

# Introduction to Distributed File Systems in Hadoop

**Dr. Javad Ghofrani**

**IMC University of Applied Sciences Krems**

**25/2/2020**

# Agenda

---

- Introduction
  - Lecture Info
  - Lecture Goals
  - References
  - Hadoop
- File Systems
  - How does File System Work?
  - Distributed File Systems
- Hadoop Distributed File Systems (HDFS)
  - Different Nodes
  - How Data is Stored in HDFS
  - HDFS Architecture
  - HDFS in Practice
  - Further Research Directions
- Summary

# Lecture Info: Who am I?

---



- Instructor: Dr. Javad Ghofrani
  - Email: [javad.ghofrani@htw-dresden.de](mailto:javad.ghofrani@htw-dresden.de)
  - Website: [ghofrani85.github.io](http://ghofrani85.github.io)
  - Office : Z902, Faculty of Informatics / Mathematics, Dresden University of Applied Sciences
- PhD From TU Clausthal, Postdoc University of Hanover and HTW Dresden
- Research Interests:
  - Distributed Systems and Architectures
  - Industrial Internet of Things
  - Software Product Lines
  - Artificial Intelligence

# What this Lecture is About?

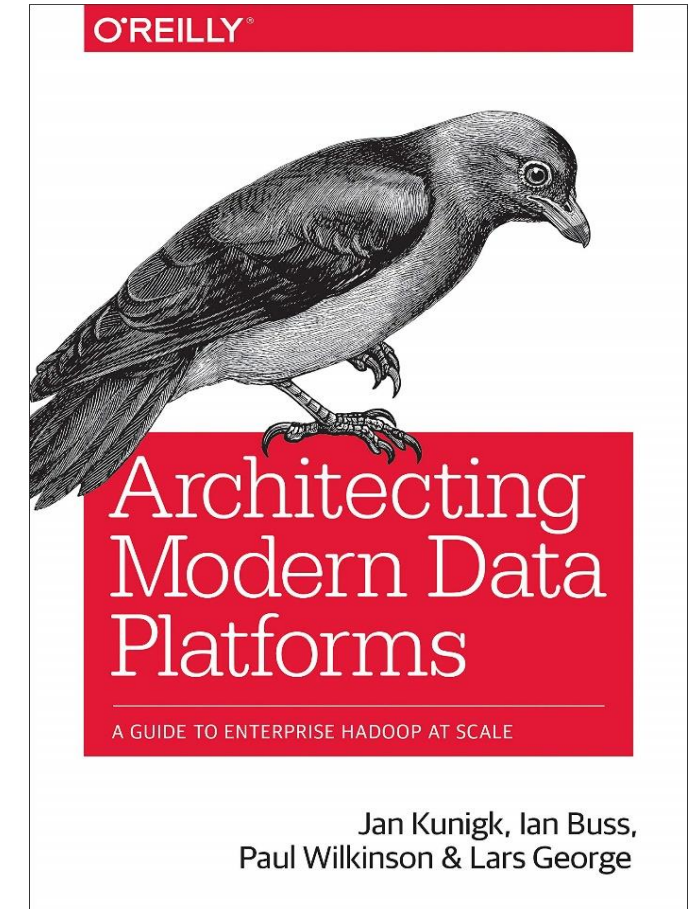
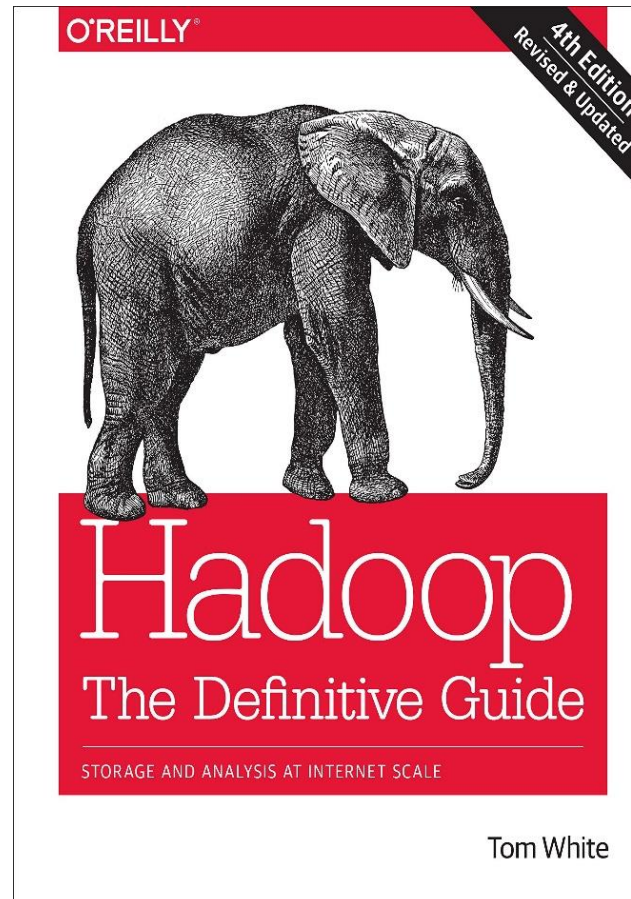
---

- 1: Concept of HDFS
  - Build a strong understanding of concepts of Hadoop Distributed File System in great details
- 2: Practical Aspects of HDFS
  - Get familiar with practical aspects of Hadoop Distributed File System and fundamental knowledge in design that store the data for parallel processing
- 3: Research Trends Around HDFS
  - Get familiar with the state-of-the-art in research around HDFS and related challenges

# Introduction

## References

<https://tinyurl.com/HDFS20>



# Introduction

---



# Introduction

---

- The Hadoop Project is a Free reimplementation of Google's in-house MapReduce and distributed file system(GFS)[1]
- Originally written by Doug Cutting & Mike Cafarella, who also created Lucene and Nutch
- Now hosted and managed by the Apache Software Foundation

[1] Ghemawat, Sanjay, Howard Gobioff, and Shun-Tak Leung. "The Google file system."  
*Proceedings of the nineteenth ACM symposium on Operating systems principles*. 2003.



