(Big) Data Engineering In Depth

From Beginner to Professional

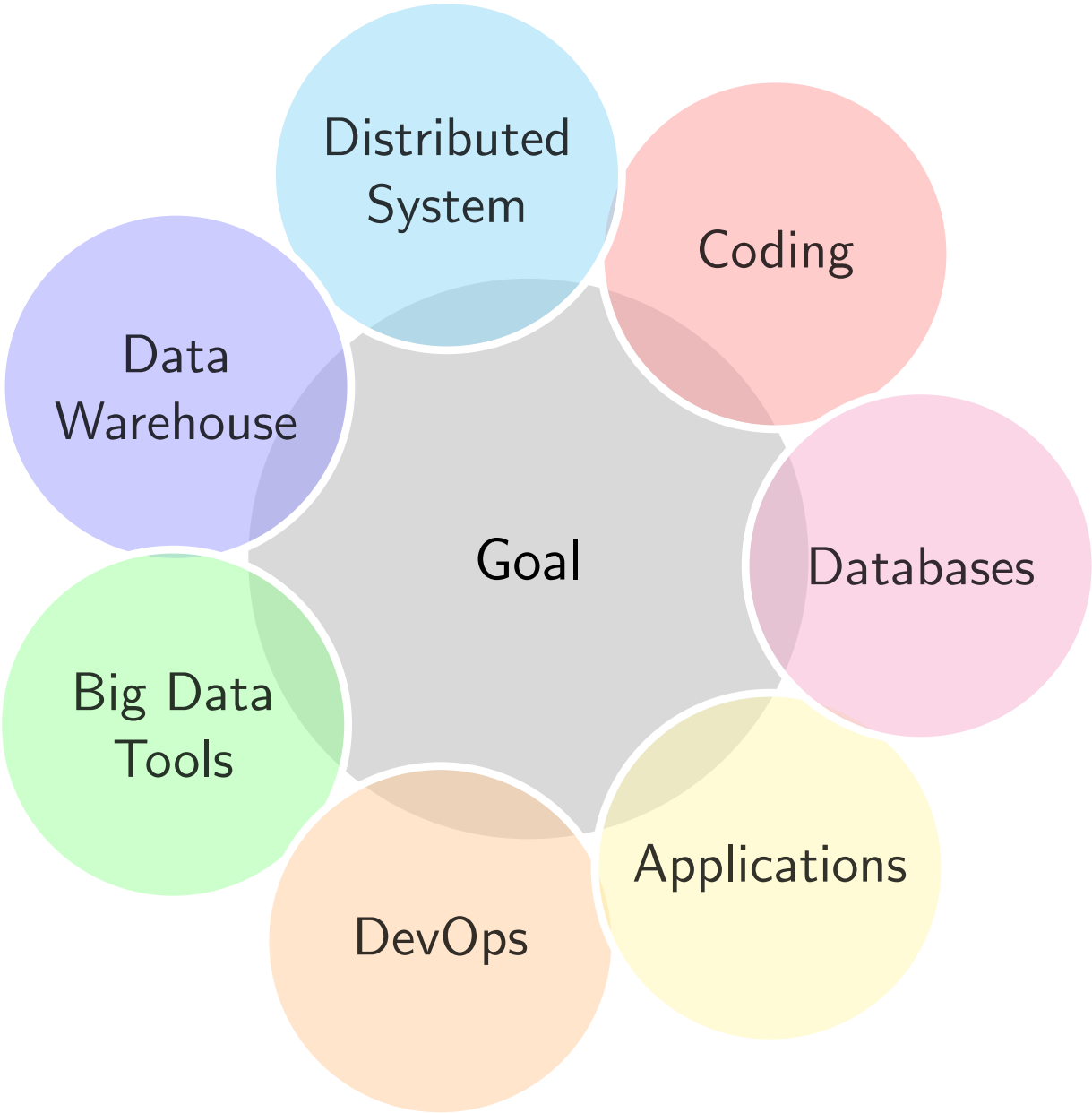
Mostafa Alaa Mohamed
Senior Big Data Engineer
MoustafaAlaa Moustafa Alaa @Moustafa_alaa22
mustafa.alaa.mohamed@gmail.com

[1]Big Data & Analytics Department, Epam Systems

The Definitive Guide to Big Data Engineering Tasks

Course Introduction

Course Target



Learning Objectives

- Understand the data management life-cycle.

Learning Objectives

- Understand the data management life-cycle.
- Illustrate the basics of distributed systems concepts

Learning Objectives

- Understand the data management life-cycle.
- Illustrate the basics of distributed systems concepts
- Be familiar with ETL for (Batch/Steaming) data over distributed systems ex: Hadoop & Spark.

Learning Objectives

- Understand the data management life-cycle.
- Illustrate the basics of distributed systems concepts
- Be familiar with ETL for (Batch/Steaming) data over distributed systems ex: Hadoop & Spark.
- Apply QA and testing for the data pipeline cycle.

Learning Objectives

- Understand the data management life-cycle.
- Illustrate the basics of distributed systems concepts
- Be familiar with ETL for (Batch/Steaming) data over distributed systems ex: Hadoop & Spark.
- Apply QA and testing for the data pipeline cycle.
- Automate the Data life-cycle process End-to-End.

Learning Objectives

- Understand the data management life-cycle.
- Illustrate the basics of distributed systems concepts
- Be familiar with ETL for (Batch/Steaming) data over distributed systems ex: Hadoop & Spark.
- Apply QA and testing for the data pipeline cycle.
- Automate the Data life-cycle process End-to-End.
- Building real-life examples.

Learning Objectives

- Understand the data management life-cycle.
- Illustrate the basics of distributed systems concepts
- Be familiar with ETL for (Batch/Steaming) data over distributed systems ex: Hadoop & Spark.
- Apply QA and testing for the data pipeline cycle.
- Automate the Data life-cycle process End-to-End.
- Building real-life examples.
- Applying machine learning over Big Data.

Learning Objectives

- Understand the data management life-cycle.
- Illustrate the basics of distributed systems concepts
- Be familiar with ETL for (Batch/Steaming) data over distributed systems ex: Hadoop & Spark.
- Apply QA and testing for the data pipeline cycle.
- Automate the Data life-cycle process End-to-End.
- Building real-life examples.
- Applying machine learning over Big Data.
- Understanding of the DevOps tools and functions in data life-cycle.

Learning Objectives

- Understand the data management life-cycle.
- Illustrate the basics of distributed systems concepts
- Be familiar with ETL for (Batch/Steaming) data over distributed systems ex: Hadoop & Spark.
- Apply QA and testing for the data pipeline cycle.
- Automate the Data life-cycle process End-to-End.
- Building real-life examples.
- Applying machine learning over Big Data.
- Understanding of the DevOps tools and functions in data life-cycle.
- Build and scale your data product.

Audience: Who Should Take This Course?

- Data Engineer who needs to get more knowledge in distributed systems and Big Data.

Audience: Who Should Take This Course?

- Data Engineer who needs to get more knowledge in distributed systems and Big Data.
- Data Warehouse Engineer who needs to know more about big data.

Audience: Who Should Take This Course?

