepik.com)

Distributed to the book trade worldwide by Springer Science+Business Media New York, 233 Spring Street, 6th Floor, New York, NY 10013. Phone 1-800-SPRINGER, fax (201) 348-4505, e-mail orders-ny@springer-sbm.com, or visit www.springeronline.com. Apress Media, LLC is a California LLC and the sole member (owner) is Springer Science + Business Media Finance Inc (SSBM Finance Inc). SSBM Finance Inc is a **Delaware** corporation.

For information on translations, please e-mail rights@apress.com, or visit http://www.apress.com/rights-permissions.

Apress titles may be purchased in bulk for academic, corporate, or promotional use. eBook versions and licenses are also available for most titles. For more information, reference our Print and eBook Bulk Sales web page at http://www.apress.com/bulk-sales.

Any source code or other supplementary material referenced by the author in this book is available to readers on GitHub via the book's product page, located at www.apress.com/978-1-4842-3521-8. For more detailed information, please visit http://www.apress.com/source-code.

Printed on acid-free paper

Table of Contents

About the Authors xi

About the Technical Reviewer xiii

Acknowledgments xv

Foreword xvii

Chapter 1: Introduction to Enterprise Data Lakes 1

Data explosion: the beginning 3

Big data ecosystem 6

Hadoop and MapReduce – Early days 7

Evolution of Hadoop 8

History of Data Lake 11

Data Lake: the concept 12

Data lake architecture 13

Why Data Lake? 15

Data Lake Characteristics 16

Data lake vs. Data warehouse 19

How to achieve success with Data Lake? 21

Data governance and data operations 22

Data democratization with data lake 25

Fast Data - Life beyond Big Data 28

Conclusion 30

Chapter 2: Data lake ingestion strategies 33

What is data ingestion? 34

Understand the data sources 35

Structured vs. Semi-structured vs. Unstructured data.....37

Data ingestion framework parameters.....39

ETL vs. ELT.....45

Big Data Integration with Data Lake 47

Hadoop Distributed File System (HDFS) 48

Copy files directly into HDFS 49

Batched data ingestion.....49

Challenges and design considerations 51

Design considerations.....52

Commercial ETL tools.....57

Real-time ingestion 58

CDC design considerations.....60

Example of CDC pipeline: Databus, LinkedIn's open-source solution.....61

Apache Sqoop.....64

Sqoop 1 64

Sqoop 2 65

How Sqoop works?.....66

Sqoop design considerations 67

Native ingestion utilities 71

Oracle copyToBDA 72

Greenplum gphdfs utility 73

Data transfer from Greenplum to using gpfdist.....76

Ingest unstructured data into Hadoop....77
Apache Flume....77
Tiered architecture for convergent flow of events....79
Features and design considerations....80
Conclusion....85

Chapter 3: Capture Streaming Data with Change-Data-Capture....87

Change Data Capture Concepts....88
Strategies for Data Capture....89
Retention and Replay....91
Retention Period....92
Types of CDC....93
Incremental....94
Bulk....94
Hybrid....95
CDC – Trade-offs....95
CDC Tools....97
Challenges....98
Downstream Propagation....98
Use Case....99
Centralization of Change Data....100
Analyzing a Centralized Data Store....101
Metadata: Data about Data....102
Structure of Data....104
Privacy/Sensitivity Information....104
Special Fields....104
Data Formats....105

Delimited Format 105
Avro File Format 106
Consumption and Checkpointing 107
Simple Checkpoint Mechanism 107
Parallelism 107
Merging and Consolidation 108
Design Considerations for Merge and Consolidate 109
Data Quality 110
Challenges 111
Design Aspects 112
Operational Aspects 112
Publishing to Kafka 115
Schema and Data 117
Sample Schema 118
Schema Repository 119
Multiple Topics and Partitioning 120
Sizing and Scaling 121
Tools 122
Conclusion 123

Chapter 4: Data Processing Strategies in Data Lakes 125

MapReduce Processing Framework 126
Motivation: Why MapReduce? 127
MapReduce V1 Refresher and Design Considerations 128
Yet Another Resource Negotiator – YARN 136
Hive 141
Hive – Quick Refresher 143
Hive Metastore (a.k.a. HCatalog) 146