# Introduction

---

**What does a File System mean?**

**How does it work?**



# Introduction

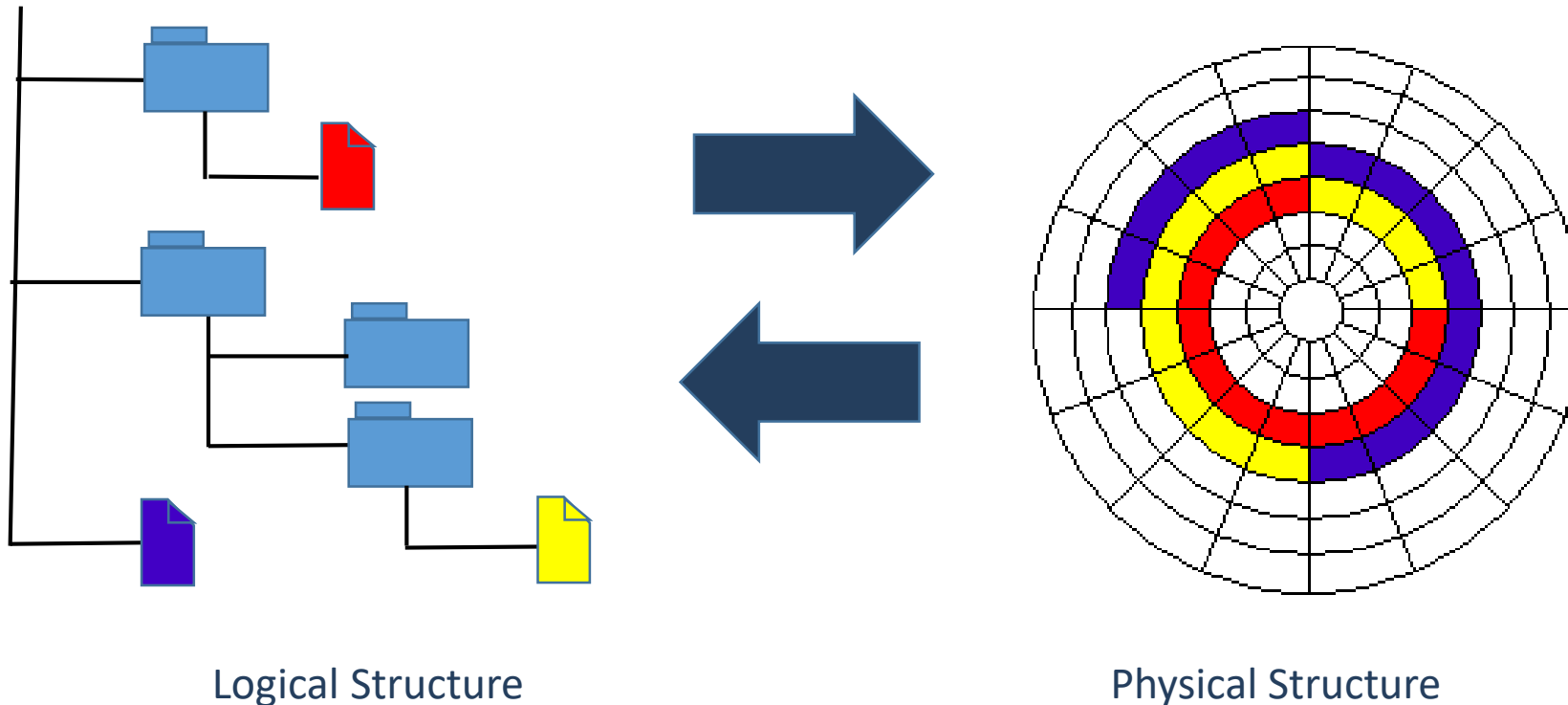
- An example from daily life: locating a book in the library
  - How to find books in a library?

Searching the Catalogue!



# File System

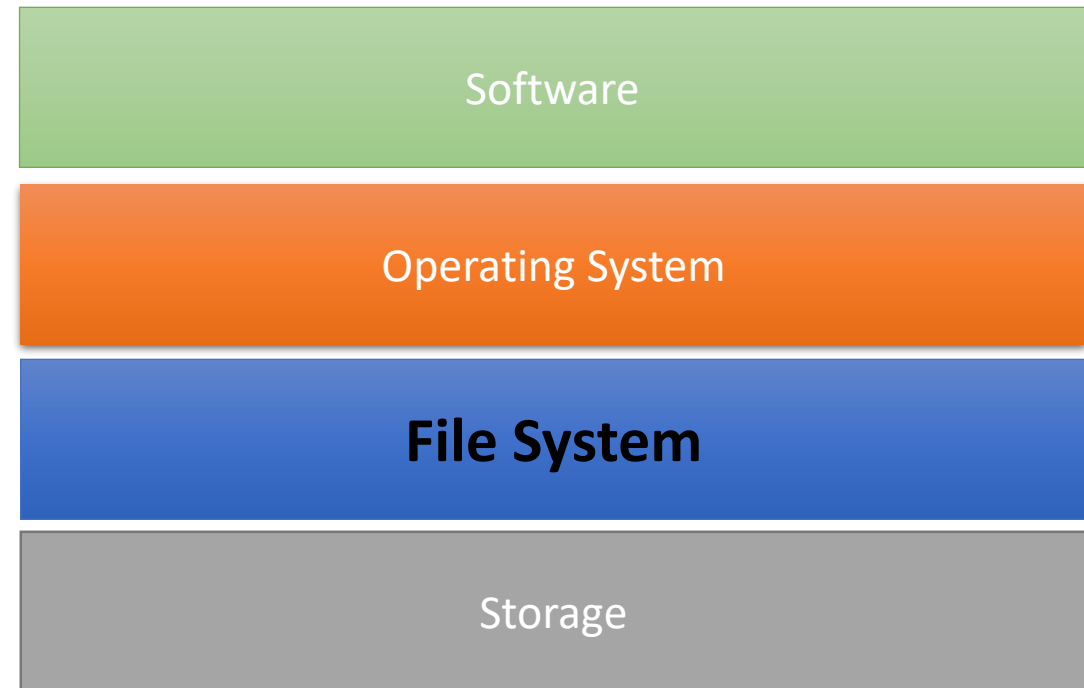
File System (often abbreviated to FS) provides an abstraction layer between logical and physical structure of the storage