- Data Engineer who needs to get more knowledge in distributed systems and Big Data.
- Data Warehouse Engineer who needs to know more about big data.
- Java developer who needs to change to data engineering track.

Audience: Who Should Take This Course?

- Data Engineer who needs to get more knowledge in distributed systems and Big Data.
- Data Warehouse Engineer who needs to know more about big data.
- Java developer who needs to change to data engineering track.
- DevOps engineers who needs to understand the concepts of big data.

Audience: Who Should Take This Course?

- Data Engineer who needs to get more knowledge in distributed systems and Big Data.
- Data Warehouse Engineer who needs to know more about big data.
- Java developer who needs to change to data engineering track.
- DevOps engineers who needs to understand the concepts of big data.
- Business or entrepreneur who needs to get more information about how to build or manage a data product.

Audience: Who Should Take This Course?

- Data Engineer who needs to get more knowledge in distributed systems and Big Data.
- Data Warehouse Engineer who needs to know more about big data.
- Java developer who needs to change to data engineering track.
- DevOps engineers who needs to understand the concepts of big data.
- Business or entrepreneur who needs to get more information about how to build or manage a data product.

Getting max benefit from this course

Take the course advantage

- Follow the videos order as described.

Getting max benefit from this course

Take the course advantage

- Follow the videos order as described.
- Read the references for each section (including the implementation of the examples if exists).

Getting max benefit from this course

Take the course advantage

- Follow the videos order as described.
- Read the references for each section (including the implementation of the examples if exists).
- Repeat the lecture code with your own.

Getting max benefit from this course

Take the course advantage

- Follow the videos order as described.
- Read the references for each section (including the implementation of the examples if exists).
- Repeat the lecture code with your own.
- Do the assignments.

Getting max benefit from this course

Take the course advantage

- Follow the videos order as described.
- Read the references for each section (including the implementation of the examples if exists).
- Repeat the lecture code with your own.
- Do the assignments.
- Ask your questions.

Getting max benefit from this course

Take the course advantage

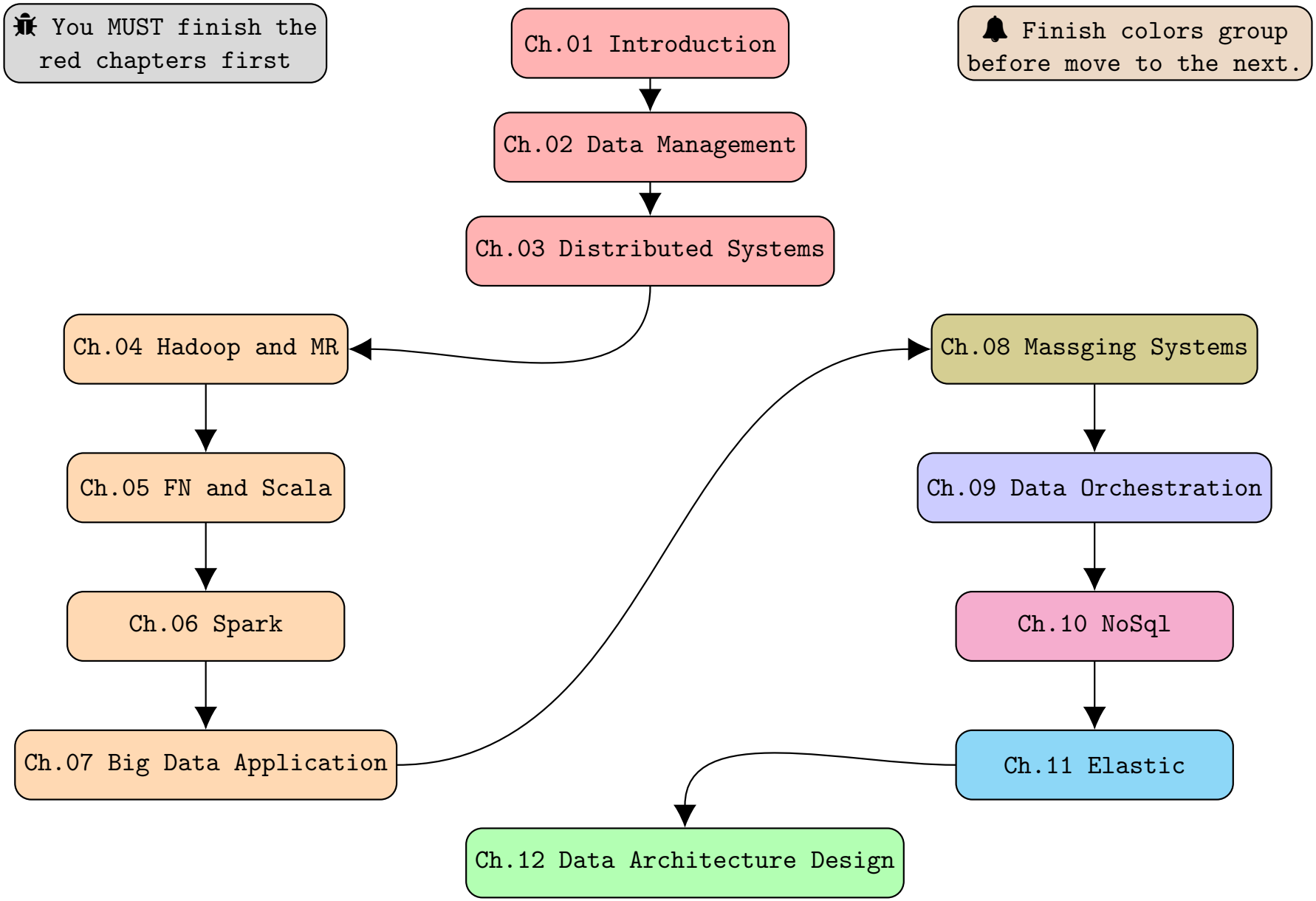
- Follow the videos order as described.
- Read the references for each section (including the implementation of the examples if exists).
- Repeat the lecture code with your own.
- Do the assignments.
- Ask your questions.
- Join the online meeting or discussions.