Hive – Design Considerations....148
Hive LLAP....158
Apache Pig....160
Pig Execution Architecture....161
Apache Spark....166
Why Spark?....167
Resilient Distributed Datasets (RDD)....169
RDD Runtime Components....171
RDD Composition....174
Datasets and DataFrames....175
Deployment Modes of Spark Application....178
Design Considerations....180
Caching and Persistence of an RDD in Spark....182
RDD Shared Variables....183
SQL on Hadoop....184
Presto....186
Oracle Big Data SQL....194
Design Considerations....197
Conclusion....199

Chapter 5: Data Archiving Strategies in Data Lakes....201

The Act of Data Governance....202
Data lake vs. Data swamp....204
Introduction to Data Archival....205
Data Lifecycle Management (DLM)....208
DLM policy actions....210
DLM strategies....211
DLM design considerations....213

Amazon S3 and Glacier storage classes 217
Design considerations 219
DLM Case Study – Archiving with Amazon 220
Conclusion 222

Chapter 6: Data Security in Data Lakes 225

System Architecture 226
Network Security 227
Hadoop Roles within a cluster 230
Host Firewalls for operating system security 232
Data in Motion 233
Communication Problem 233
Data at Rest 237
Procedure to generate and verify key in LUKS 238
Access flow for the user 238
Performance using LUKS 243
Multiple passphrases with LUKS 243
Kerberos 244
Kerberos Protocol overview 244
Kerberos components 246
Kerberos flow 247
Kerberos commands 249
HDFS ACL 256
HDFS Authorization with Apache Ranger 257
What Ranger does? 258
Conclusion 259

Chapter 7: Ensure High Availability of Data Lake 261

Scale Hadoop through HDFS federation.....262

High availability of Hadoop components.....267

Hive metastore 267

HiveServer2 and Zookeeper integration 268

Setup HA for Kerberos 269

NameNode high availability.....272

Architecture.....273

Data Center disaster recovery strategies 280

Data replication strategies.....287

Active-passive data center replication 289

Active-active data center replication.....290

Conclusion 295

Chapter 8: Managing Data Lake Operations 297

Monitoring Architecture 299

Hadoop metrics architecture 300

Identification of source components.....301

YARN metrics.....301

MapReduce metrics.....302

HDFS.....302

Metric collection tools 303

Metrics and log storage.....305

Logs and Metrics visualization.....307

Kibana 308

Apache Ambari 309
Data lake operationalization 311
Conclusion 315

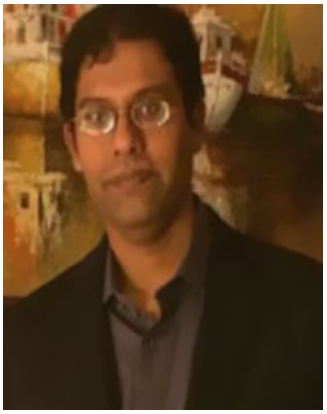
Index 317

About the Authors

Saurabh Gupta is a technology leader, published author, and database enthusiast with more than 11 years of industry experience in data architecture, engineering, development, and administration. Working as a Manager, Data & Analytics at GE Transportation, his focus lies with data lake analytics programs that build digital solutions for business stakeholders. In the past, he has worked extensively with Oracle database design and development, PaaS and IaaS cloud service models, consolidation, and in-memory technologies. He has authored two books on advanced PL/SQL for Oracle versions 11g and 12c. He is a frequent speaker at numerous conferences organized by the user community and technical institutions. He tweets at @saurabhkg and blogs at sbhoracle.wordpress.com.

Venkata Giri currently works with GE Digital and has been involved with building resilient distributed services on a massive scale. He has worked on Bigdata tech stack, relational databases, high availability, and performance tuning. With over 20 years of experience in data technologies, he has in-depth knowledge of big data ecosystems, complex data ingestion pipelines, data engineering, data processing, and operations. Prior to GE, he worked with the data teams at LinkedIn and Yahoo.

About the Technical Reviewer



As Director in LinkedIn's site reliability engineering organization, **Sai Selvaganesan** brings close to two decades of experience in data, from design, engineering, and operations to site reliability. With experience across multiple Silicon Valley companies including Apple, Yahoo, and LinkedIn, Sai's focus areas have been around scaling and optimizing data infrastructure and he holds multiple patents in the space.

Sai spearheaded strategic projects that helped forge multi-colo operations at LinkedIn. Previously, he worked on key initiatives including Yahoo's Panama project to overhaul search. Sai has a proven track record of building high-impact global teams focused on execution excellence and fuelling growth.

Sai holds a BA in Electrical Engineering from NIIT in India and is currently pursuing his MBA from UCLA.