# File System

---

- File System describes how the data is stored on the storage
- File System provides efficient and convenient access to disk
  - Create
  - Open
  - Read/write
  - Close
  - Delete



# File System

- Metadata: information about files in file system
  - Information about files (e.g., path, name, type of file, dates of creation and modification, permission to access or change, etc.)
- Different features and properties
  - E.g., FAT 32, NTFS, ext\* ( ext2, ext3, ext4...), HFS+, XFS, JFS
- Different structure and logic

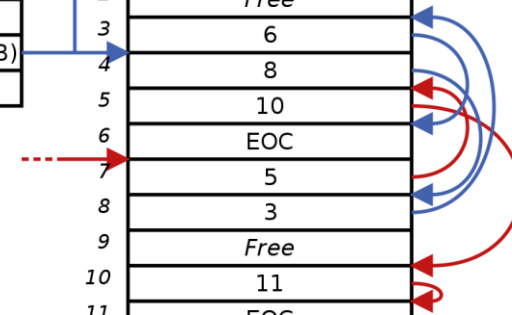
**Directory table entry (32B)**

Filename (8B)
Extension (3B)
Attributes (1B)
Reserved (1B)
Create time (3B)
Create date (2B)
Last access date (2B)
First cluster # (MSB, 2B)
Last mod. time (2B)
Last mod. date (2B)
First cluster # (LSB, 2B)
File size (4B)

**File allocation table**

0	Volume info
1	
2	Free
3	
4	6
5	8
6	10
7	EOC
8	5
9	3
10	Free
11	11
12	EOC
13	...

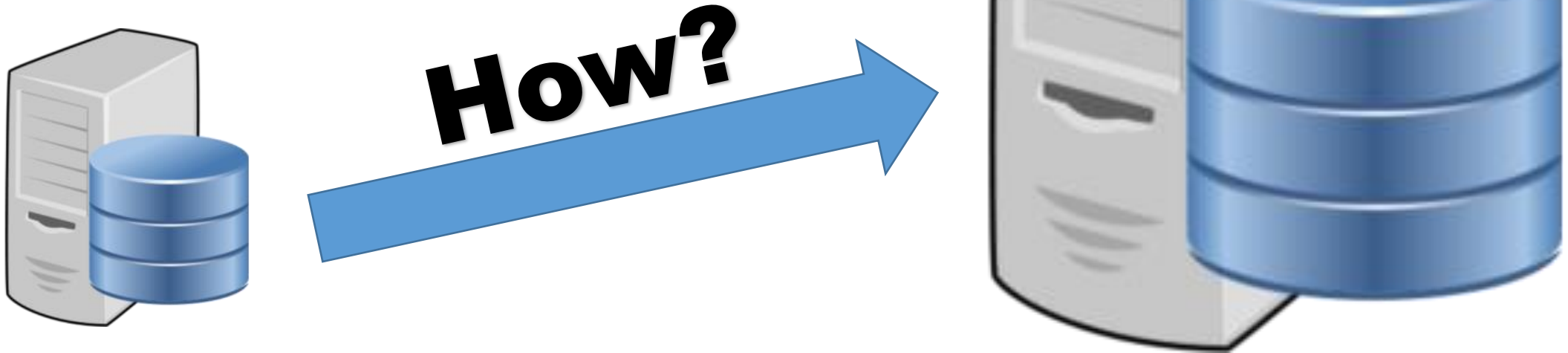
← 32b →



# Challenge

---

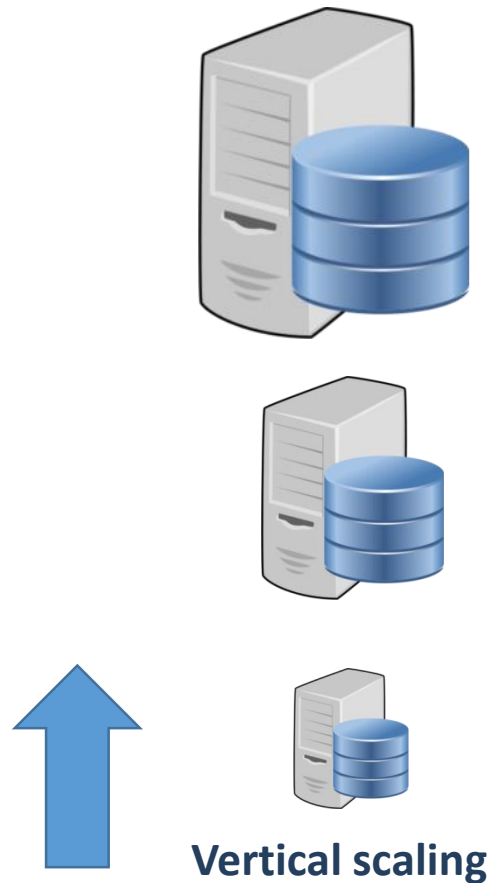
Huge amount of data  
requires  
larger storage space



# Distributed File System

---

- How to provide more storage for big size of data?