Getting max benefit from this course

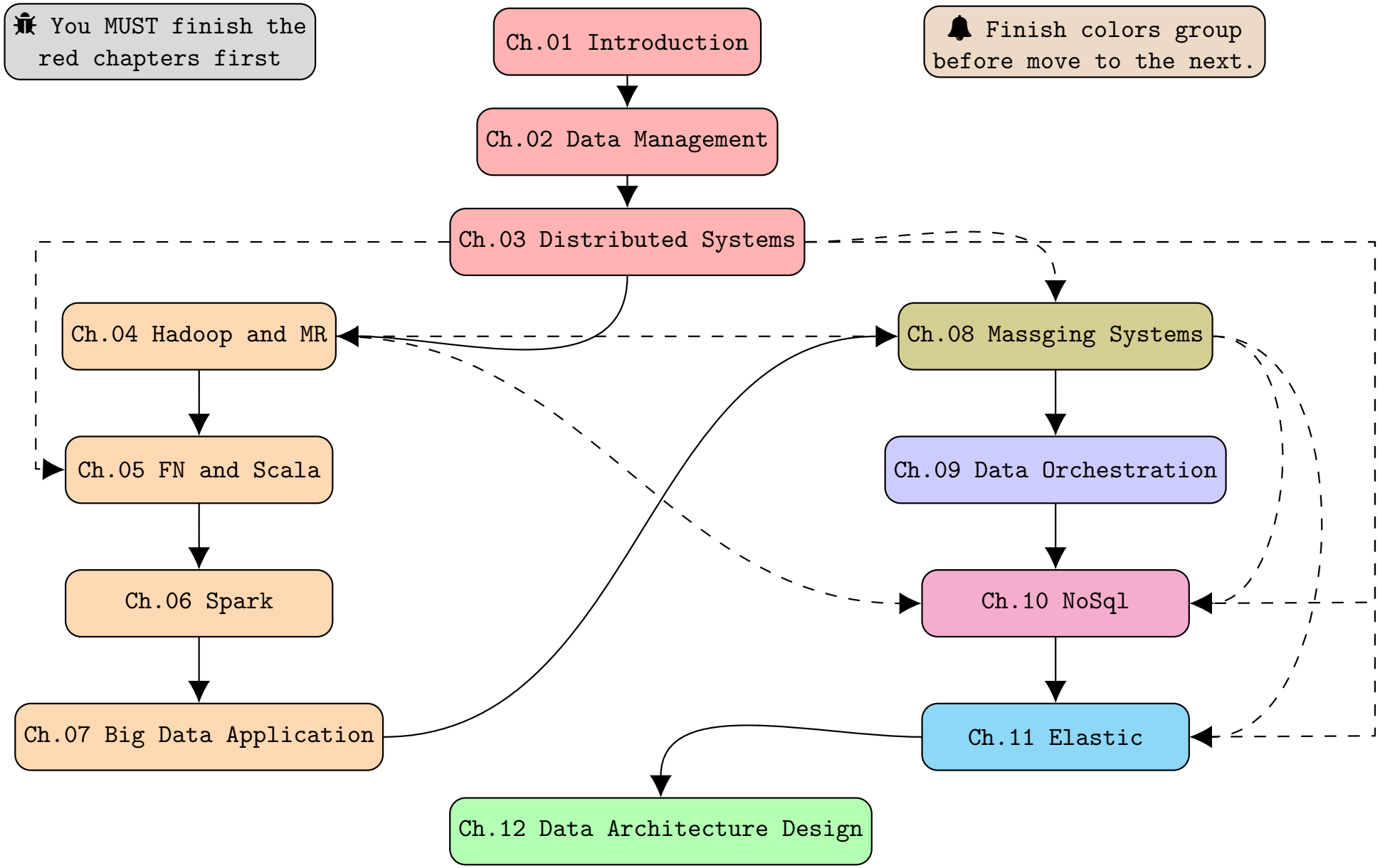
Take the course advantage

- Follow the videos order as described.
- Read the references for each section (including the implementation of the examples if exists).
- Repeat the lecture code with your own.
- Do the assignments.
- Ask your questions.
- Join the online meeting or discussions.

Chapter Dependencies



Chapter Dependencies (Jump Out Path)



Assignments and Labs

Remark

- Full project code.

Assignments and Labs

Remark

- Full project code.
- Notebooks (Jupyter or Zeppelin).

Assignments and Labs

Remark

- Full project code.
- Notebooks (Jupyter or Zeppelin).
- Read the reference.

Textbooks-1

- Hadoop: The Definitive Guide: Storage and Analysis at Internet Scale 4th Edition by Tom White.

Textbooks-1

- Hadoop: The Definitive Guide: Storage and Analysis at Internet Scale 4th Edition by Tom White.
- Learning Spark by Matei Zaharia, Patrick Wendell, Andy Konwinski, Holden Karau

Textbooks-1

- Hadoop: The Definitive Guide: Storage and Analysis at Internet Scale 4th Edition by Tom White.
- Learning Spark by Matei Zaharia, Patrick Wendell, Andy Konwinski, Holden Karau
- High Performance Spark Best Practices for Scaling and Optimizing Apache Spark By Holden Karau, Rachel Warren.

Textbooks-1

- Hadoop: The Definitive Guide: Storage and Analysis at Internet Scale 4th Edition by Tom White.
- Learning Spark by Matei Zaharia, Patrick Wendell, Andy Konwinski, Holden Karau
- High Performance Spark Best Practices for Scaling and Optimizing Apache Spark By Holden Karau, Rachel Warren.
- Kafka: The Definitive Guide by Todd Palino, Gwen Shapira, Neha Narkhede.

Textbooks-1

- Hadoop: The Definitive Guide: Storage and Analysis at Internet Scale 4th Edition by Tom White.
- Learning Spark by Matei Zaharia, Patrick Wendell, Andy Konwinski, Holden Karau
- High Performance Spark Best Practices for Scaling and Optimizing Apache Spark By Holden Karau, Rachel Warren.
- Kafka: The Definitive Guide by Todd Palino, Gwen Shapira, Neha Narkhede.
- Guide to High Performance Distributed Computing: Case Studies with Hadoop, Scalding and Spark (Computer Communications and Networks) 2015th Edition

Textbooks-1

- Hadoop: The Definitive Guide: Storage and Analysis at Internet Scale 4th Edition by Tom White.
- Learning Spark by Matei Zaharia, Patrick Wendell, Andy Konwinski, Holden Karau
- High Performance Spark Best Practices for Scaling and Optimizing Apache Spark By Holden Karau, Rachel Warren.
- Kafka: The Definitive Guide by Todd Palino, Gwen Shapira, Neha Narkhede.
- Guide to High Performance Distributed Computing: Case Studies with Hadoop, Scalding and Spark (Computer Communications and Networks) 2015th Edition
- Cassandra: The Definitive Guide: Distributed Data at Web Scale 2nd Edition.