Acknowledgments

We would like to thank Apress for giving us the opportunity to work on this project. A big shout goes out to the entire editorial team who have been extremely supportive throughout. Thanks Nikhil, Divya, and Laura. Trust me, it was not an episode, rather a journey.

Thanks Sai for accepting our request to review our content. It was indeed a great learning experience for us to have feedback from someone so humble and a master of the subject. We acknowledge your efforts in questioning us and ensuring quality of the product. We would like to graciously thank Janardh Bantupalli and Aditya for their distinguished contribution on change data capture and data operation topics.

Needless to say, all this would have never been possible without organizational support. Special thanks to GE legal for allowing us to pursue our interest. We would like to express our gratitude to Data & Analytics staff for their faith and encouragement. Thank you, Rick, Vijay, Libby, Jayadeep, Mayukh, and Diwakar.

Thanks to my family for bearing me all this time. It's not easy but whatever I am, is all because of your love and support. You are the life in me!

Foreword

When I was 10 years old, I would spend hours in the local library poring over books and recording pages and pages of notes, trying to soak up all the information I could. I was steadily building my knowledge bank so I would be ready with all the answers, whether I was applying that knowledge to write a book report or impress my parents with my rapid recall of statistics and facts about the world. I fast forward to today when my 8-year-old son calls out questions to the device on my kitchen counter and immediately gets answers, without having to access any websites, dig through books, or even leave his own house looking for that exact fact. In essence, learning from data that may be housed in a data lake instead of a structured data warehouse or in a book. The world has changed. We have volumes of data generated simply because of our ability to capture it – we are no longer limited to transactional systems or data captured only by written form. While the amount of data available is exponentially increasing, however, truly capitalizing on its value is dependent on having access when and how we need it. As technology leaders, we have the responsibility to make this data accessible so that it can be transformed into even more valuable information.

As a popularly covered topic in tech and management publications, some may ask, haven't we solved for that? Well, we've had a good start, but I would argue that new challenges have emerged. Information is not structured in the way it used to be instead it is being captured as both structured and unstructured data sets. As we lead our organizations forward, we must empower users through data democratization – putting the data in the hands of the end users so they can transform it into information in a relevant and meaningful way. The concept is powerful,

and many organizations are embracing it, but the challenge of how to do it effectively remains a barrier. What are the stages of capturing the unstructured data, processing it and then allowing access to query it. On top of that, how do you manage the access and levels of security. These are challenging new questions that technology leaders face today.

The good news is that the challenges are not insurmountable. Importantly, though, is that, as the volume of data increases, the need to manage data processing with speed becomes paramount. Enterprise users have expectations of "consumer-like" experiences where speed and ease-of-use are key. What we need now is a practical approach to address this reality. From my experience, it starts with a cohesive enterprise data lake strategy. The data lake strategy needs to be architected with end user in mind and the opportunity to enable a variety of problem statements to be tackled. Unlike traditional transactional reporting where a problem statement is articulated at the beginning of the journey, the data lake attempts to fundamentally approach this in the inverse. Data is no longer a byproduct. Instead it is waiting for the user to apply a context and connect and discover data to convert it into information that can drive outcomes. The age of a data-driven culture has arrived and the principles and architecture of an enterprise data lake need to be ready to handle to volume, complexity, and flexibility.

Monica Caldas
CIO & SVP, GE Transportation
"Digital Leader of the Year" 2018
(http://womeninitawards.com/new-york/2018-usa-winners/)

CHAPTER 1

Introduction to Enterprise Data Lakes

"In God, we Trust; all others must bring data"

—*W. Edwards Deming*, a statistician who devised "Plan-Do-Study-Act" method

It was in 1861 when Charles Joseph Minard, an 80-year-old French civil engineer, attempted to develop a visual that can narrate Napoleon's disastrous Russian campaign of 1812. Figure 1-1 depicts people movement and exhibits details on geography, time, temperature, troop count, course, and direction.