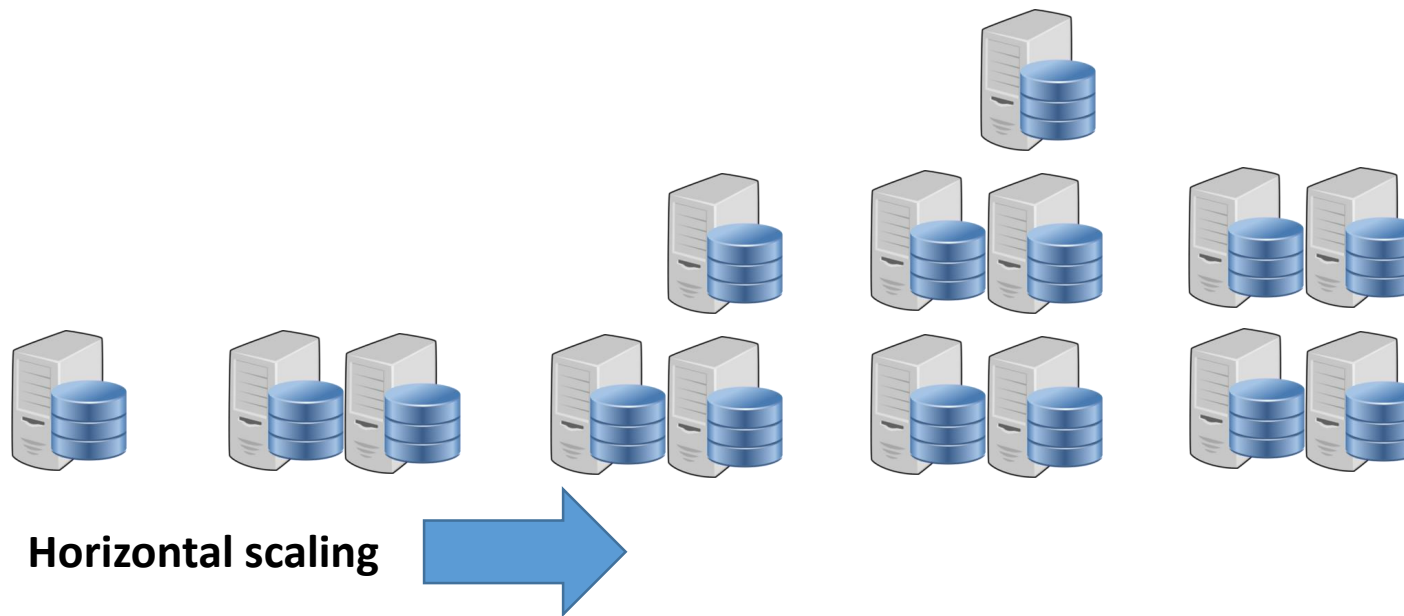


- **Challenges:**
  - Limited computation power
  - Limited storage
  - Flexibility

# Distributed File System

---

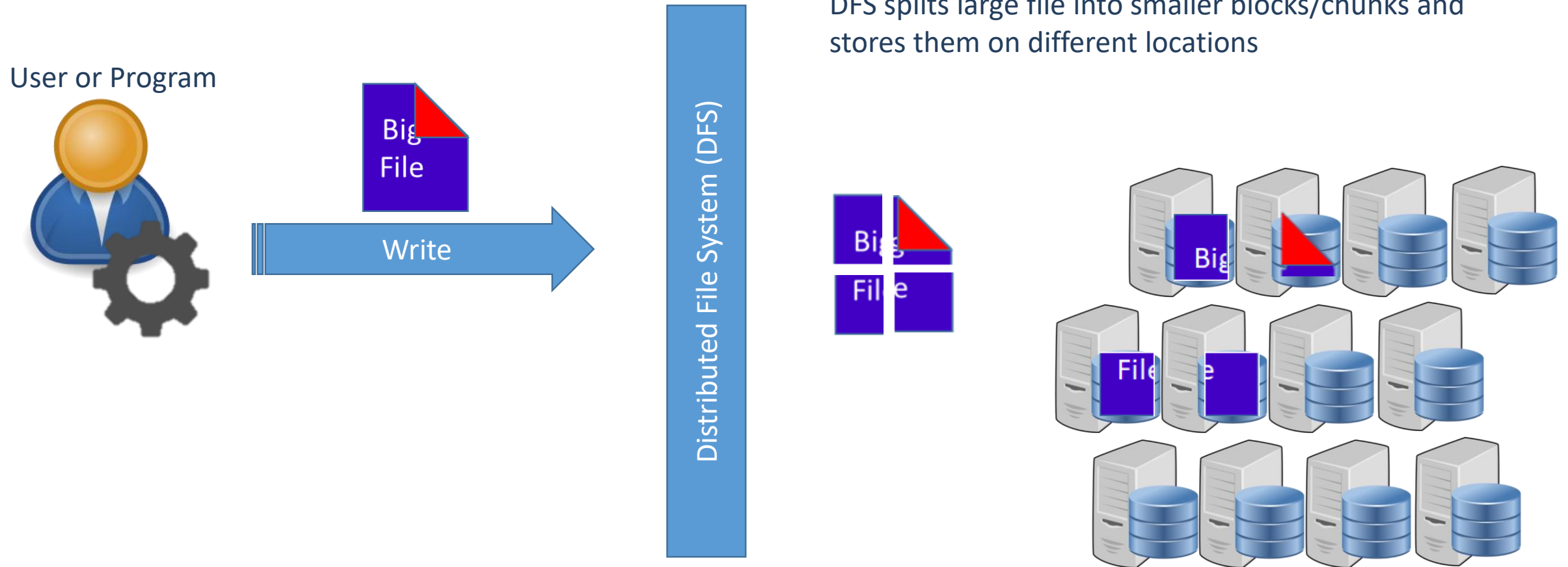
- How to provide more storage for big size of data?