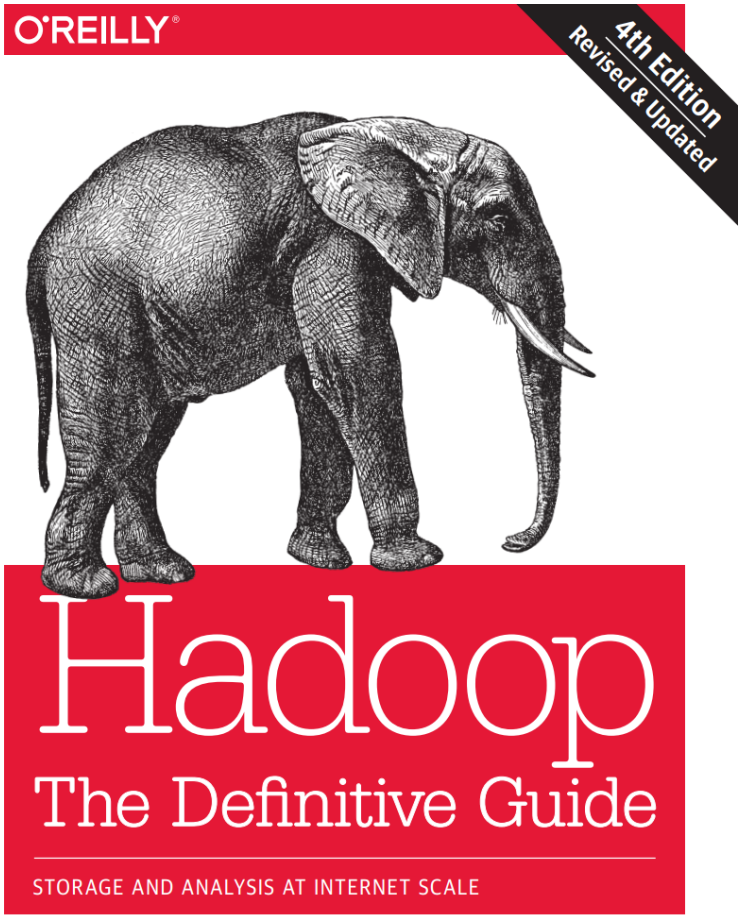
Textbooks-1

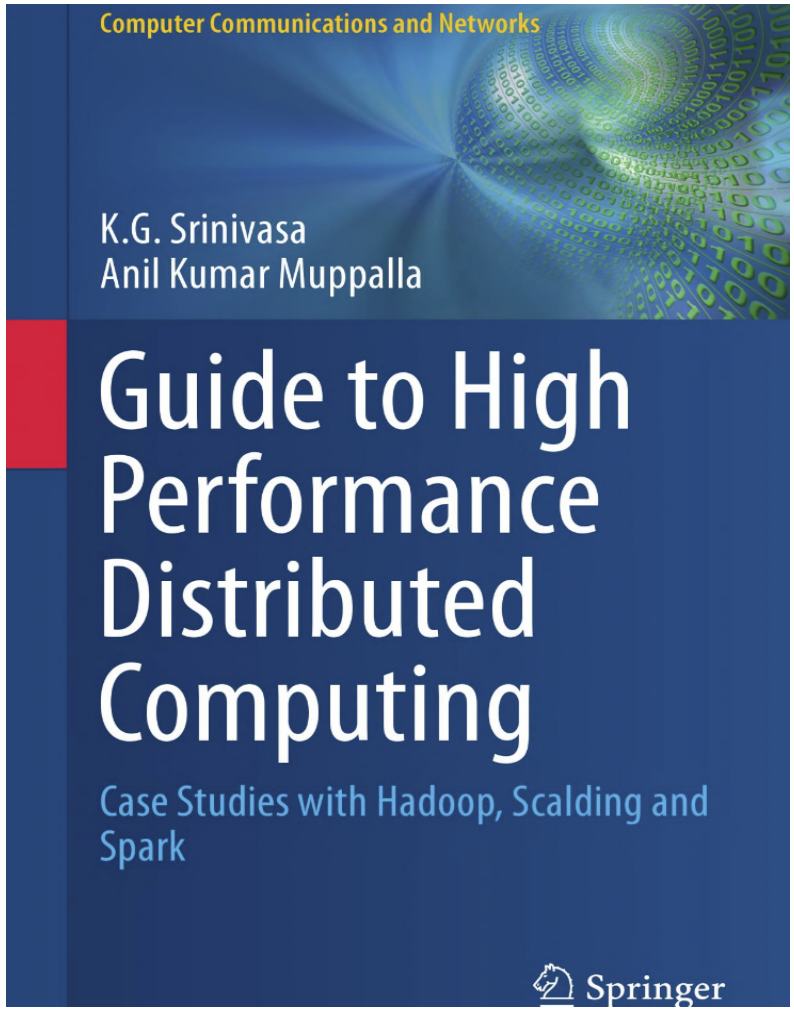
- Hadoop: The Definitive Guide: Storage and Analysis at Internet Scale 4th Edition by Tom White.
- Learning Spark by Matei Zaharia, Patrick Wendell, Andy Konwinski, Holden Karau
- High Performance Spark Best Practices for Scaling and Optimizing Apache Spark By Holden Karau, Rachel Warren.
- Kafka: The Definitive Guide by Todd Palino, Gwen Shapira, Neha Narkhede.
- Guide to High Performance Distributed Computing: Case Studies with Hadoop, Scalding and Spark (Computer Communications and Networks) 2015th Edition
- Cassandra: The Definitive Guide: Distributed Data at Web Scale 2nd Edition.
- Category Theory for Programmers Scala Edition By Bartosz Milewski, compiled and edited by Igal Tabachnik.

Textbooks-1

- Hadoop: The Definitive Guide: Storage and Analysis at Internet Scale 4th Edition by Tom White.
- Learning Spark by Matei Zaharia, Patrick Wendell, Andy Konwinski, Holden Karau
- High Performance Spark Best Practices for Scaling and Optimizing Apache Spark By Holden Karau, Rachel Warren.
- Kafka: The Definitive Guide by Todd Palino, Gwen Shapira, Neha Narkhede.
- Guide to High Performance Distributed Computing: Case Studies with Hadoop, Scalding and Spark (Computer Communications and Networks) 2015th Edition
- Cassandra: The Definitive Guide: Distributed Data at Web Scale 2nd Edition.
- Category Theory for Programmers Scala Edition By Bartosz Milewski, compiled and edited by Igal Tabachnik.
- Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems 1st Edition by Martin Kleppmann

Textbooks-2

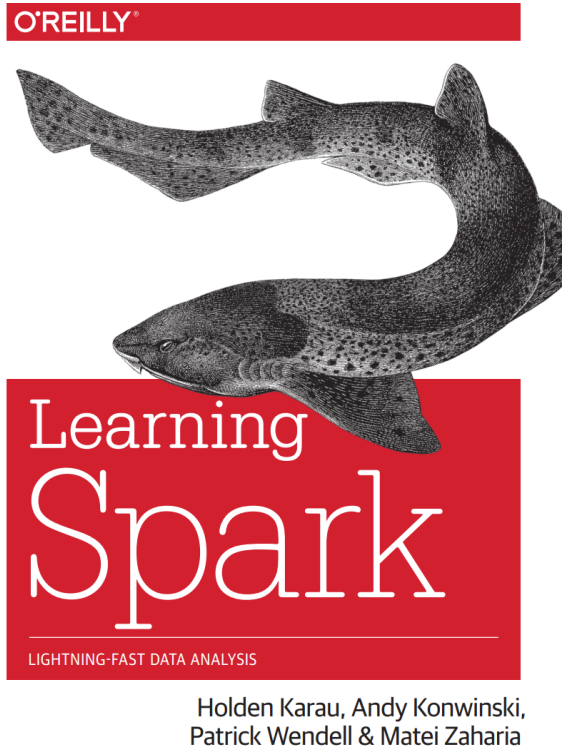

O'REILLY
4th Edition
Revised & Updated
Hadoop
The Definitive Guide
STORAGE AND ANALYSIS AT INTERNET SCALE
Tom White


Computer Communications and Networks
K.G. Srinivasa
Anil Kumar Muppalla
Guide to High Performance Distributed Computing
Case Studies with Hadoop, Scalding and Spark
Springer