S. Gupta and V. Giri, *Practical Enterprise Data Lake Insights*,
https://doi.org/10.1007/978-1-4842-3522-5_1

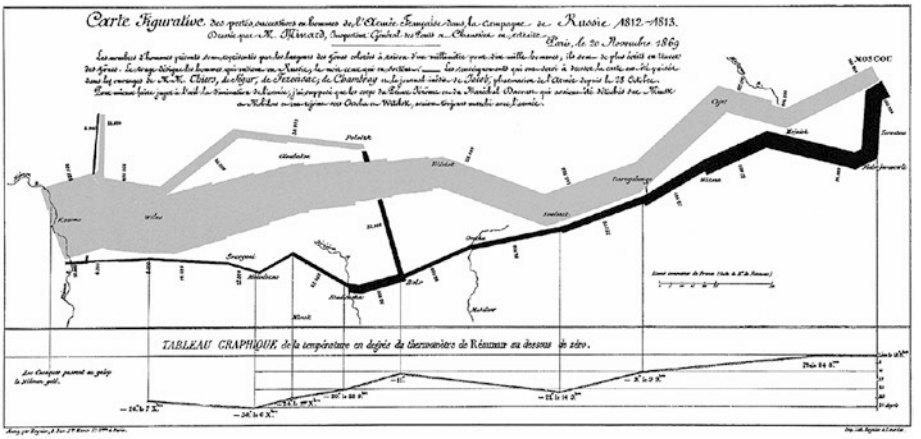


Figure 1-1. *Minard's map of Napolean's russing campaign in 1812 Source: "Worth a thousand words: A good graphic can tell a story, bring a lump to the throat, even change policies. Here are three of history's best." The Economist, December 19, 2007,* https://www.economist.com/node/10278643.

From the above chart, in 1812, the Grand Army consisted of 422,000 personnel started from Poland; out of which only 100,000 reached Moscow and 10,000 returned. The French community describes the tragedy as "C'est la Bérézina".

The chart depicts the tragic tale with such clarity and precision. The quality of the graph is accredited to the data analysis from Minard and a variety of factors soaked in to produce high-quality map. It remains one of the best examples of statistical visualization and data storytelling to date. Many analysts have spent ample time to analyze through Minard's map and prognosticated the steps he must have gone through before painting a single image, though painful, of the entire tragedy.

Data analysis is not new in the information industry. What has grown over the years is the data and the expectation and demand to churn "gold" out of data. It would be an understatement to say that data has brought nothing but a state of confusion in the industry. At times, data gets

unreasonable hype, though justified, by drawing an analogy with currency, oil, and everything precious on this planet.

Approximately two decades ago, data was a vaporous component of the information industry. All data used to exist raw, and was consumed raw, while its crude format remained unanalyzed. Back then, the dynamics of data extraction and storage were dignified areas that always posed challenges for enterprises. It all started with business-driven thoughts like variety, availability, scalability, and performance of data when companies started loving data. They were mindful of the fact that at some point, they need to come out of relational world and face the real challenge of data management. This was one of the biggest information revolutions that web 2.0 companies came across.

The information industry loves new trends provided they focus on business outcomes, catchy and exciting in learning terms, and largely uncovered. Big data picked up such a trend that organizations seemed to be in a rush to throw themselves under the bus, but failed miserably to formulate the strategy to handle data volume or variety that could potentially contribute in meaningful terms. The industry had a term for something that contained data: data warehouse, marts, reservoirs, or lakes. This created a lot of confusion but many prudent organizations were ready to take bets on data analytics.

Data explosion: the beginning

Data explosion was something that companies used to hear but never questioned their ability to handle it. Data was merely used to maintain a system of record of an event. However, multiple studies discussed the potential of data in decision making and business development. Quotes like “Data is the new currency” and “Data is the new oil of Digital Economy” struck headlines and urged many companies to classify data as a corporate asset.

Research provided tremendous value hidden in data that can give deep insight in decision making and business development. Almost every action within a "digital" ecosystem is data-related, that is, it either consumes or generates data in a structured or unstructured format. This data needs to be analyzed promptly to distill nuggets of information that can help enterprises grow.

So, what is Big Data? Is it bigger than expected? Well, the best way to define Big Data is to understand what traditional data is. When you are fully aware of data size, format, rate at which it is generated, and target value, datasets appear to be traditional and manageable with relational approaches. What if you are not familiar with what is coming? One doesn't know the data volume, structure, rate, and change factor. It could be structured or unstructured, in kilobytes or gigabytes, or even more. In addition, you are aware of the value that this data brings. This paradigm of data is capped as Big Data IT. Major areas that distinguish traditional datasets from big data ones are Volume, Velocity, and Variety. "Big" is rather a relative measure, so do the three "V" areas. Data volume may differ by industry and use case. In addition to the three V's, there are two more recent additions: Value and Veracity. Most of the time, the value that big data carries cannot be measured in units. Its true potential can be weighed only by the fact that it empowers business to make precise decisions and translates into positive business benefits. The best way to gauge ROI would be to compare big data investments against the business impact that it creates. Veracity refers to the accuracy of data. In the early stages of big data project lifecycle, quality, and accuracy of data matters to a certain extent but not entirely because the focus is on stability and scalability instead of quality. With the maturity of the ecosystem and solution stack, more and more analytical models consume big data and BI applications report insights, thereby instigating a fair idea about data quality. Based on this measure, data quality can be acted upon.