- Advantages
  - Flexibility and reliability
  - Resources optimization
    - Scale up/down
- Challenges
  - Transparency — Abstraction
  - Distribution of data and programs
  - Deal with network problems
    - Communication speed
    - Network failure

# Distributed File System

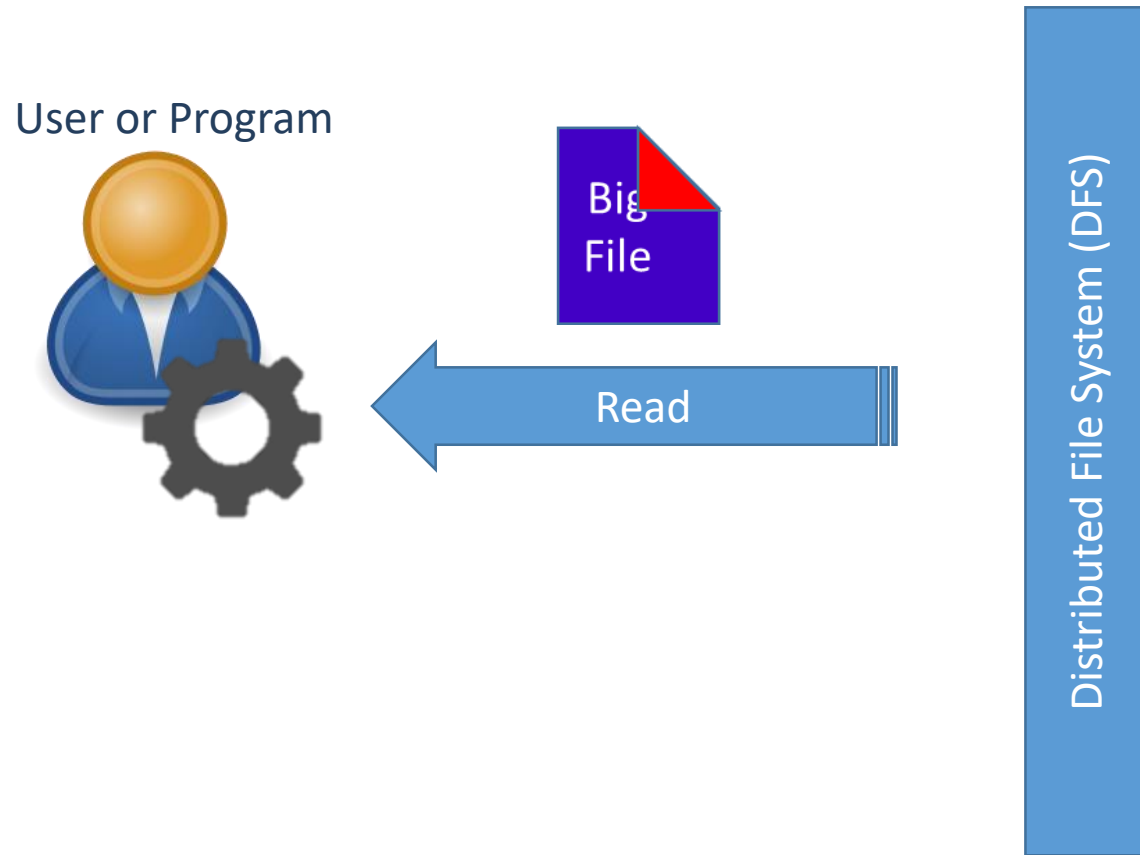
- Transparency — Abstraction



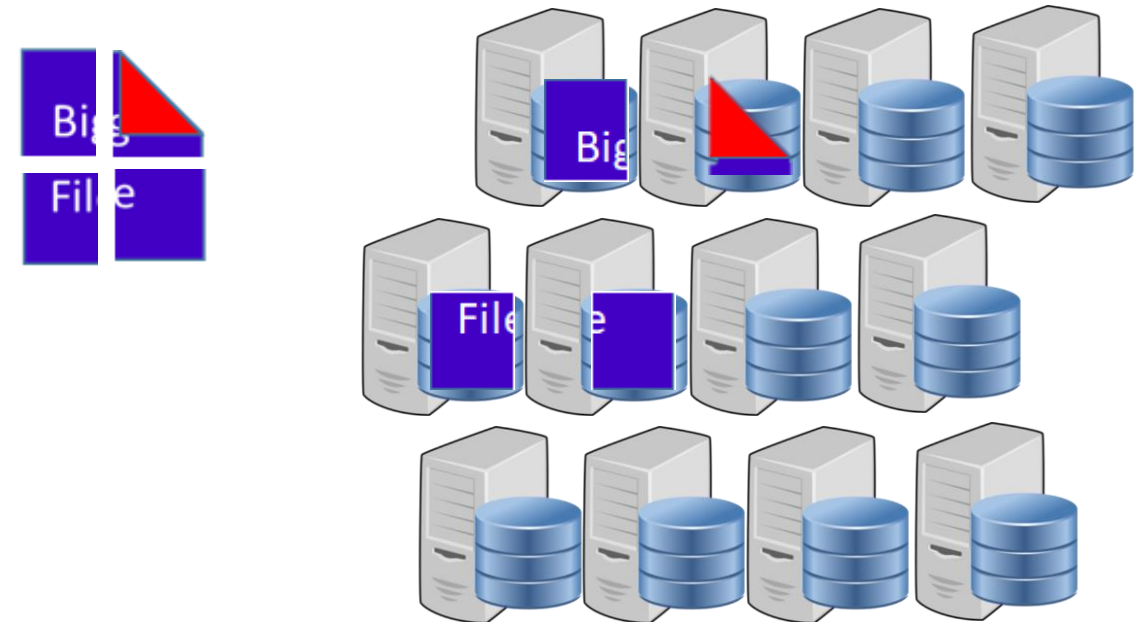


# Distributed File System

- Transparency — Abstraction

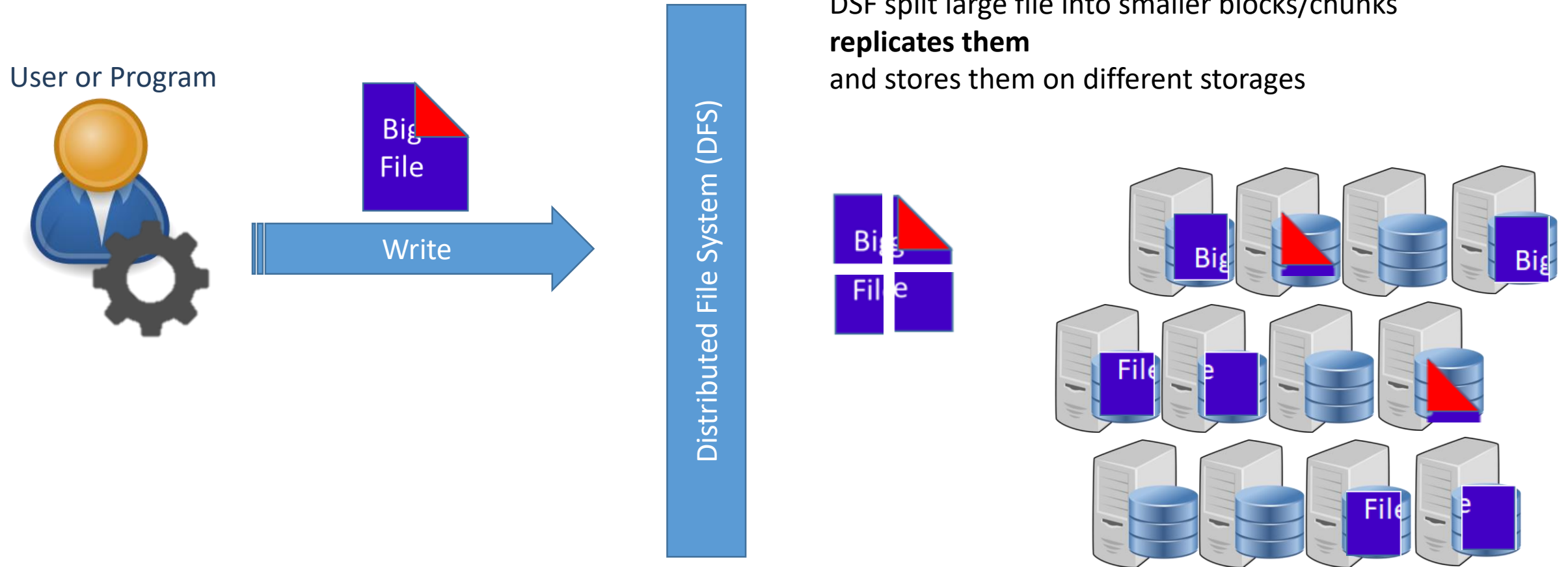


DSF merges small blocks/chunks and creates the big file



# Distributed File System

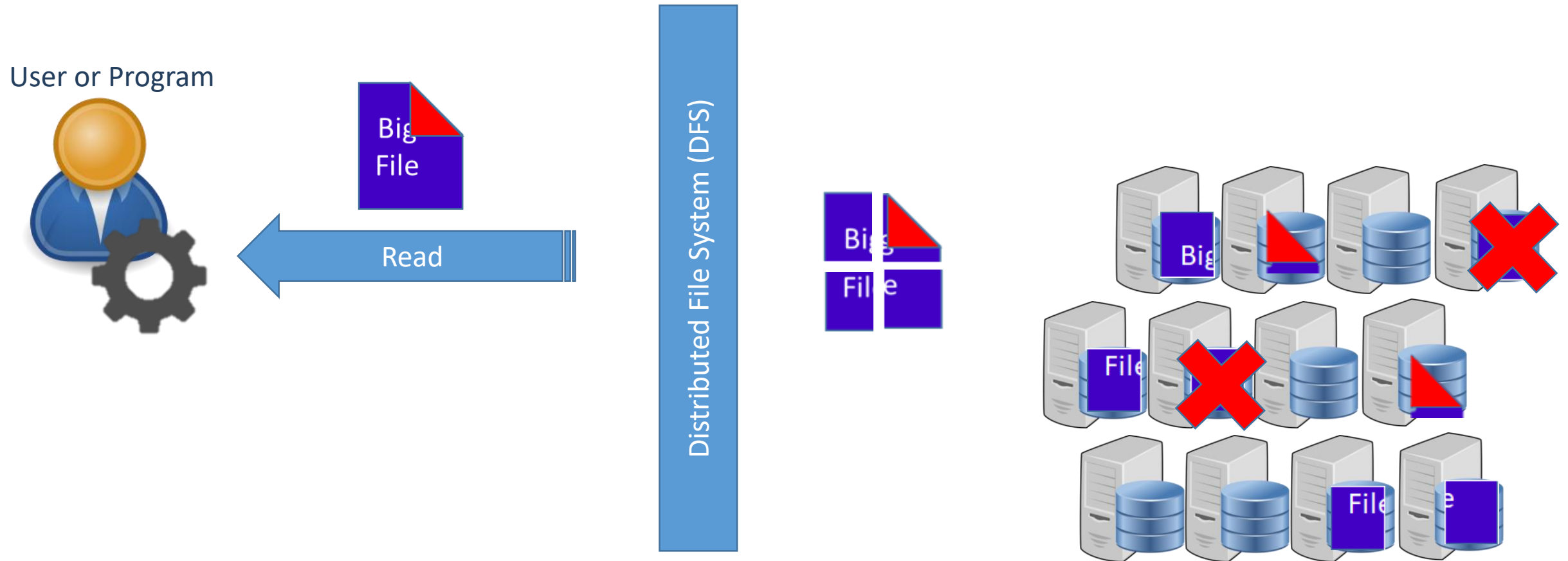
- Reliability



# Distributed File System

- Reliability

In case of a failure of one or more machines, data is still safe

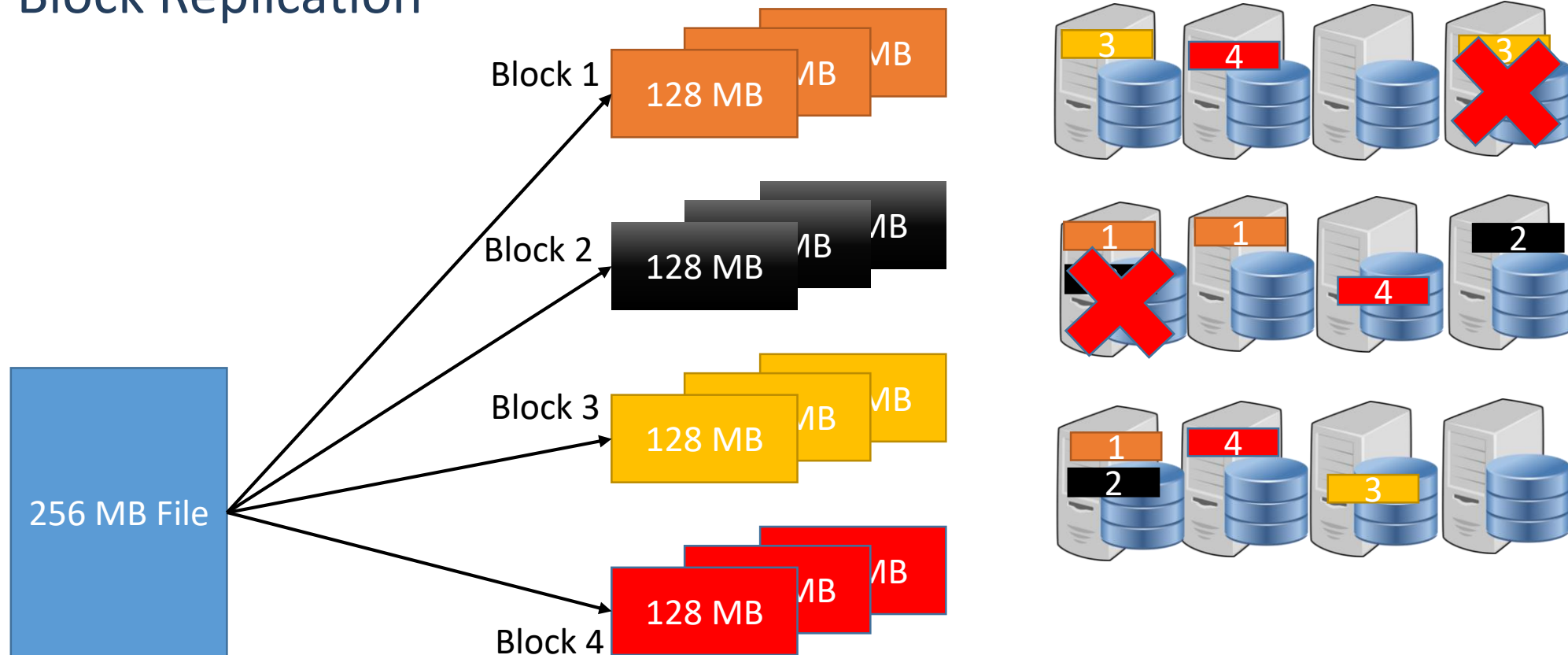


# Hadoop Distributed File System (HDFS)

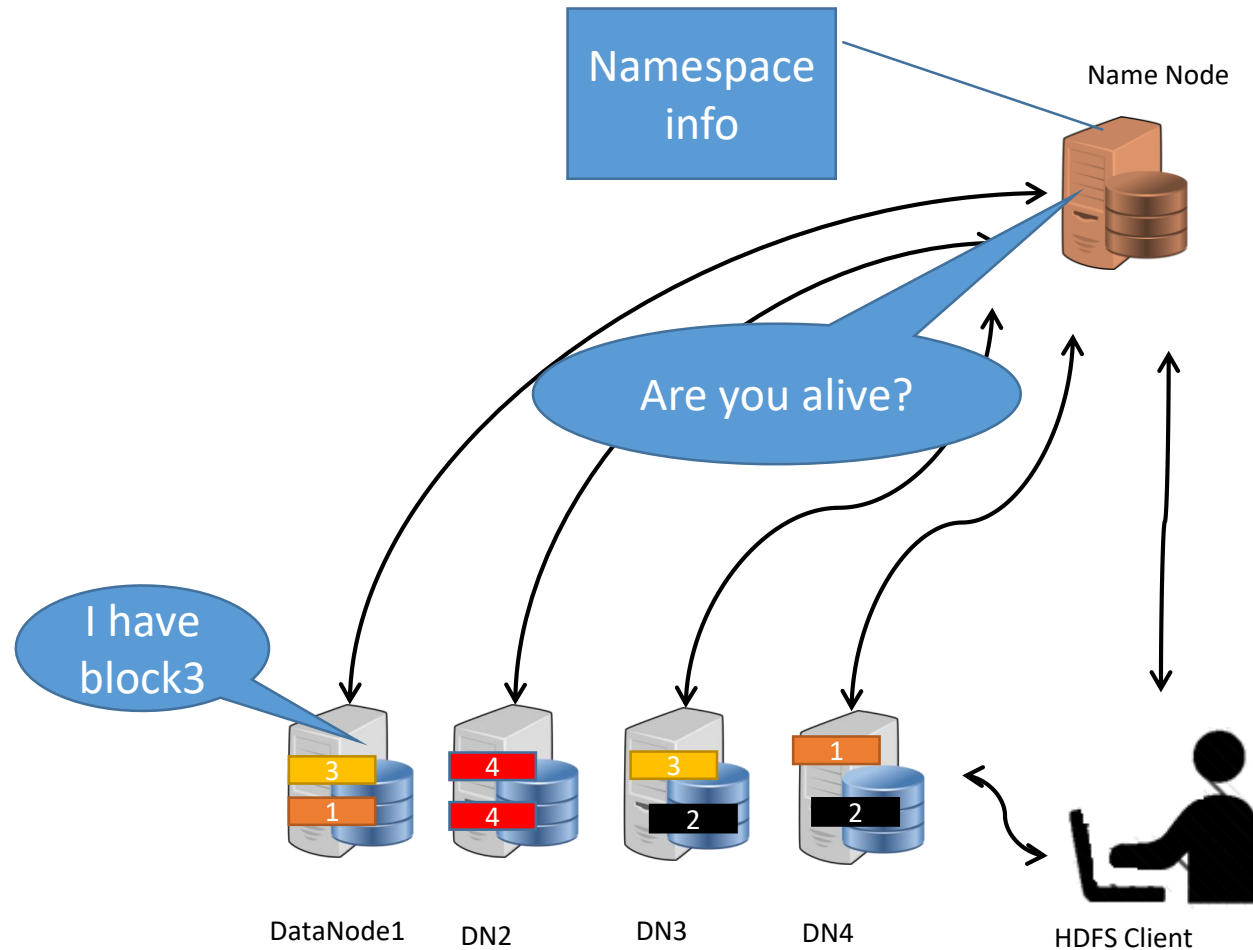
- HDFS Block

3 Copies of Each Block → Default Replication Factor = 3

- Block Replication



# Hadoop Distributed File System (HDFS)



# Hadoop Distributed File System (HDFS)

---

- Name Node (primary / master Node)
  - Very few nodes
  - Store Namespace (metadata) Information
  - Maintains and manages the slave nodes
    - Heartbeat