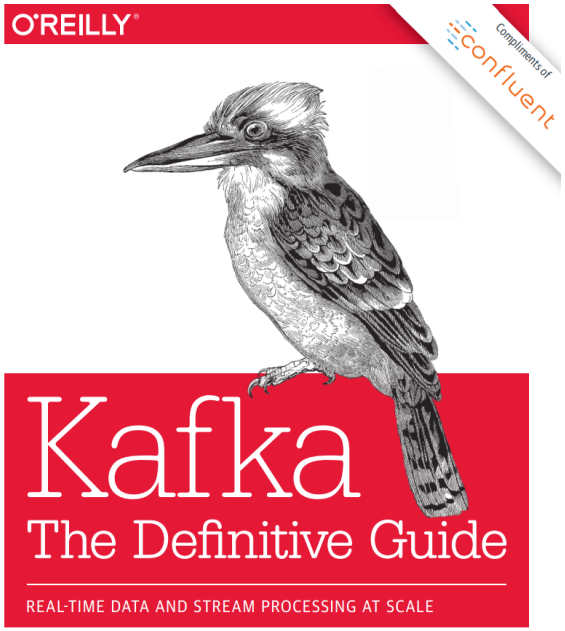
Textbooks-3

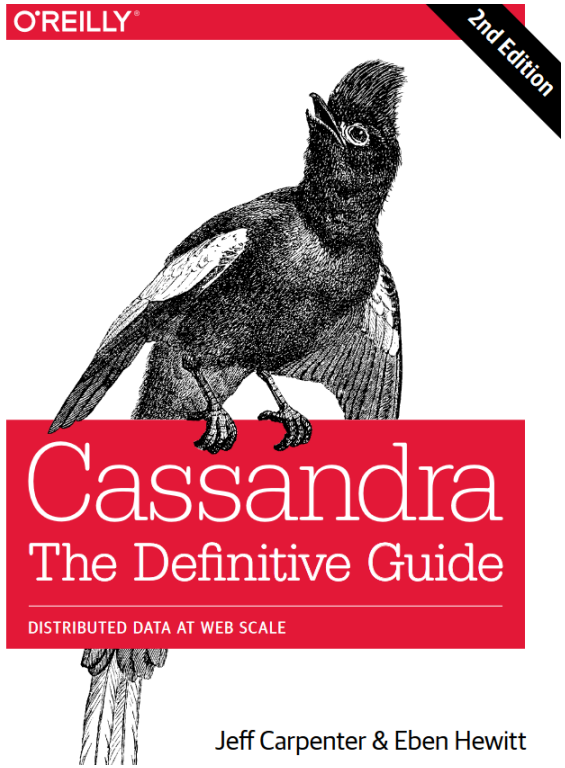

CATEGORY THEORY
FOR PROGRAMMERS
Bartosz Milewski

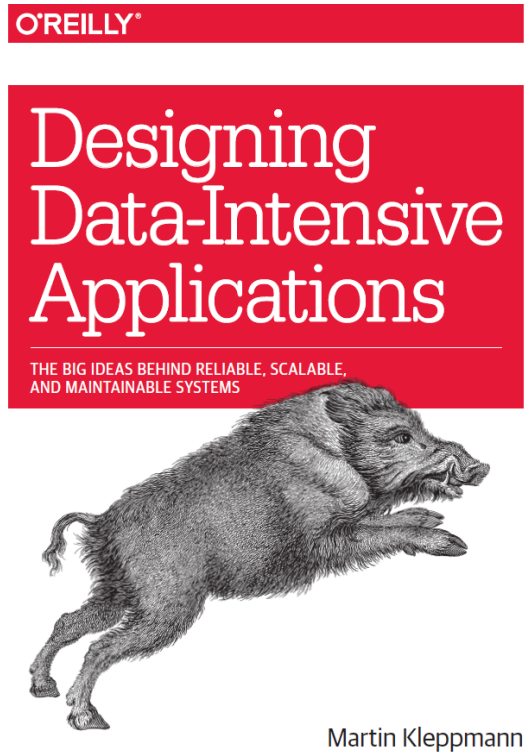

O'REILLY
High Performance
Spark
BEST PRACTICES FOR SCALING
& OPTIMIZING APACHE SPARK
Holden Karau &
Rachel Warren


O'REILLY
Learning
Spark
LIGHTNING-FAST DATA ANALYSIS
Holden Karau, Andy Konwinski,
Patrick Wendell & Matei Zaharia

Textbooks-4


O'REILLY
Compliments of
confluent
Kafka
The Definitive Guide
REAL-TIME DATA AND STREAM PROCESSING AT SCALE
Neha Narkhede,
Gwen Shapira & Todd Palino


O'REILLY
2nd Edition
Cassandra
The Definitive Guide
DISTRIBUTED DATA AT WEB SCALE
Jeff Carpenter & Eben Hewitt


O'REILLY
Designing
Data-Intensive
Applications
THE BIG IDEAS BEHIND RELIABLE, SCALABLE,
AND MAINTAINABLE SYSTEMS
Martin Kleppmann

Ugly but important

- User stories or technical discussions are not related to any of my current work or my previous companies.
- I am working at EPAM Systems. My company approved me for doing this online course public but the materials are not reviewed or assessed by my company. It is on my own responsibilities.

Table of Contents I

Table of Contents II

Table of Contents III

Table of Contents IV

Table of Contents V

Table of Contents VI

Table of Contents VII

Introduction To Data Management and Data Warehouse

Chapter Objectives

- Be familiar with data management life-cycle.

Chapter Objectives

- Be familiar with data management life-cycle.
- Introduction to data warehouse and its usage.

Chapter Objectives

- Be familiar with data management life-cycle.
- Introduction to data warehouse and its usage.
- Motivation to DWH.

Chapter Objectives

- Be familiar with data management life-cycle.
- Introduction to data warehouse and its usage.
- Motivation to DWH.
- What is the different types of DWH?

Chapter Objectives

- Be familiar with data management life-cycle.
- Introduction to data warehouse and its usage.
- Motivation to DWH.
- What is the different types of DWH?
- Use cases

Chapter Objectives

- Be familiar with data management life-cycle.
- Introduction to data warehouse and its usage.
- Motivation to DWH.
- What is the different types of DWH?
- Use cases
- Data Encoding and Formats

Chapter Objectives

- Be familiar with data management life-cycle.
- Introduction to data warehouse and its usage.
- Motivation to DWH.
- What is the different types of DWH?
- Use cases
- Data Encoding and Formats
- Challenges to build a DWH.

Chapter Objectives

- Be familiar with data management life-cycle.
- Introduction to data warehouse and its usage.
- Motivation to DWH.
- What is the different types of DWH?
- Use cases
- Data Encoding and Formats
- Challenges to build a DWH.
- Data modeling design.