Let us have a quick look at the top Big Data trends in 2017 (Figure 1-2).

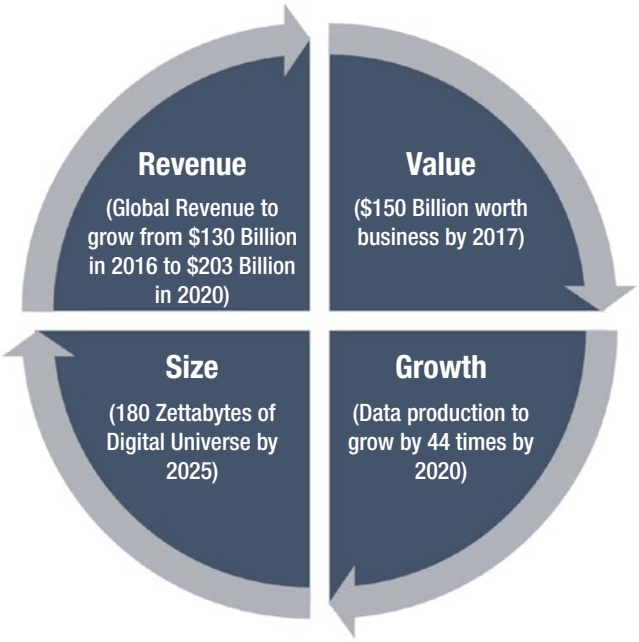


Figure 1-2. *Top big data trends in 2017. Source: Data from "Double-Digit Growth Forecast for the Worldwide Big Data and Business Analytics Market Through 2020 Led by Banking and Manufacturing Investments, According to IDC," International Data Corporation (IDC), October, 2016, https://www.idc.com/getdoc.jsp?containerId=prUS41826116.*

The top facts and predictions about Big Data in 2017 are:

1. Per IDC, worldwide revenues for big data and business analytics (BDA) will grow from $130.1 billion in 2016 to more than $203 billion in 2020.

2. Per IDC, the Digital Universe estimated is to grow to 180 Zettabytes by 2025 from pre-estimated 44 Zettabytes in 2020 and from less than 10 Zettabytes in 2015.

3. Traditional data is estimated to fold by 2.3 times between 2020 and 2025. In the same span of time, analyzable data will grow by 4.8 times and actionable data will grow by 9.6 times.
4. Data acumen continues to be a challenge. Organization alignment and a management mindset are found to be more business centric than data centric.
5. Technologies like Big Data, Internet of Things, data streaming, business intelligence, and cloud will converge to become a much more robust data management package. Cloud-based analytics to play key role in accelerating the adoption of big data analytics.
6. Deep Learning, one of Artificial Intelligence's (AI) strategies, will be a reality. It will be widely used for semantic indexing, and image and video tagging.
7. Non-relational analytical data stores will grow by 38.6% between 2015 to 2020.

Big data ecosystem

Big data IT strategy becomes critical when the nature of datasets goes beyond the capabilities of traditional (rather relational) approaches of handling data. At a high level, let us see what challenges Big Data brings to the table.

1. Data can be structured, semi-structured, or not structured at all. It is to impossible to design a generic strategy that can cater datasets of all structures.

2. Data from different sources can flow at different change rates. It may or may not have a schema.
3. How to process disparate datasets of sizes ranging from multi terabytes to multi petabytes together?
4. Common infrastructure must be cost effective and reliable, and should be fault tolerant and resilient. Total cost of ownership should be controllable to achieve high returns.

For a Big Data IT strategy to be successful, data must flow from a distinctive and reliable source system at a pre-determined frequency. Data must be relevant and mature enough to create critical insights and achieve specific business outcomes. From the cost perspective, enterprises were investing huge, in infrastructure to support storage, computing power, and parallelization.

Hadoop and MapReduce – Early days

In 2004, Google, in an effort to index the web, released white papers on data processing for large distributed data-intensive applications. The intent was to address two problem statements directly: storage and processing.