  - It should be deployed on reliable hardware
- Data Node (secondary /slave Node)
  - Majority of nodes
  - Runs with commodity hardware
    - Cheap, no special hardware needed
- HDFS Client
  - Located on user side and fulfills user requests
  - Interacts with Name Node and Data Node

# Hadoop Distributed File System (HDFS)

---

- HDFS Block: Files are broken-up into blocks
  - Size of each block = 128 MB (Hadoop 2. x) and 64 MB (Hadoop 1. x)
- Block Replication: Multiple copies of each block are stored across the cluster on different nodes
  - Default 3: two copies are in same rack and one outside the rack
  - Storage capacity is reduced (to 1/4 by default)
    - 3 replications and 1 additional scratch space for temporary data
  - Provides high availability, fault tolerance, and reliability
- Advanced Topics: e.g., load balancing, rack awareness, replication placement, read and write, programming

# HDFS in Practice

---

General format of hdfs commands

**hdfs dfs –[normalUnixCommand] –[normalUnixArguments]**

## **Making a directory in local file system**

**Unix:** mkdir

**Example:** mkdir /tmp/directory1

## **Making a directory in hadoop distributed file system**

**HDFS:** hdfs dfs -mkdir

**Example:** hdfs dfs -mkdir /tmp/directory1

## **Copy from local file system to HDFS**

hdfs dfs -put <localsrc> ... <dst>