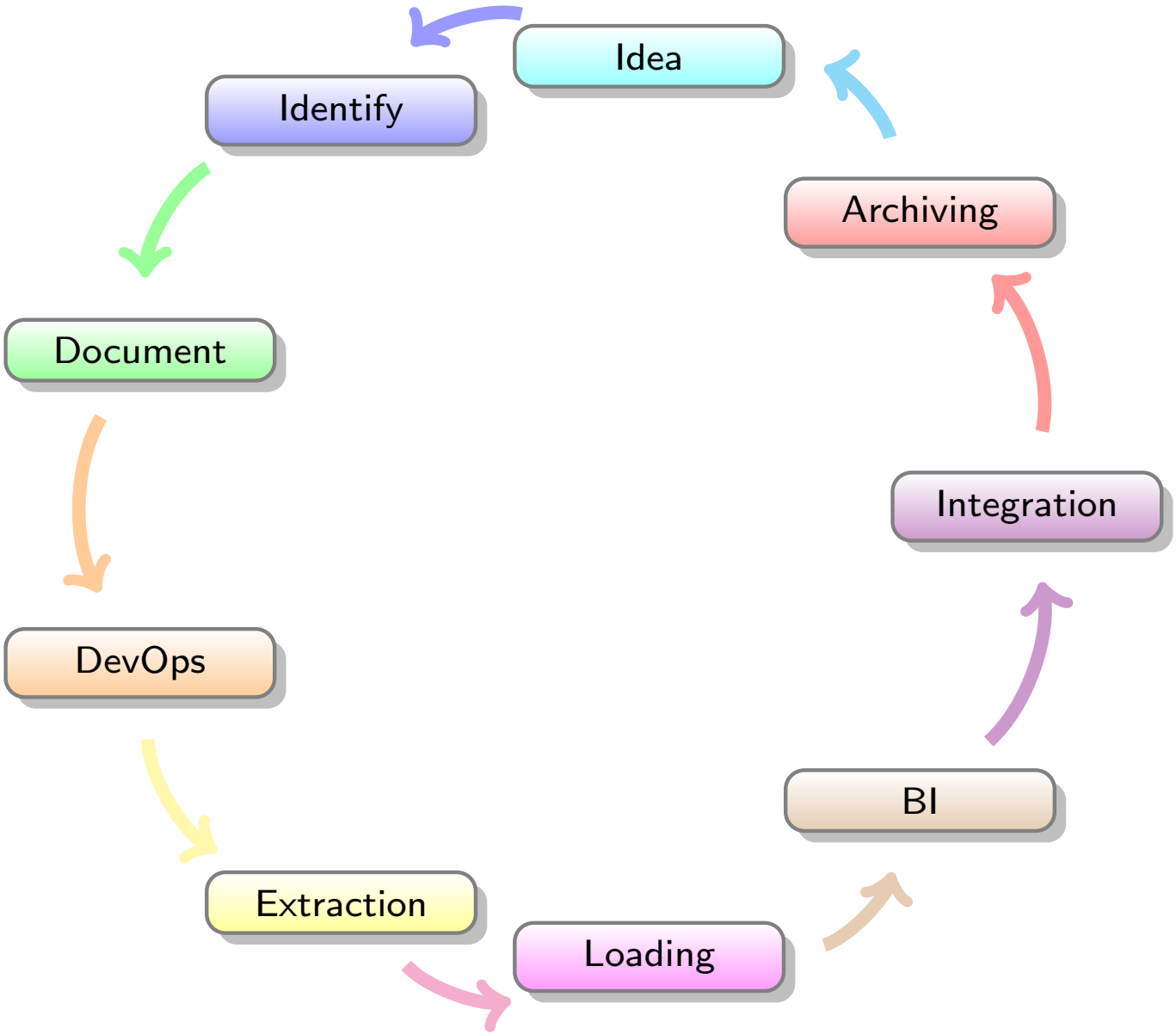
Chapter Objectives

- Be familiar with data management life-cycle.
- Introduction to data warehouse and its usage.
- Motivation to DWH.
- What is the different types of DWH?
- Use cases
- Data Encoding and Formats
- Challenges to build a DWH.
- Data modeling design.

Data Management

- Data are a product.
- Data product has a life-cycle as following (simplified):
  - Question, Idea, or service.
  - Identifying the source of information and the data type ex: (text, images, videos, audio, or sensors).
  - Document all details regarding the data including quality, security, efficiency, and access (consideration during the cycle).
  - Delivery automation (Tools and Process) AKA DevOps cycle.
  - Extraction Process (collection).
  - Transformation ex: (cleansing, Apply business logic, Organize).
  - Loading or store the transformed data based on our usage or use case.
  - Business Intelligence (BI) or data discovery (continues process).
  - Integration and publishing.
  - Data retention or archiving process ex: (Hot or Cold storage).

Data Management Life-Cycle



Motivation to Data Warehouse

- Data could be a product for some companies.

Motivation to Data Warehouse

- Data could be a product for some companies.
- It could be decision support for other products or businesses.

Motivation to Data Warehouse

- Data could be a product for some companies.
- It could be decision support for other products or businesses.
- Reporting the results after pass the data life-cycle will be from storage (Database).

Motivation to Data Warehouse

- Data could be a product for some companies.
- It could be decision support for other products or businesses.
- Reporting the results after pass the data life-cycle will be from storage (Database).
- There are some challenges facing the people who work on data management backend:

Motivation to Data Warehouse

- Data could be a product for some companies.
- It could be decision support for other products or businesses.
- Reporting the results after pass the data life-cycle will be from storage (Database).
- There are some challenges facing the people who work on data management backend:
  - Performance.

Motivation to Data Warehouse

- Data could be a product for some companies.
- It could be decision support for other products or businesses.
- Reporting the results after pass the data life-cycle will be from storage (Database).
- There are some challenges facing the people who work on data management backend:
  - Performance.
  - Integration.

Motivation to Data Warehouse

- Data could be a product for some companies.
- It could be decision support for other products or businesses.
- Reporting the results after pass the data life-cycle will be from storage (Database).
- There are some challenges facing the people who work on data management backend:
  - Performance.
  - Integration.
  - Applying analytical functions.

Motivation to Data Warehouse

- Data could be a product for some companies.
- It could be decision support for other products or businesses.
- Reporting the results after pass the data life-cycle will be from storage (Database).
- There are some challenges facing the people who work on data management backend:
  - Performance.
  - Integration.
  - Applying analytical functions.
- Vendors who are working to solve the above challenges creating their own product of DWH and their ultimate work is to optimize the above points.

Motivation to Data Warehouse (DWH)

Definition (What is Data Warehousing?)

A DWH is defined as a technique for collecting and managing data from varied sources to provide meaningful business insights. It is a blend of technologies and components which aids the strategic use of data.[1]

- The DWH is not a product but an environment.
- It is a process of transforming data into information and make it available to users in a timely manner to make a difference.
- It is an architectural construct of an information system which provides users with current and historical decision support information which is difficult to access or present in the traditional operational data store.
- The DWH is the core of the BI system which is built for data analysis and reporting.

[1] The definition mentioned in this slides copied from guru99.com

Motivation to Data Warehouse

Data warehouse system is also known by the following names:

- Decision Support System (DSS).
- Business Intelligence Solution.
- Executive Information System.
- Management Information System.
- Analytic Application.
- Data Warehouse.

The real concept was given by Inmon Bill. He was considered as a father of the DWH. He had written about a variety of topics for building, usage, and maintenance of the warehouse & the Corporate Information Factory

Motivation to Data Warehouse

Types of Data Warehouse

Enterprise Data Warehouse (EDWH) It provides decision support service across the enterprise. It offers a unified approach for organizing and representing data (DWH Model). It offers data classifications according to the subject with privileges policy.