Google introduced MapReduce as the data processing framework and Google File System (GFS) as a scalable distributed file system. What makes the MapReduce framework highly scalable is the fact that a parallelized processing layer comes down all the way to the data layer that is distributed across multiple commodity machines. Google File System was designed for fault tolerance that can be accessed by multiple clients and achieve performance, scalability, availability, and reliability at the same time.

MapReduce proved to be the game changer for data process-intensive applications. It's a simple concept of breaking down a data processing task into a bunch of mappers that can run in parallel on thousands of commodity machines. Reducers constitute a second level of data processing operations that runon top of output generated from mappers.

Evolution of Hadoop

In the year 2002, the Yahoo! development team started a large-scale open-source web search project called Nutch. While Hadoop was still in the conceptual phase, Nutch's primary challenge was its inability to scale beyond a certain page limit. Then the concept of Google's GFS was introduced to project Nutch. A GFS-like file system resolved storage-related issues by allowing large files to sit in a system that was fault tolerant and available. By 2004, an open source implementation of GFS was ready as Nutch Distributed Filesystem (NDFS).

In 2004, Google introduced the MapReduce processing framework, which for obvious reasons, was immediately added into project Nutch. By early 2005, Nutch algorithms were already working with NDFS and MapReduce at an enterprise level. Such instrumentation was the combination of NDFS and MapReduce that, in 2006, Yahoo! took this package out of project Nutch. Doug Cutting was fascinated by a little stuffy yellow elephant and named this package Hadoop for the ease of memory and pronunciation. In 2008, Apache Software Foundation took over Hadoop to work beyond web-search optimization and indexing.

In a series of events starting in 2008, Hadoop stack has been pulling some magical numbers to prove its power of processing and worth at the enterprise level. In February 2008, Yahoo! claimed to generate a web search index on 10,000 core Hadoop cluster. In April 2008, Apache Hadoop set a world record as the fastest platform to process terabyte of data with a 910-node cluster. Hadoop could sort one terabyte of data in just 209 seconds, beating the previous benchmark of 297 seconds.

Hadoop 1.0 was introduced by end of the year 2011. The basic flavor of Hadoop focused on providing the storage and processing framework. The concept, MapReduce processing coupled with Hadoop Distributed Filesystem (HDFS), gained wide traction and quick adoption in the industry. Though this setup was largely appreciated due to flexibility and ease of implementation, concerns over resource management, scalability, security, and availability were still on. These drawbacks restricted the enterprise level adoption of HDFS. A high-level architecture of Hadoop 1 exhibits key components of HDFS storage layer and MapReduce processing layer. (Figure 1-3).

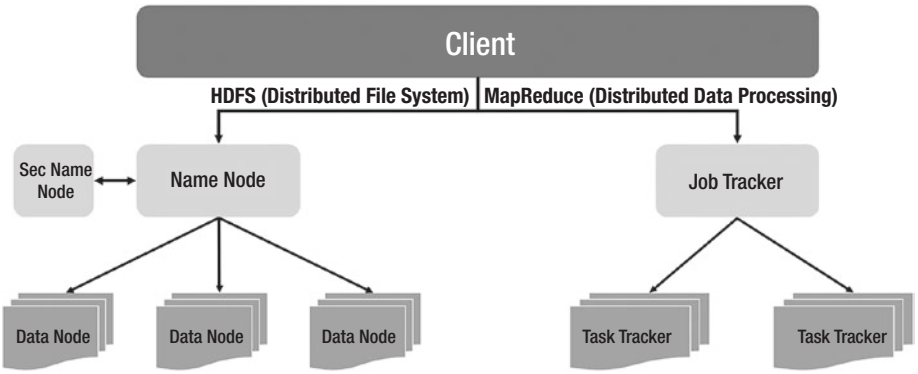


Figure 1-3. *Hadoop 1 high-level architecture*

Later in 2013, Hadoop 2.0 came out with brand new features that addressed availability and security. However, the major component in Hadoop 2.X was YARN (Yet Another Resource Negotiator). Resource management in Hadoop 1.x used to be carried out by a job tracker. Hadoop 2.x lays down another layer for resource management through YARN and segregates load management from job execution. YARN becomes responsible for resource allocation for all operations within the cluster. The MapReduce operation runs in a shell called Application Master who seeks and receives resources through YARN. It is backward compatible

with Hadoop 1.x as well. Figure 1-4 positions storage and processing components of Hadoop 2. Key callouts from the below architecture are:

- Standby NameNode to support high availability of primary NameNode
- YARN for cohesive resource management and efficient job scheduling

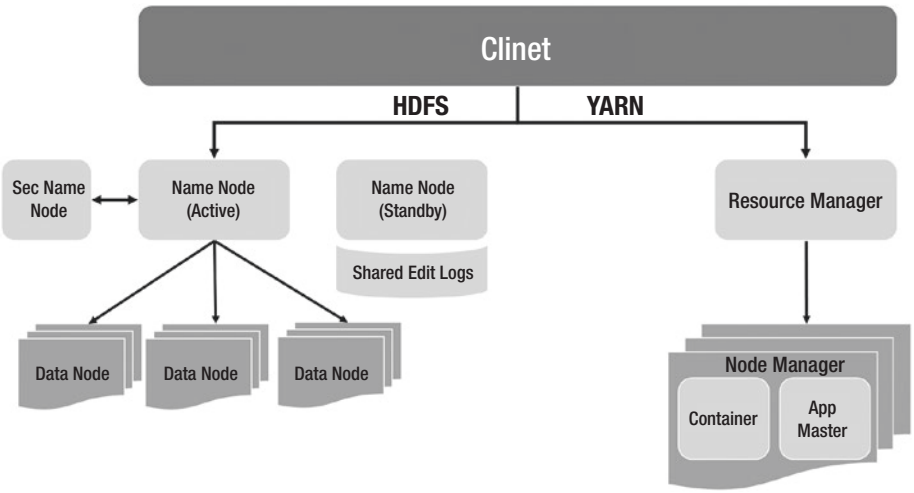


Figure 1-4. *Hadoop 2 high-level architecture*

Figure 1-5 highlights the difference between Hadoop 1.x and Hadoop 2.x at the skeleton level.

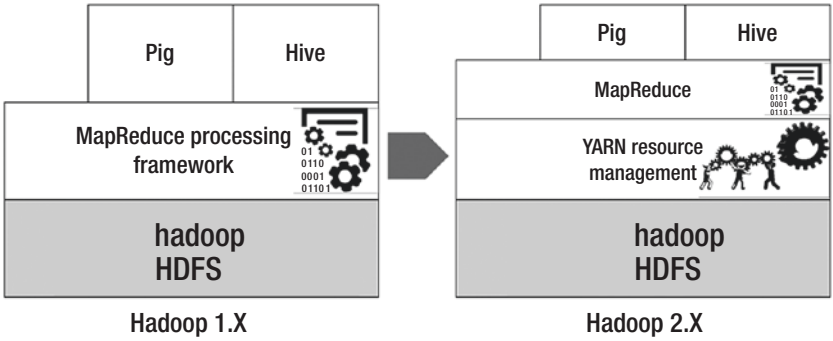


Figure 1-5. *Head to head comparison of Hadoop 1 and Hadoop 2*

History of Data Lake

Since the time Big Data trends have become buzzwords, several marketing terms have been coined to describe data management strategies. Eventually, all of them happen to represent a version of the Big Data ecosystem.

In 2010,[1] James Dixon came up with a "time machine" vision of data. Data Lake represents a state of enterprise at any given time. The idea is to store all the data in a detailed fashion in one place and empower business analytics applications, and predictive and deep learning models with one "time machine" data store. This leads to a Data-as-an-Asset strategy wherein continuous flow and integration of data enriches Data Lake to be thick, veracious, and reliable. By virtue of its design and architecture, Data Lake plays a key role in unifying data discovery, data science, and enterprise BI in an organization.

[1]Woods, Dan; "James Dixon Imagines a Data Lake that Matters," Forbes, https://www.forbes.com/sites/danwoods/2015/01/26/james-dixon-imagines-a-data-lake-that-matters/#1dd2c5e34fdb

According to James Dixon[2] *"If you think of a datamart as a store of bottled water—cleansed 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."*

The adoption rate of Data Lake reflects the progression of open-source Hadoop as a technology and its close association with the Big Data IT trend. Keep in mind that although data lake is presumed to be on Hadoop due to the latter's smooth equation with Big Data IT, it's not mandatory. Don't be surprised if you Find Data Lake hosted on relational databases. You must factor in the cost of standing a non-Hadoop stack, data size, and BI use cases.

Data Lake: the concept

Data Lake is a single window snapshot of all enterprise data in its raw format, be it structured, semi-structured, or unstructured. Starting from curating the data ingestion pipeline to the transformation layer for analytical consumption, every aspect of data gets addressed in a data lake ecosystem. It is supposed to hold enormous volumes of data of varied structures.