- **Example:** hdfs dfs -copyFromLocal Sample1.txt /tmp/diretory1/

Copy/Upload Sample1.txt available in current direcotory (local default) to /user/javad/directory1 (hdfs path)



# HDFS in Practice

---

```
ls
hdfs dfs -ls
hdfs dfs -mkdir mydir
hdfs dfs -ls
```

```
hdfs dfs -rm mydir
hdfs dfs -rm -r mydir
hdfs dfs -ls
```

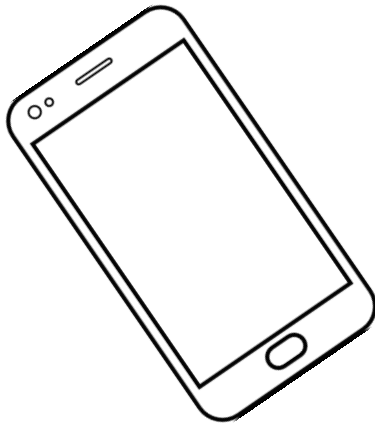
```
ls
hdfs dfs -put hadoop-3.1.3.tar.gz /tmp/
hdfs dfs -ls /tmp/
ls
du -h hadoop-3.1.3.tar.gz
hdfs dfs -du -h /tmp/hadoop-3.1.3.tar.gz
hdfs fsck /tmp/hadoop-3.1.3.tar.gz
```

# Demo

# HDFS in Practice

---

- BigData Analytics
  - Processing huge volume of data in parallel fashion



- Statistical analysis, ETL Processing, Business Intelligence
- Not recommended for Real Time use cases

# Research Directions

---

- General:
  - Polato, Ivanilton, et al. "**A comprehensive view of Hadoop research—A systematic literature review.**" *Journal of Network and Computer Applications* 46 (2014): 1-25.
  - Adam, Khalid, et al. "**Bigdata: Issues, challenges, technologies and methods.**" *Proceedings of the International Conference on Data Engineering 2015 (DaEng-2015)*. Springer, Singapore, 2019.