Operational Data Store (ODS): is a central database that provides an up-to-date (real-time) data from multiple transnational systems for operational reporting into a single DWH.

Data Mart: A data mart is a subset of the data warehouse. It specially designed for a particular line of business, such as sales, finance, sales or finance. In an independent data mart, data can collect directly from sources.

DWH vs ODS vs Data Mart

Metric	DWH	ODS	Data Mart
Latency	Day -1	Real-time	Day -1
Data level	Transnational	Transnational	Summary
Historical	Long-term	Snapshot	Aggregated Long-Term
Size	TB/PB	GB	GB/TB
Orientation	Multi sources	Multi sources	Product
Business Units	Multi organizational units	Product team	Business team

DWH vs Operational databases

- Operational databases (Transactions DB) still working as the backend for the products.

DWH vs Operational databases

- Operational databases (Transactions DB) still working as the backend for the products.
- Data warehouse mainly works as centralized storage for all the source systems regardless of the product type or their functionality.

DWH vs Operational databases

- Operational databases (Transactions DB) still working as the backend for the products.
- Data warehouse mainly works as centralized storage for all the source systems regardless of the product type or their functionality.
- Data warehouse designed to solve the huge amount of data.

DWH vs Operational databases

- Operational databases (Transactions DB) still working as the backend for the products.
- Data warehouse mainly works as centralized storage for all the source systems regardless of the product type or their functionality.
- Data warehouse designed to solve the huge amount of data.
- Most of DWH can't solve the online transactions similar to the transaction DB.

DWH vs Operational databases

- Operational databases (Transactions DB) still working as the backend for the products.
- Data warehouse mainly works as centralized storage for all the source systems regardless of the product type or their functionality.
- Data warehouse designed to solve the huge amount of data.
- Most of DWH can't solve the online transactions similar to the transaction DB.
- Transactions databases have a performance issue while handling a huge amount of data. So, analysis of a huge amount of data (including historical data) we used DWH for this purpose. On the other hand Transactions DB used for online or short historical data based on product type and requirements.

DWH vs Operational databases

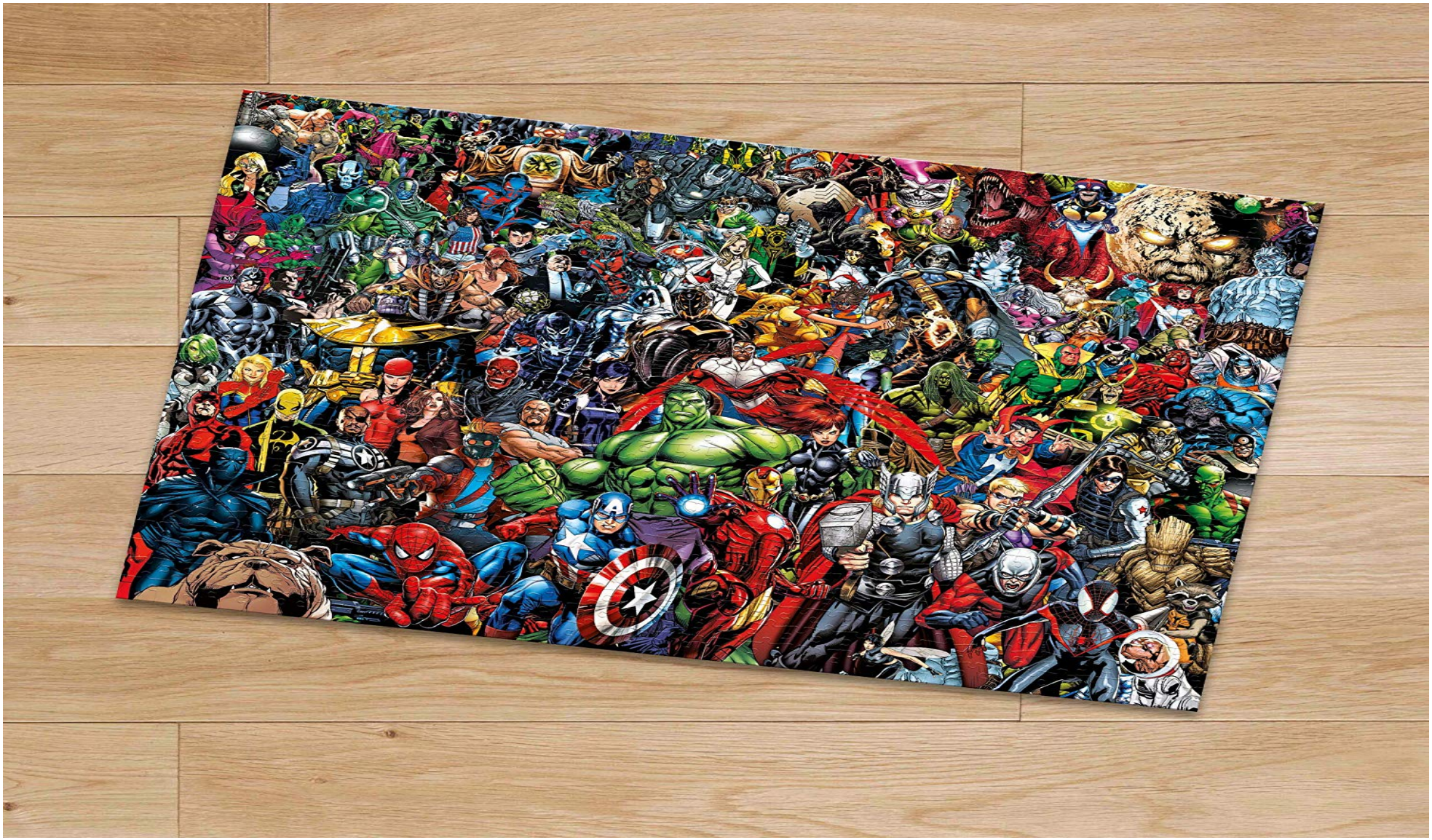
Metric	Transactions DB	DWH
Volume	GB/TB	TB/PB
Historical	Short-term	Long-Term
rows	¡1000M	1000M¿
Orientation	Product	Subject or multi products
Business Units	Product team	Multi organizational units
Normalization	Normalized	Not required (De-normalized in many use cases)
Data Model	Relational	Star Schema or Multi-dim
Intelligence	Reporting	Advanced reporting and Machine Learning
Use cases	Online transactions & operations	Centeralized storage (360°)

Transnational DB Use cases

Transnational DB Use cases

DWH Use cases

DWH Use cases



DWH Use cases

User stories Telecom company.
It has a CRM System backend database reporting the sales. vs Another backend database contains the CRM, Telecom signaling data, IN charging system, Billing
Decision is related to sales or CRM. Decision is related to company strategies.
Analytical model checking the fraud which require a CRM data with customer locations from signaling with Billing details from CAR table.
managing risk of the project in Transaction vs DWH
data model comparison

DWH Characteristics

some details about hot vs cold storage,

Cold storage vs Hot storage

some details about hot vs cold storage,

Data Encoding and Formats

- Any Big Data solution working based distributed systems.