Data Lake is largely a product that is built using Hadoop and a processing framework. The choice of Hadoop is as direct as it can be. It not only provides scalability and resiliency, but also lowers the total cost of ownership.

At a broader level, data lake can be split into a data landing layer and an analytical layer. From the source systems, data lands directly into the data landing or mirror layer. The landing layer contains the as-is copy of the data from source systems, that is, raw data. It lays the foundation for the battleground for the analytical layer. The analytical layer is a

[2] https://jamesdixon.wordpress.com/2010/10/14/pentaho-Hadoop-and-data-lakes/

highly dynamic one in the Data Lake world as it is the downstream consumer of raw data from the mirror layer. The landing or mirror layer data is fed through a transformation layer and builds up the analytical or consumption layer. The analytical layer ensures data readiness for data analytics sandbox and thus, acts as a face-off to data scientists and analysts. Pre-built analytical models can be directly plugged in to run over the consumption layer. Perhaps, dynamic analytics like data discovery or profiling models can also be made to run directly on the consumption layer. Data visualization stacks can consume data from the consumption layer to present key indicators and data trends.

Another split of data lake can be based on temporal dimensions of data. Historical raw data can be archived and stored securely within the data lake. While it will still be active to the data lake consumers, it can be moved to a secondary storage. Mirror layer that we discussed above can hold incremental data given a pre-determined timeline. In this case, consumption layer is built upon augmented mirror layer only.

This model doesn't need to have physical data marts that are custom built to serve a singular static model. Rather they transform the data in a usable format to enable analytics and business insights. On the other hand, data warehouse provides an abstract image of a specific business wing.

Data lake architecture

Cost and IT simplification are the biggest features of Data Lake. Inexpensive Hadoop storage with schema-less-write capability and in-house processing framework using hive, pig, or python largely the success of data lake. Figure 1-6 lays out a high-level wireframe of an enterprise Data Lake.

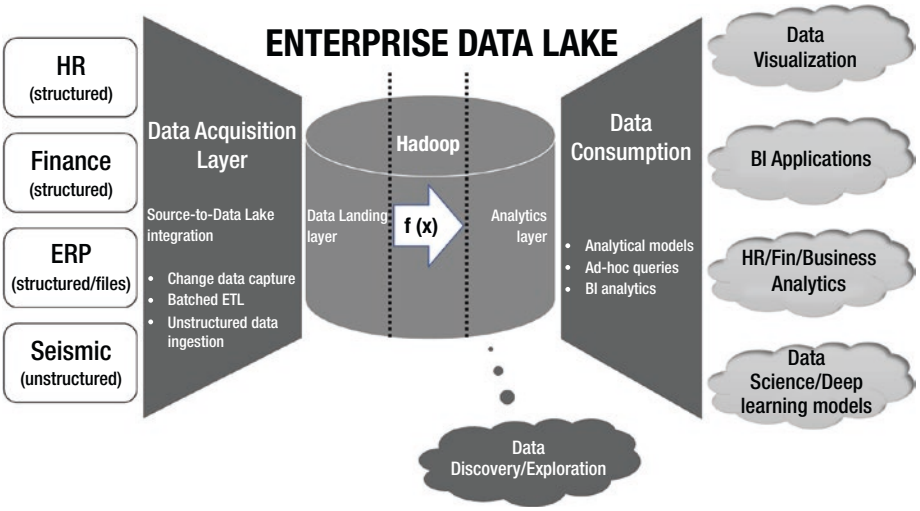


Figure 1-6. *Enterprise Data Lake Architecture*

In the above architecture diagram, there may be multiple source systems dumping data into the Enterprise Data Lake. Source systems can be of variety of nature and structure. It may come from relational sources, static file systems, web logs, or time-series sensor data from Internet of Things devices. It may or may not be structured. Without hampering the structure of data from the source and without investing into data modeling efforts in Hadoop, all source systems ingest data into Data Lake in stipulated real time.

Once data comes in the purview of Data Lake frontiers, it propagates through the processing layer to build the analytical layer. At this stage, it may be required to define the schema and structure for raw data. Thereafter, depending upon the data exchange guidelines laid down by the data governance council, data gets consumed by predictive learning models, BI applications, and data science tracks. Meanwhile, the anatomy of data discovery continues to provide a visual and exploratory face to big data in Hadoop Data Lake by directly working on raw data.