- Performance:
  - Alange, Neeta, and Anjali Mathur. "**Small Sized File Storage Problems in Hadoop Distributed File System.**" *2019 International Conference on Smart Systems and Inventive Technology (ICSSIT)*. IEEE, 2019.
  - Bende, Sachin, and Rajashree Shedge. "**Dealing with small files problem in hadoop distributed file system.**" *Procedia Computer Science* 79 (2016): 1001-1012.
  - Dai, Wei, Ibrahim Ibrahim, and Mostafa Bassiouni. "**An improved replica placement policy for Hadoop distributed file system running on cloud platforms.**" *2017 IEEE 4th International Conference on Cyber Security and Cloud Computing (CSCloud)*. IEEE, 2017.
  - Ciritoglu, Hilmi Egemen, et al. "**Towards a better replica management for hadoop distributed file system.**" *2018 IEEE International Congress on Big Data (BigData Congress)*. IEEE, 2018.
- Security:
  - Saraladevi, B., et al. "**Big Data and Hadoop-A study in security perspective.**" *Procedia computer science* 50 (2015): 596-601.
  - Wang, Fulin, et al. "**Complete Data Deletion Based on Hadoop Distributed File System.**" *Proceedings of the 3rd International Conference on Computer Science and Application Engineering*. 2019.

# What We Learned

---