Data Encoding and Formats

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Data Modeling Design

- Any Big Data solution working based distributed systems.

Data Modeling Design

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Introduction To Distributed Systems

Chapter Objectives

- Understand the distributed systems concepts.

Chapter Objectives

- Understand the distributed systems concepts.
- Replication and its usage in distributed systems.

Chapter Objectives

- Understand the distributed systems concepts.
- Replication and its usage in distributed systems.
- Partitioning and its usage in distributed systems .

Distributed Systems Concepts

- Any Big Data solution working based distributed systems.

Distributed Systems Concepts

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Distributed Systems Architecture

- Any Big Data solution working based distributed systems.

Distributed Systems Architecture

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Distributed Systems Challenges

- Any Big Data solution working based distributed systems.

Distributed Systems Challenges

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Design Simple Distributed System

- Any Big Data solution working based distributed systems.

Design Simple Distributed System

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Introduction to Hadoop and Map-Reduce

Chapter Objectives

- Introduction to Hadoop and its echo-systems.

Chapter Objectives

- Introduction to Hadoop and its echo-systems.
- Why we need Hadoop?

Chapter Objectives

- Introduction to Hadoop and its echo-systems.
- Why we need Hadoop?
- Understand the concept of HDFS and Map-Reduce.

Chapter Objectives

- Introduction to Hadoop and its echo-systems.
- Why we need Hadoop?
- Understand the concept of HDFS and Map-Reduce.
- Developing Map-Reduce applications.

Chapter Objectives

- Introduction to Hadoop and its echo-systems.
- Why we need Hadoop?
- Understand the concept of HDFS and Map-Reduce.
- Developing Map-Reduce applications.
- Using HiveQL over Map-Reduce.

Chapter Objectives

- Introduction to Hadoop and its echo-systems.
- Why we need Hadoop?
- Understand the concept of HDFS and Map-Reduce.
- Developing Map-Reduce applications.
- Using HiveQL over Map-Reduce.
- Hadoop advantages and disadvantages with use cases?

Hadoop Architecture

- Any Big Data solution working based distributed systems.

Hadoop Architecture

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Storage

- Any Big Data solution working based distributed systems.

Storage

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

YARN

- Any Big Data solution working based distributed systems.

YARN

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Hadoop I/O

- Any Big Data solution working based distributed systems.

Hadoop I/O

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Processing

- Any Big Data solution working based distributed systems.

Processing

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Map-Reduce

- Any Big Data solution working based distributed systems.

Map-Reduce

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Map-Reduce Components

- Any Big Data solution working based distributed systems.

Map-Reduce Components

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Word-Count Example

- Any Big Data solution working based distributed systems.

Word-Count Example

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Pig

- Any Big Data solution working based distributed systems.

Pig

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Hive

- Any Big Data solution working based distributed systems.

Hive

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

ZooKeeper

- Any Big Data solution working based distributed systems.

ZooKeeper

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Functional Programming

Spark Framework

Spark Framework: Spark Philosophy towards the Engine and the Programming languages

- Any Big Data solution working based distributed systems.

Spark Framework: Spark Philosophy towards the Engine and the Programming languages

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Spark Framework: Spark Basics

- Any Big Data solution working based distributed systems.

Spark Framework: Spark Basics

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Spark Basics

- Any Big Data solution working based distributed systems.

Spark Basics

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Spark Programming using RDDs

- Any Big Data solution working based distributed systems.

Spark Programming using RDDs

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Spark Programming using RDDs

- Any Big Data solution working based distributed systems.

Spark Programming using RDDs

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Spark Programming using RDDs

- Any Big Data solution working based distributed systems.

Spark Programming using RDDs

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Spark Datasets/Dataframe

- Any Big Data solution working based distributed systems.

Spark Datasets/Dataframe

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Spark Datasets/Dataframe

- Any Big Data solution working based distributed systems.

Spark Datasets/Dataframe

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Spark Datasets/Dataframe

- Any Big Data solution working based distributed systems.

Spark Datasets/Dataframe

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Spark on Production

- Any Big Data solution working based distributed systems.

Spark on Production

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Spark on Production

- Any Big Data solution working based distributed systems.

Spark on Production

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Spark For Batch Processing

- Any Big Data solution working based distributed systems.

Spark For Batch Processing

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Building custom input and output connector using Spark

- Any Big Data solution working based distributed systems.

Building custom input and output connector using Spark

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Spark Streaming

- Any Big Data solution working based distributed systems.

Spark Streaming

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Spark Streaming

- Any Big Data solution working based distributed systems.

Spark Streaming

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Spark Streaming

- Any Big Data solution working based distributed systems.

Spark Streaming

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Spark Streaming

- Any Big Data solution working based distributed systems.

Spark Streaming

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Spark using other Programming Languages

- Any Big Data solution working based distributed systems.

Spark using other Programming Languages

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Spark using other Programming Languages

- Any Big Data solution working based distributed systems.

Spark using other Programming Languages

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Spark using other Programming Languages

- Any Big Data solution working based distributed systems.

Spark using other Programming Languages

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Spark For Data Scientist

- Any Big Data solution working based distributed systems.

Spark For Data Scientist

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Spark For Data Scientist

- Any Big Data solution working based distributed systems.

Spark For Data Scientist

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Spark For Data Scientist

- Any Big Data solution working based distributed systems.

Spark For Data Scientist

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Spark Graph Dataframe/Graphx

- Any Big Data solution working based distributed systems.

Spark Graph Dataframe/Graphx

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Spark Graph Dataframe/Graphx

- Any Big Data solution working based distributed systems.

Spark Graph Dataframe/Graphx

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Tuning your Spark Jobs

- Any Big Data solution working based distributed systems.

Tuning your Spark Jobs

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Tuning your Spark Jobs

- Any Big Data solution working based distributed systems.

Tuning your Spark Jobs

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Real World Applications

Massaging Systems

Data Orchestration

NOSQL

Elastic

Data Architecture Design

Appendix

Appendix A- Shell Programming

- Any Big Data solution working based distributed systems.

Appendix A- Shell Programming

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Appendix B- Java Programming

- Any Big Data solution working based distributed systems.

Appendix B- Java Programming

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Appendix C- Scala Programming

- Any Big Data solution working based distributed systems.

Appendix C- Scala Programming

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Appendix D- SQL Programming

- Any Big Data solution working based distributed systems.

Appendix D- SQL Programming

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Appendix E- Oozie Orchestration

- Any Big Data solution working based distributed systems.

Appendix E- Oozie Orchestration

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Appendix F- DWH Concepts and Data Modeling Design

- Any Big Data solution working based distributed systems.

Appendix F- DWH Concepts and Data Modeling Design

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Appendix G- Machine Learning Concepts Data Engineers

- Any Big Data solution working based distributed systems.

Appendix G- Machine Learning Concepts Data Engineers

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?

Appendix H- Docker for Data Engineers

- Any Big Data solution working based distributed systems.

Appendix H- Docker for Data Engineers

- Any Big Data solution working based distributed systems.
- What is distributed systems in brief?